International Journal on Advances in Networks and Services

IARIA

2012 vol. 5 nr. 1&2

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

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Networks and Services
Volume 5, Number 1 & 2, 2012

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

Gordana Gardasevic, University of Banja Luka, Bosnia and Herzegovina
Marc Gilg, University of Haute-Alsace, France
Debasis Giri, Haldia Institute of Technology, India
Markus Goldstein, DFKI (German Research Center for Artificial Intelligence GmbH), Germany
Luis Gomes, Universidade Nova Lisboa, Portugal
Anahita Gouya, Solution Architect, France
Mohamed Graiet, Institut Supérieur d'Informatique et de Mathématique de Monastir, Tunisie
Christos Grecos, University of West of Scotland, UK
Vic Grout, Glyndwr University, UK
Yi Gu, University of Tennessee, Martin, USA
Angela Guercio, Kent State University, USA
Xiang Gui, Massey University, New Zealand
Mina S. Guirguis, Texas State University - San Marcos, USA
Tibor Gyires, School of Information Technology, Illinois State University, USA
Keijo Haataja, University of Eastern Finland, Finland
Gerhard Hancke, Royal Holloway / University of London, UK
R. Hariprakash, Arulmigu Meenakshi Amman College of Engineering, Chennai, India
Go Hasegawa, Osaka University, Japan
Hermann Hellwagner, Klagenfurt University, Austria
Eva Hladká, CESNET & Masaryk University, Czech Republic
Hans-Joachim Hof, Munich University of Applied Sciences, Germany
Razib Iqbal, Amdocs, Canada
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
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

International Journal on Advances in Networks and Services
Volume 5, Numbers 1 & 2, 2012

CONTENTS

A Novel Approach to the Adaptive Allocation of Bandwidth in IP/MPLS Networks in Conditions of Heavy Network Load

Tarik Čaršimamović
Directorate for Information Technologies
BH Telecom JSC
Sarajevo, Bosnia and Herzegovina
tarik.carsimamovic@bhtelecom.ba

Enio Kaljić, Mesud Hadžialić
Faculty of Electrical Engineering
University of Sarajevo
Sarajevo, Bosnia and Herzegovina
enio.kaljic@etf.unsa.ba, mesud.hadzialic@etf.unsa.ba

Abstract - **In this paper, an algorithm for adaptation layer in order to improve fairness in bandwidth allocation among different traffic classes in IP/MPLS networks under heavy traffic load is proposed. A definition of the blocking frequency of traffic flows at the entry of autonomous network domain and proportional-priority coefficient per traffic class are proposed and used as the input parameters of the adaptation mechanism. In order to evaluate the validity of the proposed algorithm, proper simulation tool is needed and for these purposes OPNET Modeler 14.5 is extended with the modules for adaptation process. Development methodology for the design of modules for adaptation process within network simulators is also proposed. The simulation results proved the hypothesis that with a proper adaptation layer, improvement of the fairness of bandwidth allocation among different traffic classes under heavy network load and at the same time keeps the required QoS conditions in the preferred boundaries is possible.**

Keywords - Adaptation layer; bandwidth allocation; blocking frequency; LSP; MPLS; NGN; proportional-priority coefficient; RSVP.

I. INTRODUCTION

One of the key requirements of the new generation network (NGN) environment is that the network is capable of handling an ever-increasing demand uncertainty, both in volume and time. In such environment, very often the total traffic demands exceed the available network capacity, and the traffic classes with higher priority could occupy the entire network capacity leaving no space for traffic flows with lower priority. Proper adaptation mechanisms could give acceptable results in adequate bandwidth allocation to the traffic variation and in fair treatment of all traffic classes. In the paper [1] was carried out testing of proposed adaptation layer in the case of normal network load. Mechanisms used for bandwidth allocation should be able to manage requests by taking into account at least three parameters: class of service, priority and the requested bandwidth. Several research works in the field of bandwidth management, including [20] [21], have taken in consideration only two parameters out of those three. We used all three parameters in our algorithm, as can be seen in generic architecture of adaptation layer (Fig. 1) and in flow chart (Fig. 2). The fairness in the resource allocation among traffic flows depends on algorithm used during the process of adaptation to the real conditions of traffic load. The fairness of adaptation algorithm represents the ability of the model to distribute available resources in such manner that the probability of traffic blocking for any particular traffic class is the same as the overall blocking probability. We can use ratio P_i of the allocated resources G_i to the requested resources B_i of the particular traffic flow demand $P_i = (G_i / P_i) x100$ as a measure of the algorithm fairness. Three types of fairness index are possible [15]: balanced fairness, max-min fairness and proportional fairness. Many of researches including [21] [24] used max-min approach as a tool to achieve fair distribution of network resources among traffic classes. Max-min fairness assumes that is not possible to increase rate of any connection without decreasing a rate of maximum value allocated to another connection. According to the results shown in this reference, proportional type of fairness is the most suitable type in the case when the network resources are distributed among different traffic classes and when the adaptive method of resource allocation is used. In the same paper, fairness index J for the proportional type of fairness among n traffic classes is proposed as such:

$$J = \frac{(\sum_{i=1}^{n} P_i)^2}{n\sum_{i=1}^{n} P_i^2} \quad (1)$$

where P_i is the fairness of traffic class i. If the value of the fairness index is equal to 1 (J = 1) there is fairness across all flows. If the value of the fairness index J is higher than 0.9, or in an extreme situation higher than 0.8, one can say that the resource allocation mechanism is fair [3]. Otherwise, variations in resource distribution are significant and blocking percentage of the lower-priority traffic classes is outside of the acceptable margins.

NGN is a packet-oriented network supporting Quality of Service (QoS) based on different type of transport technologies. The most preferred protocol in NGN is IP. There are different approaches for the QoS provisioning in IP based networks: Integrated Services (IntServ), Differentiated Services (DiffServ), combined IntServ/DiffServ, Multiprotocol Label Switching (MPLS), etc. [2]. MPLS is a popular transport technology that uses labels which are imbedded between layer two and layer three headers in order to forward packets. Packets are forwarded by switching packets on the basis of labels and not by

routing packet based on IP header. One of the major advantages of MPLS networks is the inherent support to traffic engineering. We can also use a combination of MPLS and DiffServ and treat packets of the same Forward Equivalence Class (FEC) in accordance with the DiffServ procedure. Using MPLS Traffic Engineering (MPLS-TE) based on the network state detection we can balance traffic load among different Label Switched Paths (LSPs), but we cannot dynamically change allocated bandwidth to the LSPs [25]. MPLS-TE can be used to shift traffic form overload paths to alternate path with free bandwidth, but it does not contain inherent QoS features. These features should be designed and deployed separately on top of MPLS tunnel (what is subject to adaptation algorithm). Although the MPLS-TE technology uses extension to the RSVP, the MPLS-TE RSVP reservations serve solely as an accounting mechanism. This prevents link oversubscriptions but does not result in any QoS actions.

In order to adapt to dynamics of traffic demands and to allocate sufficient bandwidth to the LSPs, as well as to improve fairness in the resource allocation among traffic flows, we introduce adaptation layer, working in two regimes:

- fuzzy controller regime, when the overall traffic demand is elastic and in average less than network capacity. In this case, the adaptation process is realized by the means of fuzzy logic [19],
- proportional-priority regime, when the overall traffic demand is higher than the network capacity. In this case the adaptation process allocates bandwidth among traffic classes in such a manner that minimal bandwidth is guaranteed to each traffic class and the rest of network capacity is shared on the proportional basis among traffic classes (equation 4).

The adaptation layer supports dynamic exchange between fuzzy controller regime and proportional-priority regime depending on the ratio between traffic load and the network capacity C. When the network load less than its capacity, all requests for bandwidth can be served. With regard to the possible large variation in the intensity of traffic flows, adaptation layer uses fuzzy controller that effectively predicts the variation. When the load is greater than its capacity, large variations in the intensity of traffic flows are not possible. Then there is no need for rapid changes in the allocated bandwidth, and adaptation layer uses a proportional-priority bandwidth allocation regime.

In order to prove validity of our adaptation layer concept and sustainability of the fairness improvement concept of bandwidth allocation among traffic flows, we need proper simulation tools. Because there are no network simulators supporting the proposed adaptation layer algorithm and dynamics of this algorithm, we established a methodology for development of adaptation layer within network simulators and we also developed the adaptation layer code in C++ within OPNET core structure of node model (Label Edge Router - LER) and within core structure of process model of the Resource Reservation Protocol (RSVP-TE) used in the OPNET modeler.

II. MECHANISMS AND ARCHITECTURES FOR ADAPTIVE TREATMENT OF TRAFFIC DEMANDS

The goals of adaptive treatment of traffic demand in NGN are to:

- Fulfill QoS requests of any traffic class,
- Reduce drops of any traffic flows,
- Decrease congestion within network,
- Rise efficiency of network capacity.

In order to successfully achieve those goals, appropriate mechanisms for the bandwidth allocation, for the routing optimization and for reaction to the failure conditions are needed. During the research project COST 257 [4], several types of reactive and preventive approaches for network control were investigated:

- the flow control scheme (fluid flow model, discrete-time Markov model or control theory model) for reactive approach,
- the admission control method (Measurement Based Admission Control - MBAC, Traffic Description Based Admission Control - TDBAC, Experience Based Admission Control - EBAC or End-point Admission Control - EAC) for preventive approach,
- the active queue management or fuzzy congestion control as a new control trends.

Preventive controls usually try to limit the number of connections or to enforce connection to use only a limited amount of resources. In IP networks, the specifications of protocols such as RSVP or MPLS make admission control possible. There has been a variety of efforts with regards to admission control [22] [23] [24]. All of them can be categorized into distributed approach or centralized approach. In distributed approach, nodes act independently relying on observed behavior rather than explicit reservation of free resources of the network. This approach leads typically in over-provisioning of the network's capacity in order to bypass imprecision of the probe data. On the other side in centralized approach all new connections must be approved through bandwidth broker. While the centralized approach can offer a precise allocation of resources, it suffers from scalability limitations. We applied an ingress node oriented resource management which combined scalability of distributed approach with the efficiency of centralized approach. MPLS traffic engineering is aimed at optimizing the network path to ensure efficient allocation of network resources, thereby avoiding the occurrence of congestion on the links [2]. During the research project Tequila (Traffic Engineering for QoS in Internet at Large Scale) [6], it was shown that a combination of MPLS and DiffServ could be acceptable solution for load balancing in IP networks when multi-path routing is used. Also, during the research project COST 239 [5] it was shown that, in case of large traffic load, the highest efficiency of the resource usage is in the networks which use border-to-border budget based network admission control (BBB NAC) as a budget-oriented method for allocation of virtual bandwidth. BBB NAC could be realized using RSVP extension for LSP tunnels establishing explicit LSPs with guaranteed bandwidth.

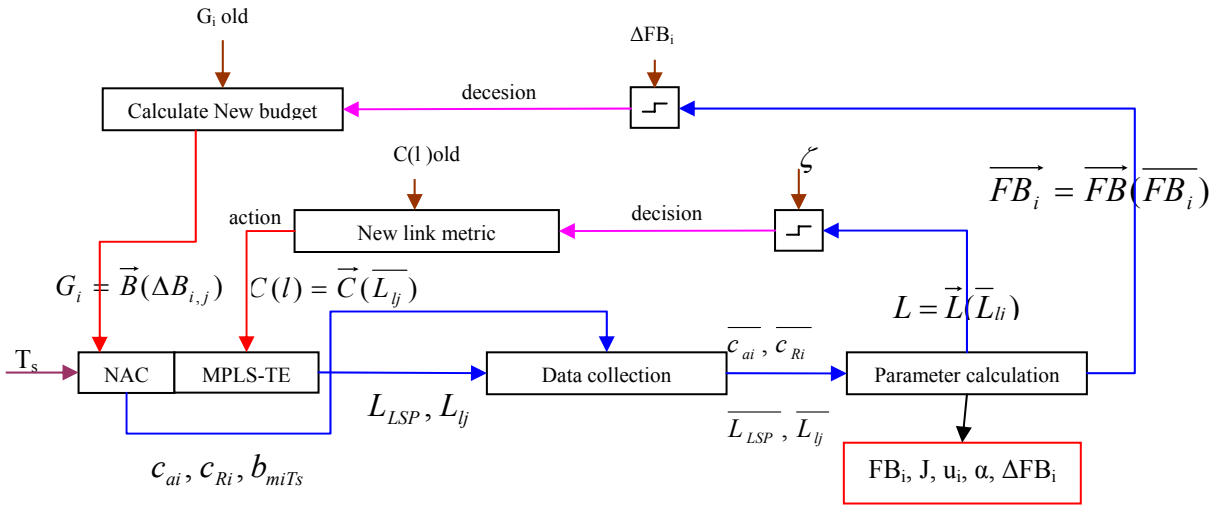


Figure 1. Generic architecture of adaptation layer

Several research projects investigate possible architecture of dynamic provisioning of QoS to the particular traffic flow. The basic result of KING (Key Components for Internet of the Next Generation) project [18] includes development of adaptive architecture in which, by continuous monitoring of network conditions, the network parameters could be adapted to traffic demand.

We customized this architecture according the requests of the logic of our adaptation algorithm, as shown in Fig 1. The figure shows the following parameters:

- c_{ai} - amount of accepted traffic of i-th traffic class,
- c_{ri} - amount of rejected traffic of i-th traffic class,
- b_{miTs} - number of rejected reservation for i-th traffic class within one sample period,
- T_s - sample time,
- L_{LSP} - LSP traffic load,
- L_{ij} - traffic load of i-th traffic class on j-th link,
- u_i - utility function according equation 6,
- FB_i - blocking frequency of i-th traffic flow,
- ζ - decision criterion.

We combined the admission control method (MBAC – BBB NAC) with policy based control of adaptation layer to dynamically adapt budget of the NAC in order to decrease blocking frequency and to raise fairness of bandwidth allocation among traffic classes.

III. DESIGN OF MODEL FOR ADAPTATION PROCESS

A. *One Possible Solution of Adaptation Layer*

In [11] - [14], an active queue management as network control mechanism is proposed. This approach requires execution of adaptation layer processes at every node in the network, so making it unsuitable for MPLS based networks.

In this paper we use a different approach for solution of the adaptation layer algorithm in order to increase the resource usage efficiency, to provide proper QoS to any traffic class and to improve fairness in bandwidth allocation among traffic flows within MPLS based networks. In the rest of this section we give a brief overview of the solution and corresponding part of pseudo code of algorithm. A full description, that includes a detailed explanation of the algorithm, is given in [16]. In this paper, MPLS is used as a transport technology on the network layer. Network capabilities to provide sufficient resources at a given time for a given traffic class are controlled at ingress node. Instead of assigning bandwidth to a particular link, provisioning of QoS requirements is done by assigning a virtual budget at ingress node for all relations between ingress and egress nodes using BBB NAC. BBB NAC in MPLS for LSP with a guaranteed bandwidth can be established by RSVP extension for LSP tunnels and then managing the right to network access can be made for each stream. Adaptation layer collects statistical data from the network layer and uses that information to generate a global view of the current state in the network. Detection of the current state in the proposed architecture is based on the utilization of the budgets allocated to the NACs, the frequency of blocked reservation (e.g. blocking frequency) and on the utilization of links' capacity. Adaptation process includes adjustment of the amount of available bandwidth for each traffic class separately and optimization of internal routing. For the link metric calculation we use gradient projection algorithm and delay of any (i) traffic class on each (j) link as metrics. But because of results achieved in previous research [5] which shows that the contribution of the link metric changes to the decrease of percentage of blocking traffic is very small and in order to keep our system stable, we switch off the link optimization loop during the simulation.

Adaptation layer follows its own internal strategy and optimization algorithms in order to adapt network performance to the traffic load variations. Adaptation layer has two regimes:

- Fuzzy controller regime, which is realized by means of fuzzy logic, based on the blocking frequency of traffic flows. The adaptation process is executed in

discrete cycles. Blocking frequency (FB) measured in particular cycle and difference in blocking frequency ((ΔFB) between two cycles are the input variables of triangle fuzzy membership function:

$$\mu(x)=\begin{cases} 0, & x \le b-a, \\ [x-(b-a)]/a, & b-a < x \le b, \\ -[x-(b+a)]/a, & b < x \le b+a, \\ 0, & x > b+a. \end{cases} \quad (2)$$

where the initial values of parameters are a = 2 b = (-6, -5, -4, -3 -2, -1, 0, 1, 2, 3, 4, 5, 6) to obtain 13 different values of the fuzzy variables. The initial design of the network based on the assessment of longer term traffic load and a relatively short period between the two adaptation cycles (15 minutes) significantly limit the value of FB and its fluctuations. This membership function and fuzzy rules, given in [16], are used for determination of the value of proportional coefficient (n_{ij}) to the bandwidth increment ΔB_i of i-th traffic class within j-th cycle. The bandwidth increment ΔB_i is given in advance for every traffic class. To adjust amount of allocated bandwidth G_{ij} of the i-th traffic class in j-th cycle to the actual traffic demand, adaptation algorithm changes allocated bandwidth of i-th traffic class in j-1st cycle with the value of n_{ij} ΔB_i in accordance with the following formula:

$$G_{ij} = G_{ij-1} + n_{ij}\Delta B_i \quad (3)$$

- Proportional-priority regime, which is based on minimum bandwidth allocated to the i-th traffic class (min_{pi}) and proportional-priority coefficient δ_{ij} of the i-th traffic class in j-th cycle, performs its functions in accordance with the following formula:

$$G_{ij} = \min_{pi} + \delta_{ij}(C - \sum_{i=1}^{n} \min_{pi}) \quad (4)$$

The criterion for switching between the two regimes is fulfilled when the sum of requested capacity of traffic classes B_i is bigger than network capacity (C):

$$\sum_{i=1}^{n} B_i > C \quad (5)$$

In the previous studies [17], behavior of the overall system, which performs its control functions automatically, autonomously and in an adaptive manner, is usually described by means of the following parameters:

- blocking frequency of traffic flows (FB),
- fairness of the allocation of resources to the traffic flows (P, J),
- utility function of network capacity (u_i)

$$u_i = \frac{\sum_{i=1}^{n} G_i}{C} \quad (6)$$

Blocking frequency of traffic flows (FB), we used as a key parameter for adaptation process, is defined as the total number of the rejected resource reservation (b_{miTN}) in all n classes of traffic within the determined time interval k. Measurement of rejected traffic flow is performed at the NAC any time new traffic flow ($c(f^{new}_{v,w})$ added to the existing traffic flows (c(f)) requests capacity which is higher than the available capacity (C(BBB)). of the given resources between nodes v and w. While the frequency of blocking can be defined as the maximum blocking probability or a relative ratio of blocked and offered traffic, this definition of blocking frequency, which treats all traffic classes simultaneously and only at the input node, is simple to measure and easy to calculate:

$$FB = \sum_{i=1}^{n} FB_i, \quad FB_i = \sum_{T_{N=1}}^{k} b_{miT_N} \quad (7)$$

$$b_{miT_N} = countif\left\{\left[c(f^{new}_{v,w}) + \sum c(f)\right] > C(BBB)\right\}$$

Fairness of resource allocation between traffic classes depends on the resource (bandwidth) allocation algorithm used during the process of adaptation to the actual traffic demands. Consideration of fairness makes sense only if the total amounts of requested resources exceed the capacity of available network resources. Otherwise, the problem boils down to utilization of network resources and to load balancing in order to assess the cost of depreciation and to even utilization of network resources. Fairness of the adaptive algorithm is the ability of the model to distribute the available resources in such a way that any traffic class does not give preference outside of the defined priority mechanism. The main goal of equitable allocation of resources assessment includes quantification of differences in distribution of resources between traffic classes by measuring variations in the ratio of allocated resources. We used equation 1 to evaluate fairness of proposed adaptation algorithm. We also compare the same fairness index achieved in the network architectures operating in adaptation mode and in the network architecture operating in non-adaptation mode to evaluate improvement in fairness.

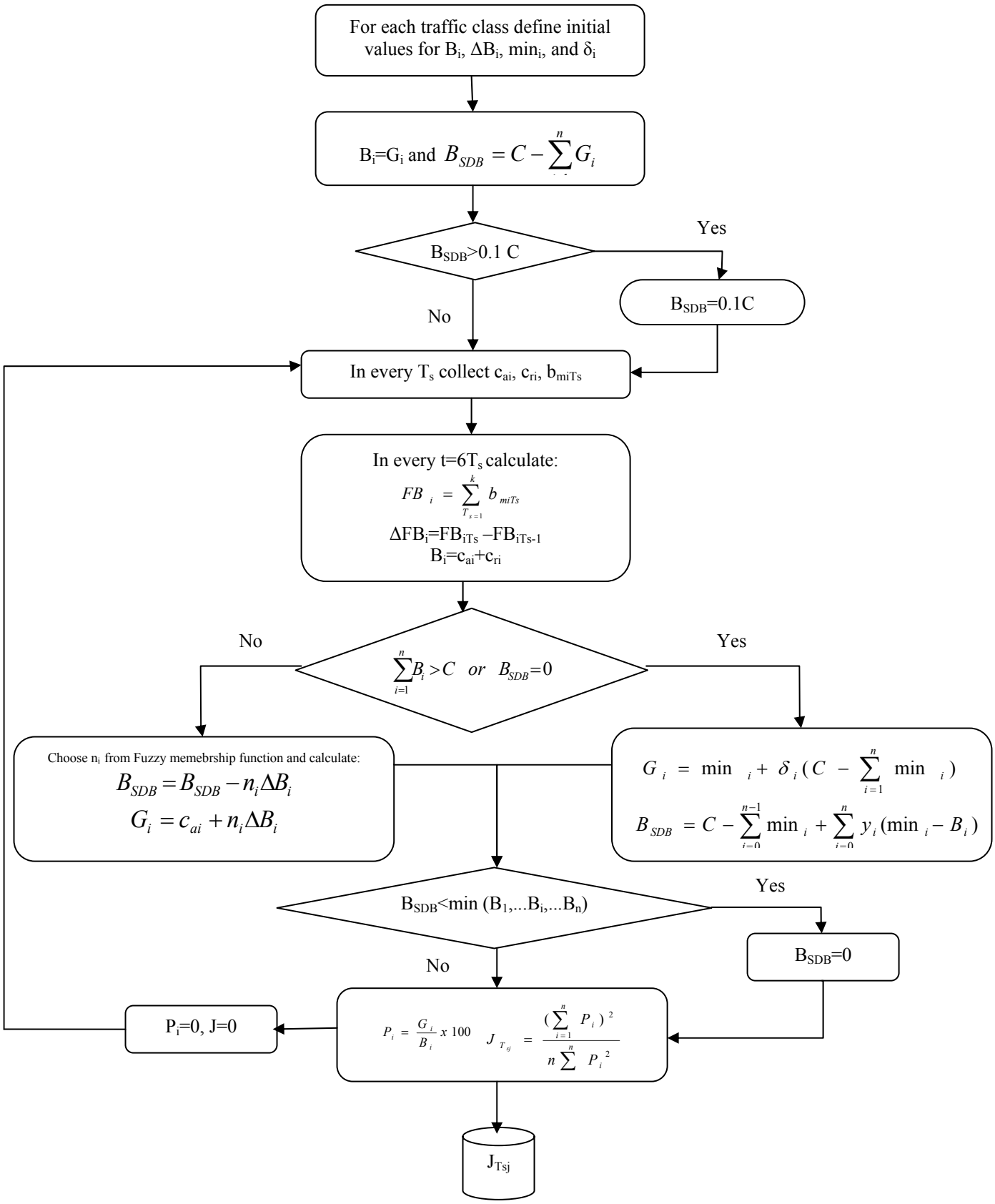


Figure 2. Flow chart of adaptation layer

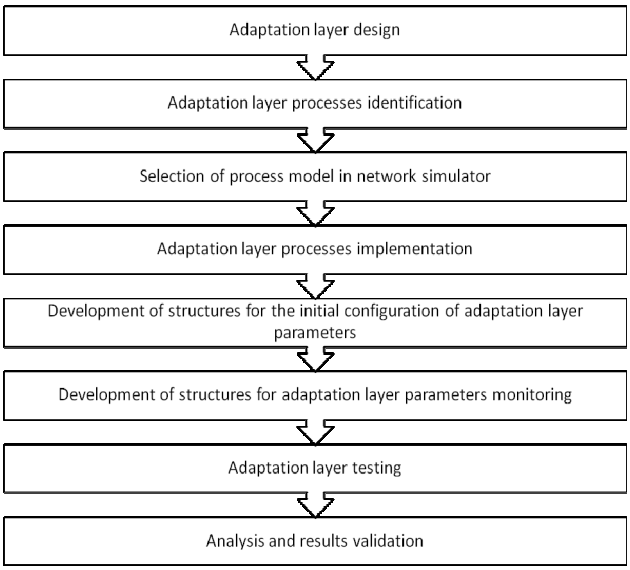


Figure 3. Development methodology of adaptation layer

Simulation model of network architecture we used consists of the adaptation and network layers. The adaptation layer performs a calculation of the new bandwidth budget for network admission controllers (NAC) and a calculation of new metrics on the links in order to adapt network performance to the actual traffic conditions. Measurements of blocking frequency (FB), accepted traffic (c_{ai}), rejected traffic (c_{ri}) and LSP load are performed in the regular time intervals at the ingress node in order to provide input data for operation of adaptation layer. The network layer autonomously executes forwarding functions of the packets and ensures QoS requirements using the capabilities of existing technologies and protocols. This type of network architecture represents the optimal set of available technologies with flexible topological landscape. Admission control functions, based on timely adjusted allocated bandwidth and load balancing functions by means of MPLS traffic engineering capabilities, are executed only at the ingress node of the autonomous network domain.

B. Looking for a Simulation Tool for Validation of Designed Model

The adaptation processes are capable of a continuous monitoring of the network parameters and performing their adaptation in accordance to the ever-changing traffic demands. Possible solution for such structure of adaptation process could be an active resource allocation based on the dynamic monitoring of their availability and of their sustainability to effectively transfer traffic.

The efficiency of such structures should be evaluated and an adequate simulation model which can adequately represent mechanisms and architecture for adaptive treatment of traffic demand is needed. Therefore, in this section we investigated possibilities of existing network simulators to support structure of adaptation algorithm we used in this paper. The analysis is based on a review of scientific studies in this field and documentation available for the network simulators.

In [8], a scheme for an adaptive bandwidth reservation in wireless multimedia networks was examined. For the purpose of validation of the proposed solution, necessary module for network simulator OPNET (Modeler 8.0) was developed. However, examination of the latest available version of the network simulator OPNET (Modeler 14.5) showed that these modules are not supported by the simulator manufacturers and as such is not included in the set of available modules. In [9], a solution for the adaptive bandwidth allocation in MPLS networks using a control with one-way feedback was given. The proposed solution was tested in a network simulator ns-2.27. But this solution is dedicated for particular problem and only in ns 2-27. Because of that it is inflexible for usage in general. Elwalid et al. [10] discussed adaptive traffic engineering in MPLS networks and their effort to develop their own simulator is the significant sign that there is a poor support for the dynamic adaptation structures in the available network simulators. Reviewing the documentation about the available network simulators we determined that none of those network simulators have built-in support for the dynamic adaptation structures. As the available simulators have no appropriate support for dynamic adaptation structures, and the same is necessary to test proposed structures, one of the objectives of this paper is to establish the methodological approach to development of adaptation layer in the network simulator. This methodology will be used for development of the adaptive layer modules within a chosen network simulator.

IV. DEVELOPMENT METHODOLOGY OF AN ADAPTATION LAYER WITHIN NETWORK SIMULATOR

In order to develop an adaptation layer which is independent of the adaptation mechanism of the used technology or of the network simulator, it is necessary to define a development methodology. We established a development methodology of an adaptation layer within network simulator which has eight steps shown in Fig. 3. Each of those steps will be explained in this section.

A. Design of the Adaptation Layer

In the section III we explained the basic functions and principles of our adaptation layer. The detailed design of adaptation layer with adaptation algorithm, input and output variables, decision criteria and pseudo code of the adaptation layer components are given in [16].

B. Adaptation Layer Processes Identification

For the purpose of execution of adaptation layer functions we identify next three processes:

- measurement of input variables,
- adaptation, and
- output parameters control.

The first process is deterministic and it is performed in regular time intervals (T_N). The task of this process is measurement of flow intensity of each traffic class and a

measurement of blocking frequency at the entry into the MPLS domain.

The process of adaptation is also deterministic, and it is performed in regular time intervals determined by the duration of discrete cycle of adaptation. The task of this process is to calculate a new budget based on input variables.

The last process is a stochastic process and its execution is caused by the decision results of adaptation process. The task of this process is to allocate a new budget to the network admission controller.

C. Selection of Process Model in Network Simulator

A number of network simulators are available today. Some of them, used in scientific researches, are ns-2/3, OPNET, OmNet++, GloMoSim, Nets, etc. Selection of adequate network simulator should be based on the characteristics of the simulators corresponding to the needs of adaptation layer developed as the target platform. We choose OPNET Modeler 14.5 as a proper network simulator considering next properties of the chosen simulator:

- simulation is based on the discrete network states (FSM-based approach),
- it supports traffic profile we intend to use during a simulation,
- it supports the network technologies and protocols we selected for a simulation model,
- it is suitable for prototype research such as this simulation model,
- it is easy to configure,
- it has relatively good documentation and support,
- it can be extended for adaptation layer (supports C-scripting language).

Since the proposed adaptive layer is a prototype of generic adaptation layer and as such does not exist in the selected network simulator, the whole adaptation layer should be developed based on the pseudo code given in [16], taking into account the constraints of simulator architecture. The architecture of the network simulator OPNET Modeler 14.5, extended with necessary modules for adaptation layer, is presented in Fig. 4.

Network simulator OPNET Modeler 14.5 is hierarchically organized. A network model is located at the highest level of hierarchy. The network model is composed of nodes and links connecting the nodes. Each node is defined by the node model (workstation, switch, router, server, etc.). Node model consists of processors that are described in process models. Process models are described in FSM's (Finite State Machine) and transfer functions written in C++ programming language. Transfer functions rely on the core functions of the simulator. The core simulator consists of pre-compiled libraries, whose source code is not available. The kernel is based on discrete event simulation.

Node at which network admission control functions are performed is the ingress LER of the MPLS domain. Bandwidth control at the entry of the network is ensured by establishing an explicit path with guaranteed bandwidth. The protocol that is responsible for setting up LSPs is RSVP-TE.

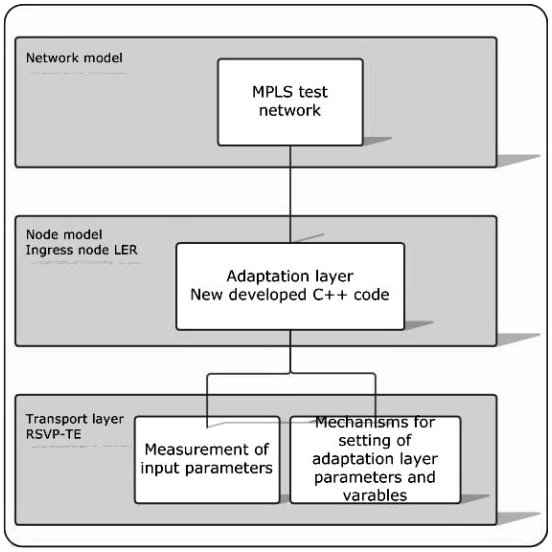


Figure 4. Hierarchical architecture of OPNET simulator

Therefore, the logical choice of process model within the simulator is RSVP process model. We used the network model which consists of five traffic sources nodes, the ingress edge router, four core MPLS routers, the egress edge router and five traffic destination nodes.

D. Adaptation Layer Processes Implementation

Ingress LER node model (Cisco 7600) consists of processors and queues associated with packet or statistic wires. We introduce the new statistical flows between the MAC (Media Access Control) queues and RSVP process model in order to take periodical measurements of the input variables, such as the mean intensity of flows, number of the rejected reservations, etc.

Process models in the OPNET network simulator are based on FSM. Passing from one state to another is initiated by different types of interruptions (packet arrival, arrival of new statistics value, user-defined stop, etc.).

During the transition stage different functions could be called. Besides the FSMs, the main components of a process model are the state variables, temporary variables, function block with headers, block functions, block for debugging and scheduling process block. Each process model also has attributes, interfaces, local and global statistics. Attributes and statistics can be promoted to a higher level, i.e. at the level of the node model.

E. Initial Configuration of Adaptation Layer Parameters

The initial parameters of adaptation layer, such as initial bandwidth per each traffic class (B_i), minimal bandwidth per traffic class ($min_{pi,}$), proportional-priority coefficient (δ_i), bandwidth increment (ΔB_i), are defined in [16]. Those initial parameters are subject to changes during the exploitation period (if the traffic environment changes dramatically) or during the simulation process (to be able to perform different simulation scenarios). For this purpose we need a proper structure within a simulator which offers changeability of the

initial configuration settings and changeable setting of its parameters. The development process of that structure has the following steps:

- definition of the adaptation layer attributes within the set of the existing process model attributes,
- promotion of the attributes from a process model level to the level of the node model,
- coding the input function in C++ to retrieve attributes when the simulation starts.

F. Monitoring of Adaptation Layer Parameters

In Sections I and III, we defined parameters which should be monitored such as blocking frequency (FB), fairness index (J), difference of blocking frequency (ΔFB) used as input variable for fuzzy membership function, etc. Those parameters should be measurable and monitored in order to qualify adaptation process execution and to use them as the input parameters to the adaptation layer. For this purpose we need a structure within simulator which offers a possibility to measure and monitor the values of the adaptation layer parameters. The development process of that structure has the following steps:

- definition of the local statistics in the process model,
- promotion of statistics on the level of the node model,
- coding of the function in C++ to record statistics.

G. Testing, Analysis and Result Validation

Those two steps of the development methodology are explained in Section V.

V. THE SIMULATION RESULTS

The simulation model created for testing purposes of adaptation layer is shown in Fig. 5 below. All nodes in the access part of the network are connected using 10 Gbps links, while the core routers are connected using 1 Gbps links. OSPF protocol is used as an IGP, and RSVP-TE protocol is used for establishment of LSPs. Between the LERs is 10 tunnels configured (two for each of five traffic classes). Traffic mapping at ingress LER is done using five different FEC based on the address of the traffic source. Traffic of each FEC is transmitted through two LSP (the first has an explicit route LER1-LSR1-LSR4-LER2, and the second-LER1 LSR1-LSR3-LER2).

For testing of adaptation layer, two scenarios are proposed:

- average load is 80% of network capacity,
- average load is 100% of network capacity.

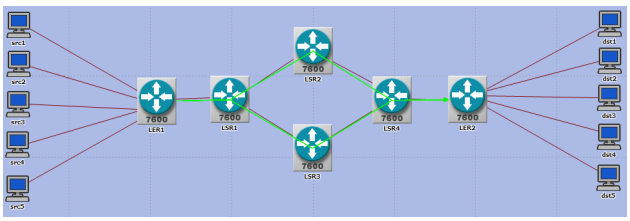


Figure 5. Simulation model for adaptation layer testing

The detailed dynamics of traffic demand of traffic classes for both scenarios, used during a simulation process, are given in the thesis [16]. In the first scenario the traffic generators are configured so that the average network load is 80%. The network capacity is 2 Gbps. The peak load of the network is about 2.5 Gbps. The initial values for those traffic classes, in the case that average network load is 80% of the network capacity, are given in Table 1 below.

TABLE I. INITIAL VALUES OF TRAFFIC SOURCES FOR 1ST SCENARIO

Traffic class	Initial BW kbps	Minimal BW kbps	Maximal BW kbps	BW incr. ΔBi	Proportional-priority coefficient δi
EF	8,270	5,990	14,000	125	$1.2\frac{B_{ij}}{\sum B_{ij}}$
AF1	465,110	319,760	700,000	10,000	$\frac{B_{ij}}{\sum B_{ij}}$
AF2	586,390	403,140	800,000	22,000	$0.9\frac{B_{ij}}{\sum B_{ij}}$
AF3	5,690	3,850	14,000	200	$0.9\frac{B_{ij}}{\sum B_{ij}}$
BE	431,310	286,528	700,000	6,000	$0.8\frac{B_{ij}}{\sum B_{ij}}$

During the simulation process we observe a distribution of requested bandwidth per each traffic class B_i and distribution of allocated bandwidth per each traffic class G_i in the same time window. We perform those observations in the adaptation mode of network architecture and in non-adaptation mode of the same network architecture in order to validate the accurate of the adaptation layer processes and to evaluate improvement in resource utilization as well as in QoS satisfaction of the requests of any traffic class.

We also observe values and distribution of fairness index (J) in both modes of network operation and values and distribution of ratio of the allocated resources to the requested bandwidth per each traffic class, in order to evaluate improvement of fairness in adaptation mode of network operation compared to the non-adaptation mode of operation. During the simulation process we take measures every 10 seconds and average those measurement values in time window of one minute, using those average values to calculate parameters which are needed for adaptation process of our adaptation algorithm.

By means of Figures 6 to 8 below, as a part of simulation results, we will show the outcomes of proposed adaptation algorithm, as well as of the extension of the OPNET structure. The whole scope of simulation results, from which we proof our entire concept, can be seen in [16].

From the Fig. 6, we can see that allocated bandwidth G for EF traffic class pretty well follows the required bandwidth B. This confirms that the adaptation layer functions properly and accurately. We can see the same results for other traffic classes and for average load of 80%.

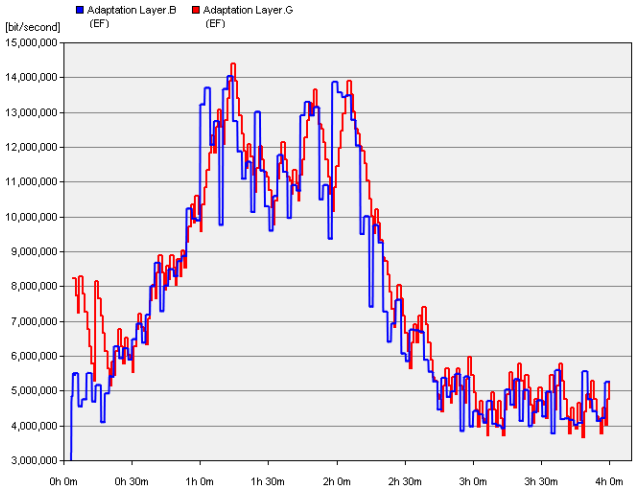


Figure 6. Requested and allocated bandwidth for EF traffic class

Fig. 7 shows us that introduction of adaptation layer improves the fairness of resource allocation in the network. During the non-adaptive mode, the ratio (P) of the allocated resources to the requested resources for EF class was unstable and goes up to 200%, while during the adaptive mode of network operation this percentage was stabilized and dropped to 100%, as is the preferred value for all traffic classes.

Fairness index (Fig. 8) is, in the adaptation mode of network operation, maintained above 0.96 with brief outages of up to 0.8, while the same index, in non-adaptation mode, is very unstable and drops up to 0.5.

Fig. 9 shows that the adaptation layer is stable structure. Blocking frequency stabilizes after a certain time on the value 10.

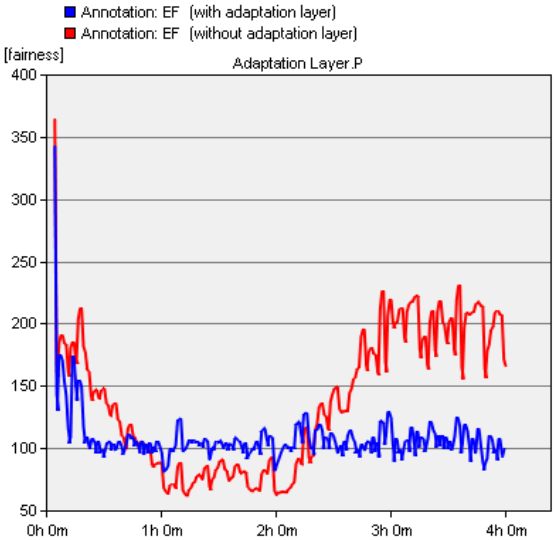


Figure 7. Ratio of the allocated resources for the EF traffic class (first scenario)

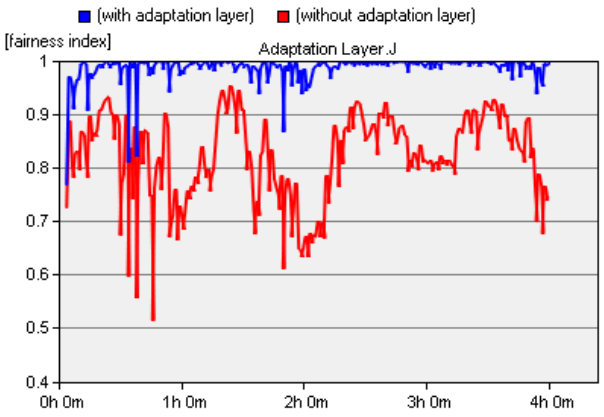


Figure 8. Fairness index (first scenario)

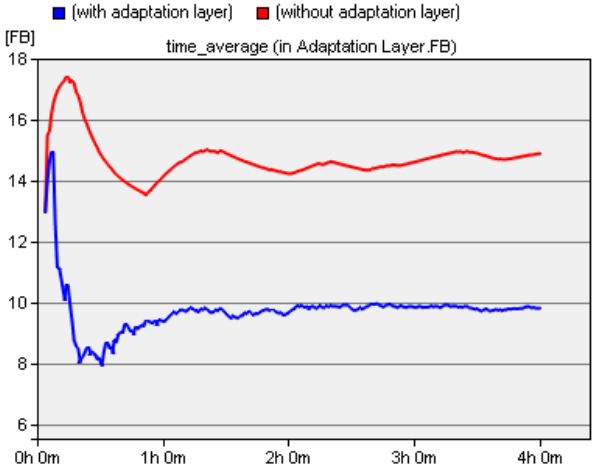


Figure 9. Blocking frequency (first scenario)

In the second scenario, the traffic generators are configured so that the average network load is 100%. The network capacity is 2 Gbps. The peak load of the network is about 3.125 Gbps.

To justify the results of simulation process we repeated the simulation and the same measurements and observations in the case that the average network load is 100% of the network capacity.

Fig. 10 shows that introduction of adaptation layer improves the fairness of resource allocation in the network even in conditions of heavy network load.

Fairness index (Fig. 11) is, in the adaptation mode of network operation, maintained above 0.8 with brief outages of up to 0.76, while the same index, in non-adaptation mode, is very unstable and drops up to 0.5.

Fig. 12 shows that the adaptation layer is stable structure - blocking frequency stabilizes after a certain time on the value 12.

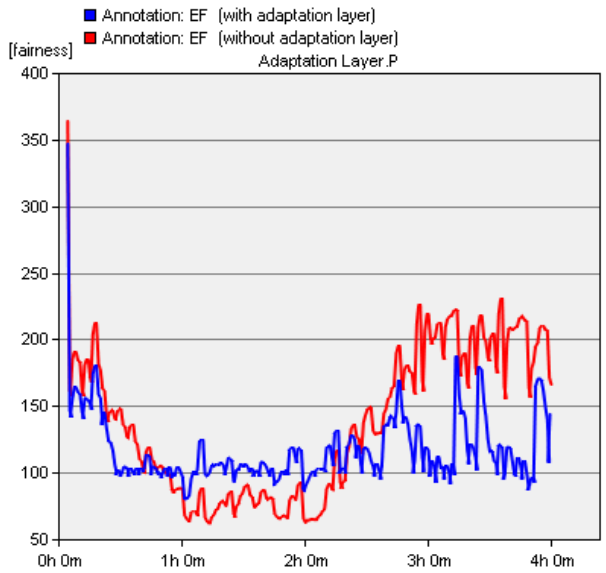


Figure 10. Ratio of the allocated resources for the EF traffic class (second scenario)

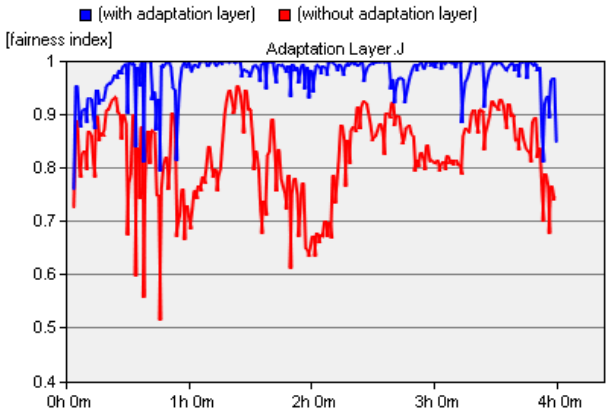


Figure 11. Fairness index (second scenario)

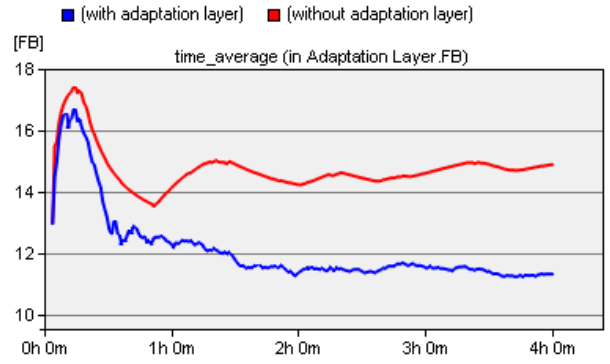


Figure 12. Blocking frequency (second scenario)

These results give us the proof of our hypothesis that with a proper adaptation layer we can improve the fairness of bandwidth allocation among different traffic classes under heavy network load and at the same time keep the required QoS conditions in the preferred boundaries. We can also conclude that the proposed adaptation algorithm behaves properly.

VI. CONCLUSION AND FUTURE WORK

The simulation results have shown that the proposed adaptation algorithm can significantly improve the fairness of bandwidth allocation among different traffic classes under a heavy traffic load in IP/MPLS networks, while keeping the required QoS conditions to any traffic class within the boundaries as preferred. The bandwidth allocated to any traffic class follows the required one, and in the case of a sufficient bandwidth, the QoS requests are guaranteed.

We see future work in researching the impact of different fuzzy algorithms and membership functions in the adaptation layer. It would also be interesting to analyze and discuss the required computational power and the protocol overhead in case of heavy network load, as well as ways of adaptation layer integration into existing network management systems.

REFERENCES

[1] T. Carsimamovic, E. Kaljic, and M. Hadzialic, "One Approach to Improve Bandwidth Allocation Fairness in IP/MPLS Networks Using Adaptive Treatment of the Traffic Demands," Proc. Seventh International Conference on Networking and Services, ICNS'11, May 22-27, 2011, Venice/Mestre, Italy, pp. 124-130, IARIA XPS Press, ISBN: 978-1-61208-133-5

[2] A. Stavdas, "Core and Metro Networks," John Wiley & Sons Ltd., 2010.

[3] C.A. Kamienski, "The Case of Inter-Domain Dynamic QoS based Service Negotiation in the Internet," Computer Communications, vol. 27, pp. 622-637, 2007.

[4] COST-257, Final Report, "Impact of New Services on the Architecture and Performance of Broadband Networks", 2000.

[5] C. Hoodgendoorn, "KING Research Project Overview," 2005.

[6] D. Godens, "Functional Architecture Definition and Top Level Design," Tequila Project, 2000.

[7] G. Hasslinger and J. Mende, "Measurement and Characteristics of Aggregated Traffic in Broadband Access Networks," in Proceedings of ITC 20, pp. 998-1010, Ottawa, 2007.

[8] X. Chen and Y. Fang, "An adaptive bandwidth reservation scheme in multimedia wireless networks," in IEEE Globecom, San Francisco, pp. 2830-2834, 2003.

[9] T. Yokoyama, K. Iida, H. Koga, and S. Yamaguchi, "Proposal for Adaptive Bandwidth Allocation Using One-Way Feedback Control for MPLS Networks," IEICE TRANS. COMMUN., vol. E90-B, no. 12, pp. 3530-3540, Dec. 2007.

[10] A. Elwalid, C. Jin, S. Low, and I. Widjaja, "MATE: MPLS Adaptive Traffic Engineering," in IEEE INFOCOM, Anchorage, pp. 1300-1309, 2001.

[11] W.-S. Lai, C.-H. Lin, J.-C. Liu, and H.-C. Huang, "Using Adaptive Bandwidth Allocation Approach to Defend DDoS Attacks," International Journal of Software Engineering and Its Applications, vol. 2, no. 4, pp. 61-72, Oct. 2008.

[12] A. Kamra, H. Saran, S. Sen, and R. Shorey, "Fair adaptive bandwidth allocation: a rate control based active queue management discipline," Computer Networks, no. 44, pp. 135-152, 2004.

[13] Y. Zheng, M. Lu, and Z. Feng, "Performance Evaluation of Adaptive AQM Algorithms in a Variable Bandwidth Network," IEICE TRANS. COMMUN., vol. E86-B, no. 6, pp. 2060-2067, 2003.

[14] R. Wang, M. Valla, M. Y. Sanadidi, and M. Gerla, "Using Adaptive Rate Estimation to Provide Enhanced and Robust Transport over Heterogeneous Networks," in Proceedings of the 10th IEEE

International Conference on Network Protocols, Washington, DC, pp. 206-215, 2002.

[15] R. Jain, "The Art of Computer System performance Analysis," Wiley, New York, 1991.

[16] T. Čaršimamović, "Selection of parameters for the adaptive treatment of traffic in next generation networks (NGN)," PhD Thesis, University of Sarajevo, Sarajevo, 2010.

[17] T. Engel, E. Nikolouzou, and A. Ricicato, "Analysis of Adaptive Resource Distribution Algorithms in the Framework of Dynamic DiffServ IP Network," Crete, 2001.

[18] U. Walter and M. Zitterbart, "Architecture of a Network Control Server for Autonomous and Effective Operation of Next Generation Network," Institut für Telematik, Karlsruhe, 2006.

[19] Runtong Zhang, Yannis A. Phillis, and Vassilis S. Kouikoglou, "Fuzzy Control of Queuing Systems," Springer, 2005.

[20] T. Shan and O.W.W. Yang, "Bandwidth Management for Supporting Differentiated-Service-Aware Traffic Engineering," IEEE Transactions on Parallel and Distributed Systems, Vol. 18, No. 9, 2007.

[21] M. Allalouf and Y. Shavit, "Centralized and Distributed Algorithm for Routing and Weighted Max-Min Fair Bandwidth Allocation," IEEE/ACM Transactions on networking, Vol. 16, No. 5, 2008.

[22] N. Blefari-Melazzi and M. Femminella, "A comparison of the utilization efficiency between a stateful and a stateless admission control in IP networks in a heterogeneous traffic case," Telecommunication Systems, KluwerAcademic Publishers, pages 231–258, March/April 2004.

[23] V. Elek, G. Karlsson, and R. Ronngren, "Admission control based on end-to-end measurements," In IEEE INFOCOM 2000, Israel, 2000.

[24] G. Bianchi and N. Blefari-Melazzi, "Admission Control over Assured Forwarding PHBs: A Way to ProvideService Accuracy in a DiffServ Framework," In *Proc. Of GLOBECOM*, San Antonio, Texas, Nov. 2001.

[25] E. Osborne and A. Simha, "Traffic Engineering with MPLS," Cisco Press, 2002.

The Mathematical Models for Different Start Video Broadcasting

Hathairat Ketmaneechairat
King Mongkut's University of Technology North Bangkok
Bangkok, Thailand
e-mail: hathairatk@kmutnb.ac.th

Phoemphun Oothongsap
King Mongkut's University of Technology North Bangkok
Bangkok, Thailand
e-mail: phoempn@kmutnb.ac.th

Anirach Mingkhwan
King Mongkut's University of Technology North Bangkok
Bangkok, Thailand
e-mail: anirach@ieee.org

***Abstract*—The different start video broadcasting is a new approach to improve the service of P2P video streaming applications. This approach allows unpunctual users to view broadcast programs from the beginning during server broadcast time. This paper proposes the non-cluster and cluster model for different start video broadcasting. The buffer management and mathematical model are proposed to estimate the performance of non-cluster and cluster model. These models are based on an application layer MESH network. These models are composed of five processes: peer join/leave, peer exchange information, peer selection, buffer organization and segment scheduling. The proposed models are simulated and verified by using NS-2. Moreover, the mathematical model is proposed to evaluate the performance metrics of server load, peer load, control message and buffer size. The results show that (i) the unpunctual users with different joining time are able to view the first video frame, (ii) the video server load is reduced drastically, (iii) the peer load is also reduced, (iv) the number of control messages exchanged between the nodes is reduced, and (v) the buffer size is constant. Moreover, the performance of cluster model is better than the non-cluster model.**

Keywords-Peer-to-Peer (P2P); IPTV; live video streamin; video on demand; Different start video broadcasting;

I. INTRODUCTION

Peer-To-Peer applications (P2P) have become very popular among Internet users. P2P technologies offer obvious advantages over content delivery network or content distribution network (CDN). P2P technologies improve system scalability with low implementation costs. P2P content delivery is an important technique for commercial systems such as IPTV. There are a lot of popular P2P file-sharing systems that support downloading such as Napster [2], Gnutella [3], Kazaa [4], BitTorrent [5], and eDonkey [6]. The main area of usage is P2P-based file sharing systems, like BitTorrent. Unlike traditional client-server architectures, peers in the network act as both client (leech) and server (seed). A peer not only downloads file from the network, but also uploads the downloaded file to other users in the network. Parts of the files are exchanged over direct connections between the peers. To enhance the system scalability and reduce the cost, several P2P video streaming applications have been published by using P2P technologies for the streaming of video and audio contents. P2P technologies are provided content distribution services for live video streaming and video-on-demand (VoD). CoolStreaming [7], PPStream [8], Sopcast [9], UUSee [10], and PPLive [11], are demonstrated by the huge popularity of P2P video streaming applications. These works [7, 8, 9, 10, 11] cause unpleasant problems. The first problem is that a far away connection increases network traffic and thus decreases network resource utilization. The second problem is a heavy tracker load. These problems can be delineated by using a hierarchical architecture as explained in [12]. In [12, 13, 14, 15], the cluster concepts for P2P systems are introduced.

For the live video streaming, live video contents are disseminated to all users in real-time. Hence, all users in the system can watch the same part of the stream at the same time. If users join the program later on, they will miss the beginning of the stream. The advantage of live video streaming is that the users can watch video stream almost immediately, without having to download an all file. The disadvantage of live video streaming is that the quality is limited by available bandwidth of each node and when the number of users is large, a server has limited bandwidth to support all users.

For the video-on-demand, the users can watch the video stream anywhere at any time. Multiple users may watch the same movie at the different playback times. The advantage of video-on-demand is higher quality. The drawback of video-on-demand is the users have to store the whole file. The result is a large buffer size.

Besides these two categories, there is another application that takes advantages of the live video streaming and the video-on-demand characteristics. This application is called the different start video broadcasting [1, 16, 17, 18]. The unpunctual users can watch the video stream from the beginning during server broadcast time. By mixing a Peer-to-Peer download concept with a live broadcasting one, a new node can find users who have the needed parts of the stream, and can use them as sources for download.

For the example, there is a game of FIFA World Cup which is start at 3 PM. and the game is end at 5 PM. A big amount of viewers will connect to the network and select the channel of the game. When the game starts at 3 PM, the viewers can load and view the game in real-time. After the game has started for 15 minutes, a new viewer decides to join the stream. The new viewer will have 2 choices: (i) view the game as the server broadcast or (ii) view the game from the beginning. With the first choice, this broadcast will feature live video streaming while the second choice will employ different start video broadcasting by mixture live streaming and video-on-demand features.

To support the different start video broadcasting applications, the non-cluster and cluster model are proposed. The non-cluster model is composed of five steps: peer join/leave, peer exchange information, peer selection, buffer organization and segment scheduling. The cluster model consists of eight steps: peer join, super node selection, backup-node selection, peer exchange information, peer selection, buffer organization, segment scheduling and leaving peer. The buffer management is organized as data buffer, buffer map and sliding window. The data buffer is divided into three parts: playback buffer (old chunks), display buffer (fresh chunks) and future buffer (future chunks). The Mesh-based architecture is used to exchange data between users. For the non-cluster model, there is only one tracker. For the cluster model, there are multiple trackers. The tracker is used to keep a list of all peers. In this paper, the mathematical model is used to determine starting delay, buffer size, peer list search and server load. The probability of download chunks and peer selection are proposed. The efficiency of the different start video broadcasting can be estimate. The proposed model is simulated and verified by NS-2. The results are affected by vary performance metrics, server load, peer load and the number of control messages. The performance of the non-cluster and the cluster model are compared.

The remainder of this paper is organized as follows: Section II describes related works, including the overview of P2P video streaming and P2P clustering. The non-cluster and cluster system design are illustrated in Section III and Section IV. The mathematical model is proposed in Section V. Section VI shows the experimental results. Finally, conclusion and future work are presented in Section VII.

II. RELATED WORK

This section describes the review of literatures regarding to the study. The challenge in P2P video streaming application is designed to be scalable and efficient for realtime streaming service on the Internet. The application supports live, video-on-demand (VoD), and both live and VoD streaming services. The application is aware of the format, the required bandwidth, the structure of the content delivery, and allows the content to be played smoothly during the delivery. There are many P2P video streaming application development of efficient and robust P2P content delivery network (CDN).

CoolStreaming or DONet [7] is a P2P live video streaming application for only one channel. There is no static streaming topology. Every node in the network can be a video-source which produces the content for neighbor nodes. Every node acts as an origin node keeping all of video segments. An original node is a single point of failure when it leaves. The departure nodes and dead nodes do not send any control messages. This may be the cause of packet loss.

PPStream [8] is a widely popular peer-to-peer (P2P) Internet TV application and it employs P2P video streaming network software that is similarly to BitTorrent. It can broadcast TV programs stably and smoothly to broadband users. Compared to traditional stream media, PPStream adopts P2P-streaming technology and supports full-scale visit with tens of thousands of users online. PPStream transfers data mainly using TCP and a few UDP packets. There are at least four types of nodes on the PPStream network: a unique channel list server, trackers, peer list servers and media chunk sharers. When the user launches the PPStream Client, the software will automatically connect to the channel list server to update the channel lists. After the user selects the channel to watch, the user has to wait several or tens of seconds for playing. During this period, the client will firstly request the peer list server for some other users watching the same channel. Each user is identified by its IP address and the listening port. The client will try to connect these peers to download data chunks.

Sopcast, UUSee and PPLive are channel-based systems, which provide a lot of different video streams on different channels. So each of the application networks need at least one media encoding server, where the video streams are created and stored, and a well known channel server where the clients can get information about available programs [10, 11, 19, 20, 21].

Sopcast [19] has a set of root servers, which maintains the information what peer is available for what channel. Sometimes also peer lists are exchanged between the peers. The most important difference of Sopcast is the usage of UDP as transport protocol [20]. This leads to fast packet transmission but also causes a lot of overhead for control. The usage of an external media player and a second buffer are very inefficient and lead to a huge start-up delay.

UUSee provides the videos by several dedicated streaming servers, so that there is no single point of failure and the video streaming quality especially the playback continuity is improved. The TCP protocol is used to communicate with all peers, exchange the buffer map, measure the round trip time (RTT) and estimate the throughput [10]. If a huge number of peers try to join the same channel in the network at short time duration, a noticeable influence on the network performance has been recognized.

PPLive [22] uses different methods to exchange information about the availability of channels or movies, chunks and pieces. A distributed hash table (DHT) is used to assign dedicated movies to dedicated trackers and to achieve a load balancing [23]. On the other side, PPLive tries to improve its playback quality at the expensive of the network architecture. A locality mechanism, which prefers physically near peers (e.g., of the same ISP) is implemented, but peers with high bandwidth are preferred. This may lead to a bad network performance also for other participants.

Most of these works [7, 8, 9, 10, 11, 19, 20, 21] have drawbacks related with low bandwidth utilization, high delay and a single point of failure. Thus to improve the performance of content distribution, the peers can be grouped in clusters. Many peers clustering approaches are proposed as the following:

The hierarchical architecture to group peers into clusters called CBT is proposed in [12]. The CBT has two novel algorithms: a peer joining algorithm and a super-peer selection algorithm. The proximity measurements of the RTT value and the TTL value between a pair of peer and

super-peer are used. The CBT system improves the performance and scalability, and can be used to build a large-scale BitTorrent-like P2P overlay network.

A novel super node overlay based on information exchange called SOBIE is proposed in [13]. The main contributions are to select the super nodes by considering the aggregation of not only the delay and distance, but also the information exchange frequency, exchange time and query similarity. The SOBIE is guaranteed the matching between the physical network and logical network. Moreover, the SOBIE has small-world characteristic to improve the efficiency and robustness.

The super node selection problem for Peer-to-Peer applications is presented in [14]. Three super nodes selection protocols for overlay P2P networks are proposed: SOLE, PoPCorn and H2O. An integrated approach to the super node selection problem built on strong graph theoretic foundations and guided by realistic applications, can benefit the Peer-to-Peer community through cross-fertilization of ideas and sharing of protocols.

An effective real-time Peer-to-Peer streaming system for the mobile environment is proposed in [15]. The peers are grouped into clusters according to their proximity using RTT values between peers as criteria for the cluster selection. The cluster leaders are using to help a service discovery server. The partial streams help to utilizing the upload capacity with finer granularity than just per one original stream. This is beneficial in mobile environments where bandwidth is scarce.

A cluster model for different start video broadcasting is proposed in [1]. The peers are grouped into cluster according to join time of each node and availability of first chunk. The cluster model consists of five processes: peer joining, super node selection, backup-node selection, download paths and leaving node process. The performance of the cluster model will be compared with the one of non-cluster model.

III. NON-CLUSTER SYSTEM DESIGN

This section introduces the concept of non-cluster system architecture and non-cluster system design. When a new node joins and wants to download chunks from the peers in the non-cluster model for different start video broadcasting, the tracker will send the random list of peers to a new node.

A. Non-Cluster-Based System Architecure

The non-cluster system architecture is composed of one server, only one tracker and normal nodes. The server is a node that provides all chunks of the live video stream. The global tracker is known by all nodes and maintains the list of all nodes in the network. The normal nodes are downloader and uploader. When a new node join in the network to used the video streaming, it will contact with tracker. The tracker will reply the random list of peers to a new node. The new node will exchange buffer map with the list of peers and selects peer neighbor to download the video streaming.
An overview of the non-cluster based system model is shown in Figure 1.

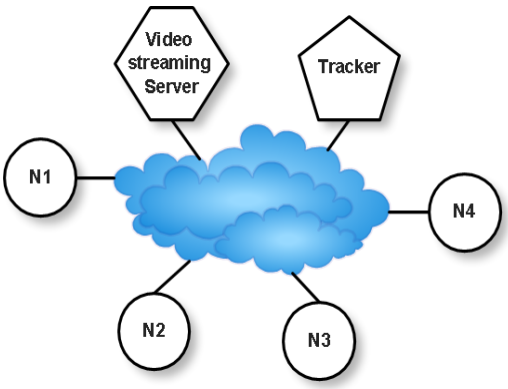


Figure 1. The non-cluster model for different start video broadcasting.

B. Non-Cluster-Based System Design

The method of the non-cluster system design consists of five stages as follow: (i) peer join/leave; (ii) peer exchange information; (iii) peer selection; (vi) buffer organization; (v) segment scheduling.

1) Peer join/leave: When a new peer joins the network and wants to use a video streaming, it sends a request message to the well known tracker server. This tracker maintains the list of all peers, which are currently streaming the same video channel, and the actual playback time of these nodes. After the tracker received the request message, the new peer is added in the peer list of the tracker and a list of nodes that are watching the same video stream at the same time is prepared. If this list is longer than a predefined value, a random subset is generated and sent to the new node. After receiving the list of peers, the new peer will exchange buffer maps with all peers in the list. With several buffer maps from different peers, the new peer can select partner peers and request video segments from them. As for leaving peers, they can leave the network at anytime.

2) Peer exchange information: A peer needs to know the availability of chunks on all of the known peers. This information is exchanged by BitTorrent-like buffer maps (BM) and HAVE or DONTHAVE messages. Since the buffer size of each node can be smaller than the whole video, parts of the video may be deleted. Hence, DONTHAVE messages had to be introduced additionally. They are used to inform neighbored nodes about the deletion of chunks. Initially, a new node creates a connection to all nodes that are reported by the tracker and requests BMs of these nodes. The replied BMs contain information about all chunks that are available on the sending node and are stored on the receiving node. When a node downloaded a new chunk or had to delete a chunk (due to restricted buffer size), it informs all connected nodes by sending a HAVE or a DONTHAVE message. The HAVE and DONTHAVE message should to be used because it may not be the cause of packet loss. Each of these messages contains the number of the new or deleted chunk. The messages are used by all receivers to update the BMs of the sending node.

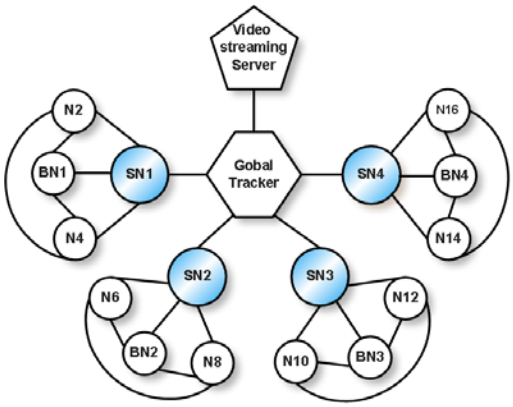


Figure 2. The cluster model for different start video broadcasting.

3) Peer selection: When the new peer knows the buffer maps of other peers, it has to select some of them to download parts of the stream. In opposite to BitTorrent, choking algorithms are not used. We figured out that these concepts leads to disruptions of the video stream. Instead, a node looks for all known peers that have the needed chunk and selects a source randomly.

4) Buffer organization: For the different start video broadcasting, each node has a buffer to store the video chunks. The length of buffer is smaller than video streaming file. The buffer of each node is organized into three parts: data buffer, buffer map and sliding window. The data buffer is used to store video frames. The buffer map is a bit vector representing the information of available segments on a node. Each node exchanges its buffer map with its partners periodically. From buffer map information, the peer will decide which partner nodes are used to fetch required segments. If there is more than one partner having the same segments, the peer node will randomly select the partners or select the partners with minimum delay or maximum bandwidth. Besides buffer map, each node needs to have a sliding window which is used to store a number of displaying segments. From this buffer organization, the video segments will be displayed continuously, and the starting delay of each node will be bounded. In this work, the circular buffer is used as buffer management. The buffer data is divided into three parts: playback buffer, displaying buffer, and future buffer as follows [1, 16, 17, 18].

- The playback buffer (old chunks) is used to buffer data stream for a certain period of time before playing the stream. The number of frames in playback buffer is calculated from the delay called playback delay between the sending and receiving peer. The playback delay between each peer is random since the mesh-based architecture is used. For simplicity, the playback delay is defined as a maximum delay bound in this group of users in this particular network. Thus, every peer will have the same playback delay.

- The displaying buffer (fresh chunks) is used to store data that will be viewed by users. This buffer is designed by using a sliding window. The frame in the beginning for buffer is the displaying frame and the next frame is the next frame in the window will be viewed in the next minutes.

- The future buffer (future chunks) is used to receive new frames. The new frames are received from other peers or partners by using sequential or rarest-first scheduling.

5) Segment scheduling: A peer client has chosen some peers to exchange segments, it is necessary to select these segments. This should be transmitted first and to select one peer client or more peer client as a source for this chunk, based on the knowledge of segment available in all its peers. The important information to calculate the chunk scheduling is the playback time. If a part of a video stream has already been played, the chunks which have to be played next time must be available in the local buffer. Two of the most important concepts are the sequential download and rarest-first download. The sequential download always chooses the chunk which is closest to the playback time. The rarest-first downloads the chunk which is available on the smallest number of peer clients and takes longer time to download. This is usually the newest segment. In this work, sequential download is used to download the segment of each peer. Several threads will be created to fill the different segment from each peer.

IV. CLUSTERING SYSTEM DESIGN

This section introduces the concept of cluster system architecture and cluster-based system design. When a new node joins and wants to download chunks from the peers in the cluster model for different start video broadcasting, the global tracker has to decide which cluster and super node will be joined.

A. Cluster-Based System Architecure

The cluster system is composed of server, global tracker (GT), super node (SN) or local tracker (LT), backup-node (BN) and normal nodes (NN). The server is a node that shares all chunks of a live video stream. The global tracker is known by all nodes and maintains the list of all super nodes. The super node acts as a local tracker keeping the list of all nodes in the cluster. All super nodes are connected with global tracker to synchronize the lists of all nodes in the cluster. The super node, normal node and backup-node are all downloader and uploader. The cluster means that the grouping of node partnerships according to their network proximity. The proximity is measured by using the join time of each node or availability of first chunk. The clustering is used to control the traffic streams within a P2P system and additionally helps to decrease the load of the server and global tracker. Based on the non-cluster model for the different start video broadcasting presented in [1, 16, 17, 18], the behavior and algorithms (peer exchange information, peer selection, buffer organization, segment scheduling) of the nodes are not changed but extended by a logical clustering mechanism. The clustering is realized by the separation of nodes, super nodes, backup-nodes and local trackers.

B. Cluster-Based System Design

The method of cluster system design consist of eight stages as follow: (i) peer join; (ii) super node selection; (iii) backup-node selection; (iv) leaving peer; (v) peer exchange

information; (vi) peer selection; (vii) buffer organization; (viii) segment scheduling.

1) Peer join: When a new node joins, it will contact with the global tracker to ask for the cluster and super node. Therefore, the peer joining algorithm has 2 important phases: connect to global tracker and connect to local tracker. For the first phase, all nodes know the address of the global tracker. When a new node contacts with the global tracker and asks for the first chunk. The global tracker will contact SN of each cluster to search for nodes having the first available chunk. The global tracker then selects the cluster which has the maximum number of nodes containing the first chunk. The new node gets the address of the local tracker (SN) and registers there. For the second phase, the new node contacts with the local tracker. The local tracker returns a random list of neighbor peers in the same cluster to the new node. The new node receives a random list of neighbor peers and sends the message to exchange buffer maps with neighbor peers. The new node selects neighbor peers to download chunks.

2) Super node selection: When the first node joins, the global tracker will set the first node to be a super node and local tracker of the first cluster. The cluster size is limit to C nodes. After the global tracker received joining message from a new node, it will check the cluster size to select appropriate cluster. The global tracker will verify the member size of the selected cluster. If the size of the selected cluster is less than C, the address of SN in that selected cluster will be send to the joining node. If the size of the selected cluster is full (equals C), a new cluster is created. The global tracker will split a node that have first chunk available in the old cluster to be a SN for a new cluster. If there is no cluster in the system, the first cluster is created and the first joining node will be a SN of the first cluster.

3) Backup-node selection: If the size of cluster is full (equals C), the BN will be selected from all normal nodes. The backup-node keeps a list of all peers in the cluster by contact with SN. When the SN leaves from the cluster, a BN will be a new SN. The BN will receive the list of all peers in the cluster from SN. The BN can be selected by three different methods as follow.

- Select the node joining the cluster after the first node. (the second joining node, 2nd)
- Select the node joining the middle of the group. (the $\frac{C^{th}}{2}$ node)
- Select the lasted node that joining the group. (the Cth node)

For the first method, all nodes in the cluster will have an equal chance to be a SN and BN. The drawback of this approach is a frequent SN and BN selection. The second method selects a new SN and BN not often and works well. The third method selects a new super node not often but may cause packet losses. In this paper, the second method is implemented in the simulation.

4) Leaving peer: If a node leaves from cluster, the local tracker will delete it from the list of peers. If the leaving node is a local tracker (SN), the backup-node will be a new local tracker (SN). If the last node leaves the cluster, the cluster is deleted. The local tracker always tells the global tracker about leaving nodes to synchronize the list of SN in the global tracker. The leaving-node process can be divided into three cases: the leaving of SN, BN and NN. For the first case, when the SN is leaving from the cluster, it sends flooding message to all nodes. The all nodes in the cluster will send keep-alive message to their SN. The SN sends keep-alive message to the global tracker. For the second case, the BN exchanges information periodically with the SN. If the BN leaves, it sends the message to inform the super node. For the last case, the NN can leave the network at anytime.

The peer exchange information, peer selection, buffer organization and segment scheduling of cluster model are similar to non-cluster model as described in Section III.

V. MATHEMATICAL MODEL

The mathematical model is used to determine starting delay, buffer size, peer list search and server load. The probability of peer download chunks and peer selection to download are proposed. The efficiency of the different start video broadcasting can be estimate.

A. Starting Delay Estimation

There are four types of delay: startup delay, starting delay, playback delay and delay. The startup delay, denoted T_{stu}, is the time that user supposes to wait until first frame arrives in the buffer. The playback delay, denoted T_{pb}, is the time that the user waited to fill chunk in the buffer until play smooth. The delay, denoted T_{join}, is an initial time according to the server time. Then the starting delay, denoted T_{sti}, is the total waiting time that user supposes to wait until displaying the first frame. The delay will be calculated as following.

The startup delay is depending on the number of neighbors that are used for content discovery and download, and the time used to exchange buffer maps. The startup delay can be considered in two cases, (1) only one available neighbor and (2) more than one neighbor.

1) Startup Delay (T_{stu}): The startup delay is mainly influenced by the network parameters like bandwidth and delay.

a) Only one neighbor

T_{stu} = Transmission delay + Propagation delay + Tracker time + Exchange buffer map time + Peer selection time.

$$T_{stu} = \sum_{i=1}^{n_{link}} \left(\frac{Packet\ \ Size}{Transmission\ \ Rate_i}\right) + \sum_{i=1}^{n_{link}} (Propagation\ Delay_i) \quad (1)$$

$$+\text{Tracker time} + \sum_{j=1}^{L} \sum_{i=1}^{n_{link}} \left(\frac{buffer\ map\ package\ size}{Transmission\ \ Rate_i}\right) + Peer\ selection\ time$$

Note: assume that queuing delay and processing delay is negligible.

n_{link} denoted the number of link from the sender to the receiver.

L denoted the number of peer in the list.

Transmission Rate denoted the number of bits per second.

Tracker time is the time that user contacts with tracker to receive the list of peers.

Exchange buffer map time is the time that the new comer exchanges buffer map with the peers in the received list.

Peer selection time is the waiting time that user is selected a peer to download the first chunk.

b) More than one neighbor

Since, there are several neighbors that there will be several paths. The first frame will receive from the neighbor that has the smallest delay. Thus P_i denoted a path is walk through nodes from a source (selected from the i^{th} neighbor) to a destination.

This case is calculated from the time that is necessary to download the first chunk. It depends on transmission delay and the propagation delay that is necessary for the sender to provide this packet.

$$T_{stu} = Min\left(Delay_{P_1}, Delay_{P_2}, \ldots, Delay_{P_k}\right) = Min\left(Delay_{P_i}\right) : i=1, 2, 3, \ldots, k \quad (2)$$

From Eq. (1), Let's $Delay_{P_i} = T_{stu}$

K is the number of neighbors.

P_i denoted path from the i^{th}neighbor.

2) Starting Delay (T_{sti}):

The starting delay is the total waiting time required to display the first frame. It relies on the join time, the playback time and the startup delay. The starting delay can be calculated as Eq. (3)

$$T_{sti} = Max(T_{join}, T_{pb}) + T_{stu} \quad (3)$$

B. Buffer Size Estimation

The buffer size can be estimated by using the join time or the release time. The unit of buffer size is defined as seconds because the waiting time for displaying video depends on the fill rate of node.

In case of few users, the delay has an impact on the different start video broadcasting viewing process, the peer joining late may not be able to view the whole stream. Then, the arrival condition is proposed. The arrival condition is used as maximum threshold of the delay of each peer. The arrival condition value is equal to Eq. (4).

$$\frac{N}{M} - T_{stu} \leq T_{pb} + T_{release} \quad (4)$$

Note: *N* is the total number of chunks.

M is the total number of nodes.

$T_{stu}$is the waiting time until the first chunk arrives in the buffer.

T_{pb} is the playback time.

$T_{release}$ is the release time.

In case of several users, the delay has no impact on the different start video broadcasting viewing process, the peer can view the whole video streaming.

If a join time is under the arrival condition, the new node can use the different start video broadcasting concept. If a join time is over the arrival condition, the new node will use the live-video streaming.

The buffer size estimation is calculated as following:

1) If the user joins under the arrival condition: The buffer size will be equated as in Eq. (5) and Eq. (6).

2) If the user joins over the arrival condition: The buffer size will be equated as in Eq. (6).

Note: The future time (T_{future}) is the maximum number of future chunks that receiver can receive in a unit of time, as shown in Eq. (7).

The release time, denoted $T_{release}$, is the time to wait until buffer is released.

In this paper, the buffer size is calculated from Eq. (6). The buffer size is constant.

$$Case\ 1 : Buffer\ size = Max\left(T_{join}, T_{pb}\right) + T_{future} \quad (5)$$

$$Case\ 2 : Buffer\ size = T_{release} + T_{pb} + T_{stu} + T_{future} \quad (6)$$

$$T_{future} = \frac{Link\ \ Speed\ \ (bandwidth)}{Fill\ \ Rate} \quad (7)$$

Since, the future time is used to receive new frames. The playback buffer and release buffer are constant. The fill rate usually refers to the number of chunks send to buffer per second.

C. Peer List Search

For the non-cluster model, the tracker uses the sequential search to find the list of peers in the peer list table. With the non-cluster model, the searching time of tracker is *O(M)* where *M* is the total number of nodes as shown in Eq. (8).

For cluster model, the global tracker employs the binary search to seek the proper super node in the peer list table. The local tracker employs the sequential search to seek the list of peers in the peer list table. The global tracker keeps a list of all peers and arrival time of each node. The cluster model reduces the searching time in the global tracker and local tracker. It groups peers into cluster according to joining time (arrival time) of each node. Let server starts broadcast at the time, $T = 0$ and ends at the time, $T = t$. The arrival time of each node will be referenced with the server broadcast time and sorted from minimum to maximum value. Each node will be grouped to each cluster according to its arrival time, as shown in Figure 3. Then, the number of clusters is in order of 2^X. The tracker will check the arrival time of each node and then assign the proper cluster to that particular node. Thus, the number of nodes in each cluster is a random number. Let *Ci* is the number of nodes in each cluster, *M* is the total number of nodes in the system and 2^X is the number of clusters. With this structure, the binary search is used to find the proper super node in GT. The searching time of GT is *O(log* 2^X*)*. The sequential search is

used to find the list of peers in LT. The searching time of LT is *O(C)*. Thus, the total searching time of cluster model is equal to is $O(\log 2^X + C)$ as shown in Eq. (9).

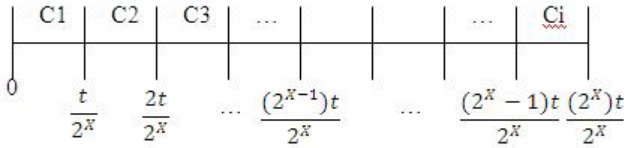


Figure 3. Even time line.

$$Non\text{-}Cluster\ Model = O(M) \quad (8)$$

$$Cluster\ Model = O(\log 2^X + C) \quad (9)$$

D. Server Load

For the non-cluster model for different start video broadcasting, the server load has two scenarios. In the first scenario, all users join at the same time like the server and start to download directly from the server only. In the second scenario, all users join over the arrival condition time, the all users will use the live-video streaming and download from server. From Eq. (10) is worst case of the non-cluster model. The server load is highest. When all peers can not download chunks from peer neighbors, the all peers will use the live-video streaming from server. The worst case of the non-cluster model is occur, when $peer_i$ join over 25 chunks of $peer_{i-1}$. (The value of 25 chunks is calculated from $T_{release} + T_{pb}$, this value is guaranteed that the new node can download available of first chunk). Assume that the video contents have N chunks. N is divisible by 25. The maximum number of peers is $\frac{N}{25}$. The maximum number of chunks that has been sent by server is equal to $\sum(25 + 50 + 75 + \cdots + N)$ or $25\sum(1 + 2 + 3 + \cdots + M)$. The best case of the non-cluster model is shown in Eq. (11). The server supports only one node in the non-cluster model.

For the cluster model for different start video broadcasting, the server load has two cases: worst case and best case. The worst case is the server supports equal to maximum number of super nodes follow from Eq. (12). The best case is the server supports only one super node follow from Eq. (13). Then, the server load can be calculated in Eq. (10), Eq. (11), Eq. (12) and Eq. (13).

Non-clustering with more than one node contacting server (Worst Case):

$$SL_{NC} = \frac{25xM}{2}(M + 1) \quad (10)$$

Non-clustering with only one node contacting server (Best Case):

$$SL_{NC} = 1\ x\ N \quad (11)$$

Clustering with more than one super node contacting server (Worst Case):

$$SL_C = SP\ x\ N \quad (12)$$

Clustering with only one super node contacting server (Best Case):

TABLE I. NOTATIONS AND DEFINITIONS OF DIFFERENT START VIDEO BROADCASTING

Notation	Definition
$\Omega = \{1,2, ...,N\}$	All currently available chunks across the entire network.
$N = \|\|\Omega\|\|$	The number of all currently available chunks.
K	Assume that all peers have exactly K neighbors.
M	The total number of nodes.
S	The total number of peers in the session.
l^{ext}	The buffer length of each node, $l^{ext} \leq N$ $l^{ext} = l^{old} + l^{fresh} + l^{future}$ or $l^{ext} = l_{xi} + l^{future}$
l^{old}	The number of old chunks in the buffer that waiting for overwritten, $l^{old} < l^{ext}$
l^{fresh}	The number of fresh chunks waiting for play in the buffer, $l^{fresh} < l^{ext}$
l^{future}	The number of chunks scheduled to download, they are not in the buffer, $l^{future} < l^{ext}$
l_{xi}	The length of the upload chunks, $l_{xi} = l_{xi}^{old} + l_{xi}^{fresh}$
x_i	The ith peer's conceptual view is downloader.
x_j	The first piece in the jth peer's conceptual view is $x_j + 1$ The jth peer's conceptual view is uploader. Assume x_j is uniformly distributed over [0, N]
$\Omega j = \{x_j + 1, x_j + 2, ..., x_j + l_{xj}\}$	All upload chunks in the jth peer's buffer, $\|\|\Omega j\|\| = l_{xi}$, $\Omega = \cup\ \Omega j$ $1 \leq j \leq S$
$\Omega' j = \{x_j + l_j + 1, x_j + l_j + 2, ..., x_j + l_{xj}^{ext}\}$	The chunks scheduled to download by the jth peer, $\|\|\Omega' j\|\| = l_{xj}^{future}$
η	The efficiency of different start video broadcasting,defined as the probability of a peer having at least one chunk interested to at least one of her neighbors. $\eta = 1 - (1 - \Pr\{\Omega j \cap \Omega' i \neq \emptyset\})^k$

$$SL_C = 1xN \quad (13)$$

Note: SL_{NC} denote the server load of non-cluster.
SL_C denote the server load of cluster.
M is the total number of nodes.
N is the total number of chunks.
SP is the number of super nodes.

E. Probability of peer selection to download

The probability of peer selection to download for the different start video broadcasting, the peer neighbor is important to get needed chunks. The downloader peer has to select the peer neighbors to download chunks. Therefore, a possible metric to evaluate the performance of peer selection for the different start video broadcasting is the Binomial probability distribution of a peer being in the downloading status as shown in Eq. (14) to Eq. (15).

$$Pr\{Selecting\ K\ peers\} = \binom{L}{K} P^K (1-P)^{L-K} \quad (14)$$

P = *Probability of the neighbor having at least one interested chunk*

$$= \sum_{i=1}^{l_{xi}} P(having\ i\ chunks)$$

$$Pr = \binom{L}{K} \left(\frac{l_{xi}(l_{xi}+1)}{2N} \right)^K \left(1 - \frac{l_{xi}(l_{xi}+1)}{2N} \right)^{L-K} \quad (15)$$

Note : L denotes the number of peers in the peer list.
K denotes the number of peer neighbors.
N denotes the total number of chunks.
l_{xi} denotes the length of upload chunks.
P denotes the probability of the neighbor having at least one interesting chunk.

F. Probability of download chunks

The probability of download chunks for the different start video broadcasting is from the concept in [24] that the new peer has to download chunks from the peer neighbors. When a new peer joins in the network, it will contact with the tracker. The tracker replies the list of peers. A new node will exchange buffer map with peers in the list. Then, it select peer and starts to download chunks. Since a new peer is downloading and uploading at the same time. This model will start the analysis by focusing on only two peers in the neighborhood first for peer i (x_i) and peer j (x_j). Peer j is downloading and uploading. Peer i is downloading only. Suppose at a given time t_0, there are totally N chunks available across the entire network, denoted by the set $\Omega = \{1, 2, ..., N\}$. The all chunks is uploading in the j^{th} peer's buffer as $\Omega j = \{x_j + 1, x_j + 2, ..., x_j + l_{xj}\}$, $j = 1, 2, ..., S$, where l_{xj} is the buffer length of upload chunk's peer j, l_{xj} is equal $l_{xj}^{old} + l_{xj}^{fresh}$. Obviously, each peer holds only a subset of Ω, namely, $0 \le x_j \le N - l_{xj}$, $j = 1, 2, ..., S$ and $\Omega = \Omega_1 \cup \Omega_2 \cup ... \cup \Omega_S$, where S is the total number of peers in the session. x_j is independent discrete random variable following a uniform distribution over [0, N].

The chunks are scheduled to download in the j^{th} peer's buffer as $\Omega' j = \{x_j + l_j + 1, x_j + l_j + 2, ..., x_j + l_{xj}^{ext}\}$, where l_{xj}^{ext} is equal l_{xj}^{future}. On the other hand, at *time* = t_0, for the i^{th} peer's buffer, the chunks scheduled to download are $\Omega' i = \{x_i + 1, x_i + 2, ..., x_i + l_{xi}^{ext}\}$. The i^{th} peer is interested in chunk's peer j if and only if $\Omega j \cap \Omega' i \neq \emptyset$. This condition can be simplified to $X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1$ as show in Figure 4. The list of all notations and definitions are shown in Table 1.

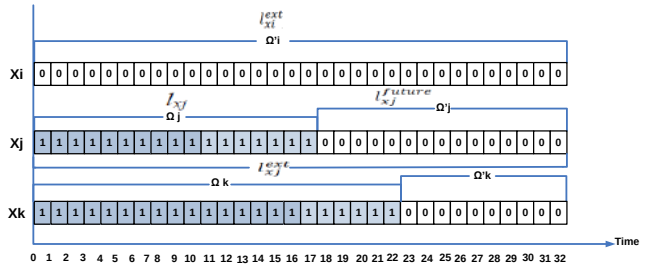


Figure 4. Download and upload chunks of each peer.

In the different start video broadcasting, the share resources are peers' buffer and bandwidth. All participating peers contribute their uploading bandwidth to increase the overall system throughput. Therefore, a possible metric to evaluate the performance of download chunks for different start video broadcasting is the probability of a peer being in the downloading status as shown in Eq. (16).

$$Pr = \{i \text{ is interested in at least one of } j\text{'s pieces}\}$$

$$Pr = \{\Omega j \cap \Omega' i \neq \emptyset\}$$

$$Pr = \{X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1\} \quad (16)$$

Then, this equation $Pr\{X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1\}$ will proof under the assumption that x_i and x_j are independent discrete random variables following a independent and identically distributed (i.i.d.) as shown in Eq. (17) and Eq. (18).

The probability of download chunks can be divided into two conditions: $N - l_{xi} \ge l_{xi}^{ext}$ and $N - l_{xi} < l_{xi}^{ext}$. The probability of download chunks is as follows (the whole proof will be shown in the appendix):

Condition 1 : when $N - l_{xi} \ge l_{xi}^{ext}$

$$\Pr\left(X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1\right)$$

$$= \sum_{X_j=0}^{l_{xi}^{ext}-1} Pr\left(X_j\right) Pr\left(0 \le X_i \le X_j - 1\right)$$

$$+ \sum_{X_j=l_{xi}^{ext}}^{N-l_{xi}} Pr\left(X_j\right) Pr\left(X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1\right)$$

$$= \sum_{X_j=1}^{l_{xi}^{ext}-1} \frac{1}{N - l_{xi} + 1} \times \frac{X_j - 1}{N - l_{xi} + 1}$$

$$+ \sum_{X_j=l_{xi}^{ext}}^{N-l_{xi}} \frac{1}{N - l_{xi} + 1} \times \frac{X_j - 1 - (X_j - l_{xi}^{ext} + 1)}{N - l_{xi} + 1}$$

$$= \frac{(l_{xi}^{ext} - 2)\left(2\,(N - l_{xi} + 1) - l_{xi}^{ext} - 1\right)}{2(N - l_{xi} + 1)^2} \quad (17)$$

Condition 2 : when $N - l_{xi} < l_{xi}^{ext}$

$$\Pr\left(X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1\right)$$

$$= \sum_{X_j=0}^{N-l_{xi}} Pr\left(X_j\right) Pr\left(0 \le X_i \le X_j - 1\right)$$

$$= \sum_{X_j=1}^{N-l_{xi}} \frac{1}{N-l_{xi}+1} \times \frac{X_j}{N-l_{xi}+1}$$
$$= \frac{N-l_{xi}}{2(N-l_{xi}+1)} \quad (18)$$

Combination two cases together,

$$Pr\ (X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1)$$
$$\begin{cases} = \frac{(l_{xi}^{ext}-2)\,(2\,(N-l_{xi}+1)-l_{xi}^{ext}-1)}{2(N-l_{xi}+1)^2} & ; \text{when } N\text{-}l_{xi} \ge l_{xi}^{ext} \\ = \frac{N-l_{xi}}{2(N-l_{xi}+1)} & ; \text{when } N\text{-}l_{xi} < l_{xi}^{ext} \end{cases}$$

G. Efficiency of the system

The system efficiency [24] of the different start video broadcasting can be considered from Eq. (17) and Eq. (18). The Eq. (18) is not selected for different start video broadcasting because the buffer size of each node is bigger than the number of chunks. For the different start video broadcasting, the buffer size of each node is less than the number of chunks. Therefore, the Eq. (17) is selected for different start video broadcasting. The Eq. (17) can be transformed to a more compact form. To calculate efficiency of the different start video broadcasting is shown in Eq. (19).

$$\text{Efficiency } (\eta) = \Pr\{\text{for arbitrary peer } i,\ i \text{ is in downloading status}\}$$
$$= 1\text{- Pr}\{\text{all } k \text{ neighbors of } i \text{ are not interested in } i\text{'s pieces}\}$$
$$= 1\text{- Pr}\{\text{peer } j, \text{ an arbitrary neighbor of } i, \text{ is not interested in } i\text{'s pieces}\}^k$$
$$= 1\text{- (1 - Pr}\{i \text{ is interested in at least one of } j\text{'s pieces}\})^k$$
$$= 1\text{- (1 - Pr}\{\ \Omega j \cap \Omega' i \neq \emptyset\ \})^k$$
$$= 1\text{– (1 – Pr}\{X_j - l_{xi}^{ext} + 1 \le X_i \le X_j - 1\})^k$$

$$\eta = 1\text{– (1– }[\frac{(l_{xi}^{ext}-2)\,(2\,(N-l_{xi}+1)-l_{xi}^{ext}-1)}{2(N-l_{xi}+1)^2}]^k)$$
$$\eta = 1\text{– (1– }[\frac{(l_{xi}^{ext}-2)(2\alpha-l_{xi}^{ext}-1)}{2\alpha^2}]^k) \quad (19)$$

where $\alpha = \frac{N-l_{xi}+1}{l_{xi}^{ext}} > 1$.

VI. Experimental Results

This section describes the experimental setups, the experimental results and evaluates the performance of non-cluster and cluster model for the different start video broadcasting. The discrete event simulator NS-2 [25, 26, 27] is used to create network topology. Network simulation (NS-2) is such an open source simulation tool that operates on the UNIX-based operating systems.

A. Simulation Setup

1) Non-Cluster Model

The experimental setup of the non-cluster model for different start video broadcasting creates one video media server. The server generates a live video streaming. The video stream length is 64 Mbytes, the size of each chunk is 64 Kbytes and the number of chunks is 1024. The video stream bitrate is 512 Kbps. Hence, each chunk represents exactly 1 second of video. The playback buffer size and release buffer size are set to 10 sec and 15 sec, respectively. The number of nodes varies from 50 to 200 nodes and the number of peer neighbors varies from 2 to 16. The delay of each physical network link equals to 2 ms and the bandwidth of each link is set to 2 Mbps. The video play rate is 1 chunk/1 sec. The random joining time of each node depends on the arrival condition from Eq. (4). The random joining time of each node is varied as 18.48, 8.24, 4.83 and 3.12, respectively. The startup delay (T_{stu}) is 3 sec (This value is estimated from simulation). The buffer size of each node equals 32 sec.

2) Cluster Model

The experimental setup of the cluster model for different start video broadcasting creates one video media server. The server generates a live video streaming. The video stream length is 64 Mbytes, the size of each chunk is 64 Kbytes and the number of chunk is 1024 chunks. The video stream bit rate is 512 Kbps. Hence, each chunk represents exactly 1 second of video. The playback buffer size and release buffer size are set to 10 sec and 15 sec, respectively. The number of nodes equals to 200 nodes and the number of neighbors varies from 2 to 16. The number of clusters is varied as 2 (100 nodes), 4 (50 nodes), 8 (25 nodes) and 16 (13 Nodes), respectively. The delay of each physical network link equals to 2 ms and the bandwidth of each link is set to 2 Mbps. The video play rate is 1 chunk/1 sec. The random joining time of each node depends on the arrival condition from Eq. (4). The random joining time of each node equals to 2.12. The startup delay (T_{stu}) is 3 sec (This value is estimated from simulation). The buffer size of each node equals 32 sec.

B. Simulation Results

1) Non-Cluster Model

The non-cluster model is implemented to evaluate the performance of different start video broadcasting. The performance is considered from varies the number of nodes and the number of peer neighbors. The performance metrics of the non-cluster model for different start video broadcasting is follows:

a) Server load

The result of chunks that are downloaded directly from the server is shown in Figure 5. The chunks download from server is plotted for varies number of nodes and number of neighbors. The number of chunks that are downloaded from the server represents the server load. The x-axis is the number of nodes and the y-axis is the number of chunks downloaded from the server. The result shows that the number of nodes and the number of peer neighbors have impact on sever load. If the number of nodes and the number of neighbors are increased, the server load is reduced.

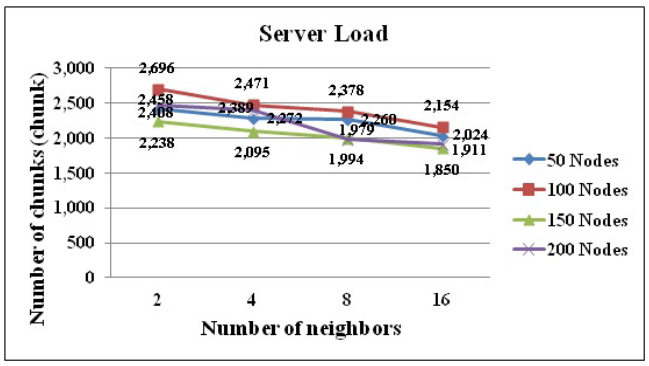


Figure 5. The server load of non-cluster model for different start video broadcating.

b) Peer load

The result of the average number of chunks downloaded from one peer for non-cluster model is shown in Figure 6. The chunks downloaded from one peer is plotted for varies the number of nodes and the number of peer neighbors. The number of chunks that are downloaded from a peer represents the peer load. The x-axis is the number of nodes and the y-axis represents the number of chunks downloaded from one peer. The result shows that the number of nodes has impact on the peer load but the number of peer neighbors has no impact on the peer load. If the number of nodes is increased, the peer load is increased. The peer load is better performance because one peer serves chunks less than 1,024 chunks. (The peer load of system is increased follow the number of nodes is increased)

c) The number of control message

The control messages are used for communication between nodes. TCP is used as transport protocol for control messages. UDP is used for the exchange of video chunks. The result of control message for non-cluster model is shown in Figure 7. The number of TCP control messages of each network used to exchange information between nodes. The x-axis represents the number of nodes and the y-axis represents the number of control messages between all nodes. The result shows that the number of nodes and the number of neighbors have impact on the number of control messages. If the number of nodes and the number of neighbors are increased, the number of control messages grows accordingly.

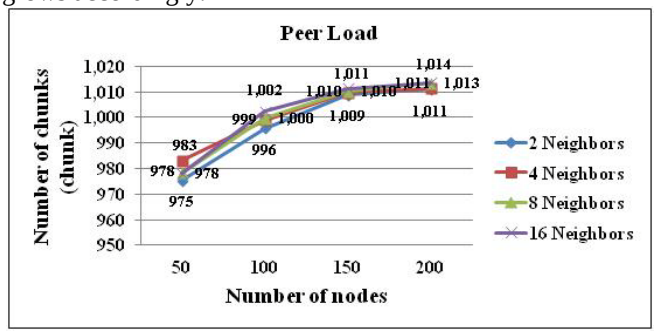


Figure 6. The peer load of non-cluster model for different start video broadcating.

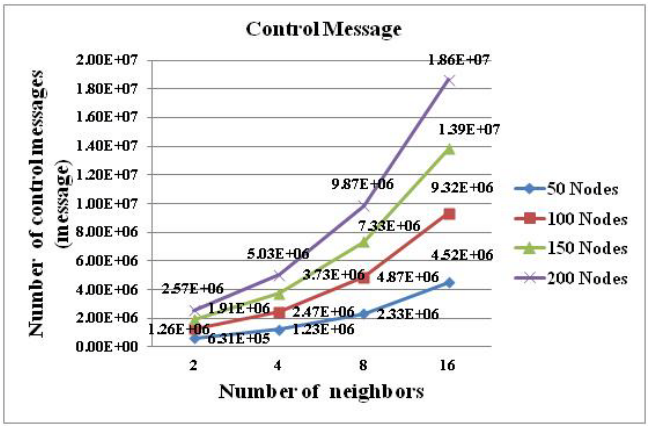


Figure 7. The control message of non-cluster model for different start video broadcating.

2) Cluster Model

The cluster model is implemented to evaluate the performance of different start video broadcasting. The performance is considered from varies the number of clusters and the number of peer neighbors. The performance metrics of the cluster model for different start video broadcasting is follows:

a) Server load

The number of chunks downloaded directly from the server in a cluster model is shown in Figure 8. The chunks downloaded from a server are plotted for various the numbers of clusters and the number of neighbors. The chunks downloaded from server have effect for server load. The x-axis is the number of clusters and the y-axis is the number of chunks downloaded from server. The result shows that the number of clusters and the number of neighbors have impact on sever load. If the number of clusters and the number of neighbors are increased, the server load decreased equal to constant. It means that server serves only one peer. The other peers can download chunks from neighbor peers in the cluster.

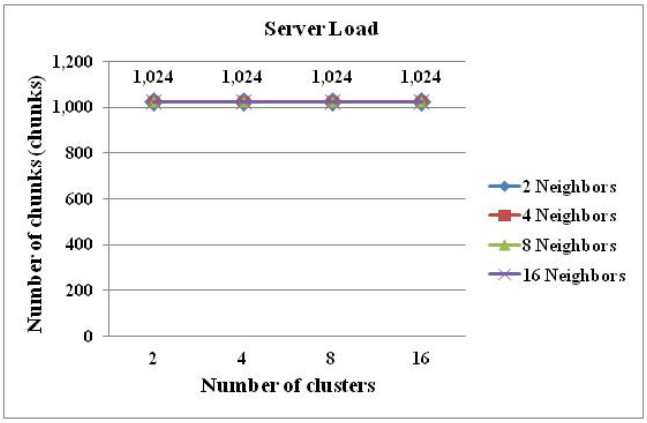


Figure 8. The server load of cluster model for different start video broadcating.

b) Peer load

The result of the number of chunks downloaded from one peer for non-cluster model is shown in Figure 9, the chunks downloaded from one peer is plotted for varies the number of clusters and the number of peer neighbors. The number of chunks that are downloaded from a peer represents the peer load. The x-axis is the number of clusters and the y-axis represents the number of chunks downloaded from one peer. The result shows that the number of clusters and the number of peer neighbors have impact on the load of one peer. If the number of clusters and the number of peer neighbors are increased, the peer load decreased equal to constant. The peer load is better performance because one peer serves chunks less than 1,024 chunks. (The peer load of system is increased follow the number of nodes is increased)

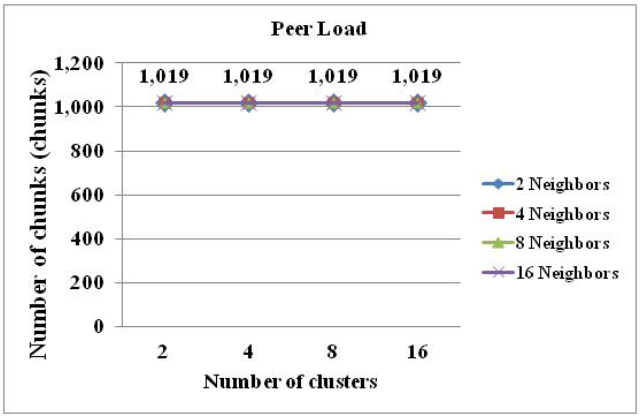


Figure 9. The peer load of cluster model for different start video broadcating.

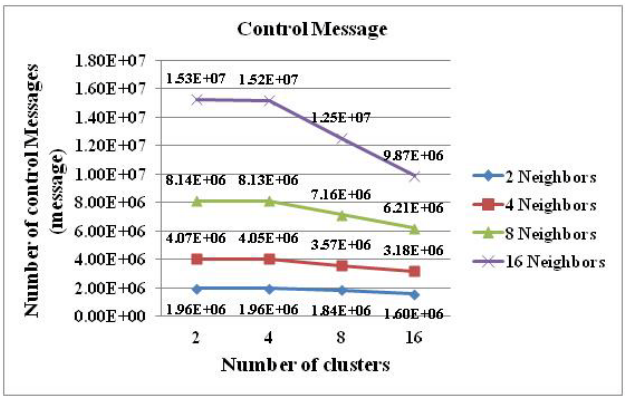


Figure 10. The control message of cluster model for different start video broadcating.

c) The number of control message

The control messages are used to communication between nodes. TCP is used as transport protocol for control messages. UDP is used for the exchange of video chunks. The result of control message for cluster model is shown in Figure 10. The number of TCP control messages of each network used to exchange information between nodes. The x-axis represents the number of clusters and the y-axis represents the number of control messages. The result shows that the number of clusters and the number of neighbors have impact with the control message. If the number of clusters is increased, the control message is decreased. If the number of neighbors is increased, the control message is increased.

VII. CONCLUSION AND FUTURE WORK

This paper proposed the non-cluster model and cluster model for different start broadcasting to improve the service of P2P video streaming. The different start video broadcasting used both characteristics of the live video streaming and the video-on-demand, created on an application layer Mesh network and the BitTorrent concept. The system design of non-cluster model and cluster model for different start video broadcasting is proposed. The non-cluster model for different start video broadcasting consists of five stages: (i) peer join/leave; (ii) peer exchange information; (iii) peer selection; (vi) buffer organization; (v) segment scheduling. The cluster model for different start video broadcasting consists of eight stages: (i) peer join; (ii) super node selection; (iii) backup-node selection; (iv) peer exchange information; (v) peer selection; (vi) buffer organization; (vii) segment scheduling; and (viii) leaving peer. The system model is evaluated using the mathematical model such as starting delay, buffer size, peer list search, server load, probability of peer selection to download, probability of download chunks and system efficiency. The performance of the non-cluster model and the cluster model for different start video broadcasting are compared. As a result, the performance of the cluster model is better than non-cluster model. The cluster model can improve the utilization of available bandwidths for upload and download, reduces the server load, reduces peer load and reduces the number of control messages. The unpunctual users can view the video stream from the beginning during server broadcast time. Furthermore, the tracker traffic, dynamic node joins and leaving nodes have to be implemented. The backup-node selection method has to be reconsidered. The different start video broadcasting should be investigated on mobile network environment.

REFERENCES

[1] H. Ketmaneechairat, P. Oothongsap and A. Mingkhwan. "Peer Clustering System for Different Start Video Broadcasting." Proceeding of The sixth International Conference on Digital Telecommunications (ICDT 2011) : 116-122.

[2] Napster [online], available at http://www.napster.com [Accessed: Aug. 10, 2010]

[3] Gnutella [online], available at http://www.gnutella.com [Accessed: Aug. 10, 2010]

[4] Kazaa [online], available at http://www.kazaa.com [Accessed: Aug. 10, 2010]

[5] BitTorrent [online], available at http://www.bittorrent.com [Accessed: Aug. 10, 2010]

[6] eDonkey[online], available at http://en.wikipedia.org/wiki/EDonkey 2000 [Accessed: Aug. 10, 2010]

[7] X. Zhang, et al. "CoolStreaming/DONet: A data-driven Overlay Network for Efficient Live media streaming." IEEE Computer and Communications Societies. (2005) : 2102-2111.

[8] X. Su and L. Chang. "A Measurement Study of PPStream." IEEE Communications and Networking in China. (2008)

[9] S. Tang, et al. "Topology dynamics in a P2PTV network." Proceeding of the 8th International IFIP-TC 6 Networking Conference. (2009) : 326-337.

[10] C. WU, B. Li and S. Zhao. "Exploring Large-Scale Peer-to-Peer Live Streaming Topologie." ACM Transaction on Multimedia Computing, Communications and Applications. (2008) : 1-23.

[11] Y. Huang, et al. "Challenges, Design and Analysis of a Large-scale P2P-VoD System." Proceeding of the ACM SIGCOMM Conference on Data communication. (2008) : 375-388.

[12] J. Yu and M. Li. "CBT: A Proximity-Aware Peer Clustering System in Large-Scale BitTorrent-like Peer-to-Peer Networks." Journal of Computer Communication's Special issue on Foundation of Peer-to-Peer Computing. (2008) : 591-602.

[13] Z. Chen, et al. "SOBIE: A novel super-node P2P overlay based on information exchange." Journal of Computers. (2009) : 853-861.

[14] V. Lo, et al. "Scalable supernode selection in peer-to-peer overlay networks." Proceeding of the 2nd International Workshop on Hot Topics in Peer-to-Peer Systems. (2005) : 18-25.

[15] J. Peltotalo, et al. "An RTSP-based Mobile P2P Streaming System." Journal of Digital Multimedia Broadcasting. (2010) : 1-15.

[16] H. Ketmaneechairat, P. Oothongsap and A. Mingkhwan. "Smart Buffer Management for Different Start Video Broadcasting." Proceeding of the 2nd International Conference on Interaction Sciences : Information Technology, Culture and Human. (2009) : 615-619.

[17] H. Ketmaneechairat, P. Oothongsap and A. Mingkhwan. "Buffer Size Estimation for Different Start Video Broadcasting." Electrical Engineering/Electronics Computer Telecommunications and Information Technology ECTI-CON'10. (2010) : 924-928.

[18] H. Ketmaneechairat, P. Oothongsap and A. Mingkhwan. "Communication Size and Load Performance for Different Start Video Broadcasting." Proceeding of the PGNET'10. (2010).

[19] A. Sentinelli, et al. "Will IPTV ride the peer-to-peer stream?." Proceeding of the IEEE Communications Magazine. (2007) : 86-92.

[20] B. Fallica, et al. "On the Quality of Experience of SopCast." Proceeding of the Next Generation Mobile Applications, Services and Technologies. (2008) : 501-506.

[21] X. Hei, et al. "Insights into PPLive: A measurement study of a large-scale P2P IPTV system." Proceeding of the IPTV Workshop, International World Wide Web Conference. (2006).

[22] C. Wu, B. Li and S. Zhao. "Multi-channel Live P2P Streaming: Refocusing on Servers." Proceeding of the IEEE INFOCOM'08. (2008) : 1355-1363.

[23] L. Vu, et al. "Mapping the PPLive network: Studying the impacts of media streaming on P2P overlays." Proceeding of the UIUC Technical Report. (2006).

[24] H. Liu and G. Riley. "How Efficient Peer-to-Peer Video Streaming Could Be?," Proceeding of the Consumer Communications and Networking Conference. (2009) : 1-5.

[25] The Network Simulator (NS-2). [serial online] 2006. [cited 2007 Aug 17]. Available from : URL http://www.isi.edu/nsam/ns/

[26] T. Issariyakul and E. Hossain. "Introduction to Network Simulator NS2." Journal of Springer. (2009) : 1-438.

[27] K. Eger, et al. "Efficient Simulation of Large-Scale P2P Networks: Packet-level vs. Flow-level Simulations." Proceeding of the 2nd Workshop on the Use of P2P, GRID and Agents for the Development of Content Networks. (2007).

APPENDIX

The probability of peer selection to download is proof as follows:

$$Pr\{\text{Selecting K peers}\} = \binom{L}{K} P^K (1-P)^{L-K}$$

P = *Probability of the neighbor having at least one interested chunk*

$$= \sum_{i=1}^{l_{xi}} P(\text{having } i \text{ chunks})$$

$$= \sum_{i=1}^{l_{xi}} \frac{i}{N}$$

$$= \frac{1}{N} + \frac{2}{N} + \frac{3}{N} + \ldots + \frac{l_{xi}}{N}$$

$$= \frac{l_{xi}(l_{xi}+1)}{2N}$$

$$Pr = \binom{L}{K} \left(\frac{l_{xi}(l_{xi}+1)}{2N}\right)^K \left(1 - \frac{l_{xi}(l_{xi}+1)}{2N}\right)^{L-K}$$

The probability of download chunks is proof as follows:

Condition 1 : when $N-l_{xi} \geq l_{xi}^{ext}$

$$\Pr\left(X_j - l_{xi}^{ext} + 1 \leq X_i \leq X_j - 1\right)$$

$$= \sum_{X_j=0}^{l_{xi}^{ext}-1} Pr\left(X_j\right) Pr\left(0 \leq X_i \leq X_j - 1\right)$$

$$+ \sum_{X_j=l_{xi}^{ext}}^{N-l_{xi}} Pr\left(X_j\right) Pr\left(X_j - l_{xi}^{ext} + 1 \leq X_i \leq X_j - 1\right)$$

$$= \sum_{X_j=1}^{l_{xi}^{ext}-1} \frac{1}{N-l_{xi}+1} \times \frac{X_j - 1}{N-l_{xi}+1}$$

$$+ \sum_{X_j=l_{xi}^{ext}}^{N-l_{xi}} \frac{1}{N-l_{xi}+1} \times \frac{X_j - 1 - (X_j - l_{xi}^{ext} + 1)}{N-l_{xi}+1}$$

$$= \frac{1}{(N-l_{xi}+1)^2} \times \left[\sum_{X_j=1}^{l_{xi}^{ext}-1} X_j - \sum_{X_j=1}^{l_{xi}^{ext}-1} 1\right]$$

$$+ \frac{1}{(N-l_{xi}+1)^2} \sum_{X_j=l_{xi}^{ext}}^{N-l_{xi}} (l_{xi}^{ext} - 2)$$

$$= \frac{1}{(N-l_{xi}+1)^2} \times \left[\frac{(l_{xi}^{ext}-1)(l_{xi}^{ext}-1+1)}{2} - \frac{2(l_{xi}^{ext}-1-1+1)}{2}\right]$$

$$+ \frac{l_{xi}^{ext}-2}{(N-l_{xi}+1)^2} \times \sum_{X_j=l_{xi}^{ext}}^{N-l_{xi}} 1$$

$$= \frac{1}{(N-l_{xi}+1)^2} \times \left[\frac{(l_{xi}^{ext}-1)(l_{xi}^{ext}-2)}{2}\right]$$

$$+ \frac{l_{xi}^{ext}-2}{(N-l_{xi}+1)^2} \times \left[\frac{2(N-l_{xi}-l_{xi}^{ext}+1)}{2}\right]$$

$$= \left[\frac{(l_{xi}^{ext}-1)(l_i^{ext}-2)}{2(N-l_{xi}+1)^2}\right] + \frac{(l_{xi}^{ext}-2) \times (2N - 2l_{xi} - 2l_{xi}^{ext} + 2)}{2(N-l_{xi}+1)^2}$$

$$= \frac{(l_{xi}^{ext}-2) \times (l_{xi}^{ext} - 1 + 2N - 2l_{xi} - 2l_{xi}^{ext} + 2)}{2(N-l_{xi}+1)^2}$$

$$= \frac{(l_{xi}^{ext}-2)\,(2\,(N - l_{xi} + 1) - l_{xi}^{ext} - 1)}{2(N-l_{xi}+1)^2}$$

Condition 2 : when $N-l_{xi} < l_{xi}^{ext}$

$$\Pr\left(X_j - l_{xi}^{ext} + 1 \leq X_i \leq X_j - 1\right)$$

$$= \sum_{X_j=0}^{N-l_{xi}} Pr\left(X_j\right) Pr\left(0 \leq X_i \leq X_j - 1\right)$$

$$= \sum_{X_j=1}^{N-l_{xi}} \frac{1}{N-l_{xi}+1} \times \frac{X_j}{N-l_{xi}+1}$$

$$= \frac{1}{(N-l_{xi}+1)^2} \sum_{X_j=1}^{N-l_{xi}} X_j$$

$$= \frac{1}{(N-l_{xi}+1)^2} \times \frac{(N-l_{xi})(N-l_{xi}+1)}{2}$$

$$= \frac{N-l_{xi}}{2(N-l_{xi}+1)}$$

A Resource Management Strategy Based on the Available Bandwidth Estimation to Support VoIP across Ad hoc IEEE 802.11 Networks

Janusz Romanik
Radiocommunications Department
Military Communications Institute
Zegrze, Poland
j.romanik@wil.waw.pl

Piotr Gajewski, Jacek Jarmakiewicz
Faculty of Electronics
Military University of Technology
Warsaw, Poland
{pgajewski, jjarmakiewicz}@wel.wat.edu.pl

***Abstract* — This paper focuses on performance evaluation to study the effects of the resource management strategy in ad hoc networks. The presented strategy is a result of the new outlook on the IEEE 802.11 networks capabilities and performance enhancement. The proposed solution is dedicated to real-time services support and is based on the concept of the Resource Manager that organizes and controls the whole traffic in the network. Novel procedures were developed and applied in order to organize the network and manage the real-time traffic. A method for measuring and estimating available bandwidth was introduced. This method provides wireless station the capability of measuring the bandwidth independently of other stations in the network. Large-scale simulations for different numbers of voice sources and various voice codecs have been carried out. They show an increase of channel utilization reaching over 80% and significant growth of the network capacity, when synchronous transmission is applied.**

Keywords - IEEE802.11 WLANs, ad-hoc networks, VoWiFi, resource management

I. INTRODUCTION

This paper deals with the issue of the resource management strategy in ad hoc wireless networks to support real-time services [1].

For over ten years a permanent development of IEEE 802.11 Wireless Local Area Networks (WLANs) is being observed [2]. Among the many advantages they offer, users appreciated the convenience and simplicity when accessing the network and establishing high data rate wireless connection. Thanks to the low-cost small-size devices, wireless networks seem to be ubiquitous. WLAN drivers are embedded in many different devices like notebooks, mobile phones, Personal Data Assistants (PDAs), cameras, etc. Despite the fact that WLANs were originally designed for data transport, today it is also demanded of them to be efficient for real-time services support.

Another advantage of WLANs results from the ad hoc mode, which is a method for wireless devices to directly communicate with each other. Operating in ad hoc mode allows all wireless devices within each other's range to discover and communicate in a peer-to-peer manner without involving the central access point. This mode offers mobility and communications between users in areas without infrastructure or in all places with damaged infrastructure. From this point of view, WLANs operating in ad hoc mode can be a very promising solution for users, such as the fire brigade, rescue team, police squad or small military unit [3][4]. The possible scenario is to use the ad hoc network for public-safety or search-and-rescue operations.

An important issue for such network is the ability to support cooperation between two or more emergency services, e.g., the fire brigade, police squad, rescue team, medical service. On the other hand, it must be stressed that the performance of the network decreases as the number of wireless users grows. For this reason, a smart mechanism should be introduced, which allows topology control and network scalability [5]. The effect of the hidden node is one of the most difficult problems to solve, because it is intrinsic to the nature of the WLANs. The RTS/CTS mechanism is not recommended for the transmission of small packets, e.g., VoIP. A possible solution is to use an additional signaling channel, however it requires changes in the physical layer. This issue was widely discussed in [15][16].

When considering the hierarchical structure of the command system of the emergency services, different ranks of users should be taken into account. This will affect the priority of users, as well as the type of services allowed.

Although the most common weakness of WLANs is the insufficient support of the real-time services [6][7], the authors formulated a new outlook on the IEEE 802.11b network capability and possible performance enhancement. Despite the fact that there is a wide range of WLANs specifications, the issue of network optimization still remains open. QoS mechanisms were the subject of the IEEE 802.11e standard [8]. However, these mechanisms cannot guarantee the quality of services, although they slightly improve the network efficiency [9].

This paper presents the general concept of the resource management strategy and provides information on introduced procedures. All proposed mechanisms are integrated with each other and interact within an individual device as well as within the whole network.

The rest of the paper deals with the related work (Section 2), concept and assumptions (Section 3), the description of the proposed mechanisms (Section 4), simulation results and

their discussion (Section 5), conclusions (Section 6) and future work (Section 7).

II. RELATED WORK

The voice capacity of IEEE 802.11 networks is gaining increasing attention in the literature. Methods of voice traffic optimization, including voice codec negotiation, audio packets aggregation as well as the MAC protocol adaptation, can be found in many papers. In [9], the influence of the MAC protocol on the network performance was shown. This protocol operates in contention mode and thus inevitably introduces the PHY layer overheads, Backoff and protective periods, ACK frames and retransmissions in some cases. In [10], the authors analyzed the effect of the coding rate and packet size on the voice capacity of the Distributed Coordination Function (DCF).

In [11], dynamic Contention Window (CW) adaptation was suggested in order to minimize the number of collisions. The idea of the voice coding bit rate adaptation to the available network bandwidth was described in [23]. Experimental results confirmed the efficiency of the new scheme. The impact of different configuration parameters on the ad hoc network performance was presented in [24]. The following parameters were analyzed, the type of codec, packetization interval and the data rate. In [25], the authors presented the results of the capacity measurement of the IEEE 802.11e network for each access category. They also analyzed the effect of the TCP traffic on VoIP streams. In conclusion, they stated that 802.11e standard can protect the quality of VoIP if there is TCP traffic added. However, it cannot improve the capacity of the network.

Although proposed methods can improve network efficiency, the question as to how to guarantee the quality of services still remains open. In [14], the authors proposed to introduce additional signaling channel to inform other stations that the main channel is busy. This approach requires modification in physical and data link layers. Furthermore, there is still a lack of an efficient Call Admission Control (CAC) mechanism to protect voice traffic. In [13], a dynamic admission control mechanism was presented. This mechanism is based on the traffic analysis model in order to guarantee the QoS, however is dedicated to the network with infrastructure.

III. CONCEPT AND ASSUMPTIONS

In the case under consideration, the aim of the network optimization is to get as high as possible number of VoIP streams with guaranteed voice quality. The assumed network operates in ad hoc mode and consists of small group of users, e.g., fire brigade or rescue team. In emergency situations they typically use voice communication. Therefore the authors made an assumption that there is only one type of service, namely VoIP.

Users of the network have different ranks, which enables to determine some differences between priorities. Thus, the tradeoff between the available bandwidth, the allowed number and the rank of users is introduced intentionally.

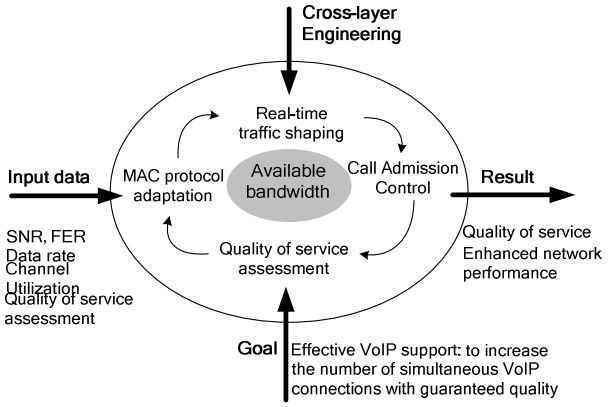


Figure 1. Performance enhancement of WLAN for VoIP support. [1]

The network model assumes WLAN based solution. The authors decided to use the IEEE 802.11b standard as offering good throughput and modulations more resistant to interferences, which is a real advantage of the network operating in ad hoc mode. MAC QoS mechanisms defined in the IEEE 802.11e standard were also taken into account. These mechanisms are a good starting point to enable prioritization and bandwidth reservation in ad hoc network [13][14]. In particular, the authors introduced the adaptation of a CW size to the type of frame and the rank of the user.

Another issue concerns the optimal balance between the traffic load and the services quality in ad-hoc networks.

It is expected that the proposed range of adaptation and introducing of new mechanisms will not demand a high cost of implementation and will be feasible.

Fig. 1 illustrates the concept of efficiency improvement of WLAN for VoIP support. The available bandwidth level is the main factor allowing assessment of the traffic load in the network. Cross-layer mechanisms are crucial for network performance improvement. They enable the real-time traffic shaping or MAC adaptation if the available bandwidth is too small or if the level of service is not satisfactory. The CAC mechanism prevents new VoIP calls if the available bandwidth level is too low.

In the proposed solution, the network consists of different rank users. For the sake of simplicity, high rank users shall be denoted as special users while the rest shall be referred to as commercial users. It perfectly corresponds to the scenario involving humanitarian aid operations, when volunteers help people in service. Another example can be the situation when the fire brigade, police and civilians cooperate within small groups while strengthening an embankment during a flood. However, if the available bandwidth is too small, special users prevail over the network. Eventually, the lowest rank users can be completely blocked.

A separate question is how to assess the resources of the network, e.g., channel utilization. Since ad-hoc networks are bandwidth limited, not all measurement methods can be applied [16][17]. From that reason, non-intrusive methods are proposed for ad hoc networks [18].

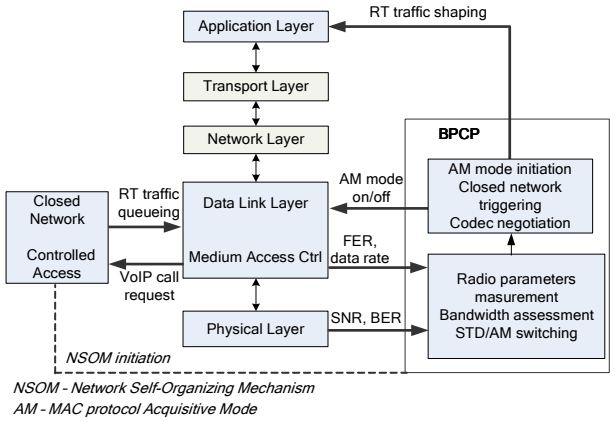


Figure 2. BPCP alignment with protocol stack.

Fig. 2 illustrates the proposed approach on the basis of the extended protocol stack with cross-layer interactions. Bandwidth Prediction Control Protocol (BPCP) allows monitoring of parameters in the physical layer, to measure the channel utilization level and also to switch MAC protocol states, as explained in subsequent sections. Real-time (RT) traffic shaping relays on codec negotiation and audio packets aggregation. Closed Network Mode is based on the concept of the Resource Manager that controls traffic in the network.

BPCP, MAC Acquisitive Mode (AM) and RT traffic shaping mechanisms do not enforce changes to IEEE 802.11 standard. These new mechanisms only enhance functionality of the station that still operates in standard way. Closed Network Mode requires completely new procedures and set of extra frames exchanged in order to organize the network and the RT queue. Although stations operate according to non-standard scheme, this mode offers a real advantage of better channel utilization, which results in increased number of voice streams [26].

In the proposed solution, BPCP provides CAC mechanism with the knowledge of the current and expected channel utilization. These mechanisms exchange the data thanks to the cross-layer engineering, as the CAC operates outside standard WLAN sublayer.

A. Bandwidth Estimation

The available bandwidth is crucial for optimization of the Wi-Fi ad-hoc network. Therefore, the authors proposed to implement BPCP that enables to measure the channel utilization level and to estimate the available bandwidth.

BPCP takes advantage of WLAN card drivers that enable the measurement of signal to noise ratio (SNR) in the PHY layer and bit error rate (BER) calculation, and passing these parameters to the Data Link Layer. If nodes operate in promiscuous mode, they can receive all the traffic sent across the network. As a result, the bandwidth utilization is assessed in all nodes of the network independently and continuously for predefined periods called Sampling Intervals.

From the PHY layer point of view, stations can detect the channel state (idle - no transmission or busy - transmission) and if they operate in promiscuous mode, they can receive and process all frames. The type of received frames (RTS, CTS, DATA, ACK) is recognized in the data link layer.

Knowing the bit rate and the length of received frames it is possible to calculate their transmission duration in the radio channel, denoted as t_{AF} in (1).

$$t_{AF} = \frac{AF\_length\ (bits)}{AF\_bitRate\ (bits/\sec)} \quad (1)$$

Having knowledge of t_{AF} parameters, it is then possible to determine the channel utilization coefficient for the interval, e.g., from t_1 to t_2

$$U = \frac{t_{d1} + t_{d2} + ... + t_{dn} + n \cdot t_{ACK} + n \cdot SIFS + n \cdot DIFS}{t_2 - t_1} \quad (2)$$

where: n is a number of frames; t_{dn} denotes the duration of the n-th data frames; t_{ACK} represents the ACK frame duration, Fig. 3.

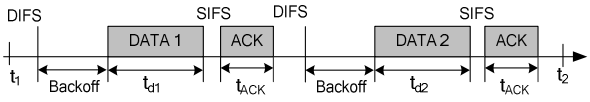


Figure 3. Transmission scheme in a contention mode of WLAN.

When the current and the previous channel utilization levels are estimated, BPCP makes forecasts for the next period.

To validate the model of BPCP, enabling the measurement of the network throughput, theoretical analysis was performed and simulations were made using the OMNET++ v4.0 simulation tool with the INET Framework.

Fig. 4 shows the extended WLAN sublayer of the mobile node. This sublayer contains Throughput Meter In and Throughput Meter Out components to measure all incoming and outgoing traffic. This information is used to assess the total traffic load as well as the available bandwidth.

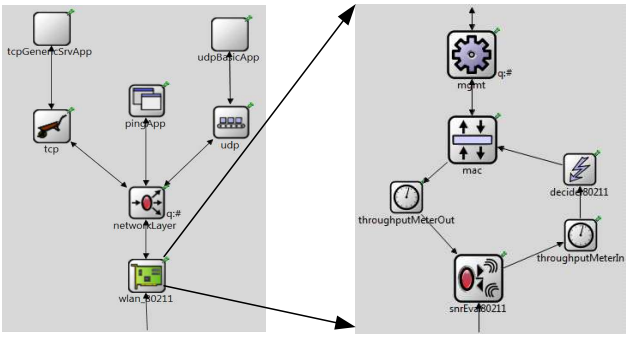


Figure 4. The extended WLAN sublayer of the mobile node.

TABLE I. MAC PARAMETERS

Parameter	Value
DIFS	50 µs
SIFS	10 µs
Slot Time	20 µs
CWmin	32
CWmax	1023
Data Rate	2Mbit/s
PHY header	192 µs
MAC header	34 bytes
ACK	304 µs

For the purposes of analysis and simulation, the following parameters were assumed: G.711 voice codec; typical protocol headers (MAC header = 30B, IPv4 header =20B, UDP header = 8B); free space propagation model and lack of mobility. The issue of mobility is crucial for NRM determination and is the topic of further study.

The values of the MAC and PHY parameters are listed in Table I. The main attributes of the G.711 codec are shown in Table II.

TABLE II. G.711 CODEC CHARACTERISTICS

Codec	G.711
Bit rate [kbit/s]	64
Framing interval [ms]	20
Payload [B]	160
Packets/sec	50

The results of the simulation are presented in Fig. 5. Normal distribution of a throughput estimator was assumed, as well as a confidence interval with α=0,1 and β=1,64 (for cumulative distribution function equal to 0,9).

The period of time required for the transmission of one data frame and the acknowledging frame takes nearly 1,8ms. For that reason it is possible to send 11 acknowledged frames during one second. Audio packets are generated by codec periodically every 20ms. Assuming that stations work synchronously, i.e., after the first one had transmitted a packet, the second one generates it, then it is possible to obtain the network throughput equal to 1,3Mb/s, Fig. 5. Higher traffic load will cause an increase of the collision rate and a drop in network efficiency.

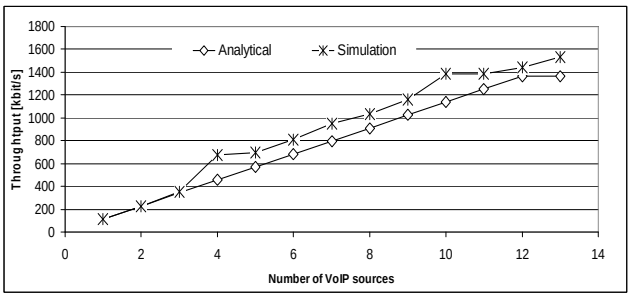


Figure 5. Wi-Fi network throughput - contention mode, data rate 2Mbit/s.

B. The Network Throughput

The purpose of the second part of the simulation was to assess the network throughput and then to determine the levels, where the BPCP triggers MAC AM and Closed Network Mode. After preliminary tests and analysis, the optimal size of the Sampling Interval was set to 1s.

Experiments were performed under the assumption, that the network is dedicated to the VoIP service. Consequently, small size packets are transported across the network, see Table III. The CBR encoding was applied and the interval between packets was set to 20ms in the first case and to 40ms in the second. In both cases the number of VoIP streams was increased until the maximum network throughput was achieved. The collision index was measured as an additional parameter. This collision index is defined as the number of collided frames (coming from all stations) per the total number of frames sent across the network during the Sampling Interval.

TABLE III. SIZE OF PACKETS

Case	UDP payload	Interval
1	40B, 80B, 160B	20ms
2	160B, 240B, 320B	40ms

The results of simulations are presented below. Fig. 6 and Fig. 7 show the network throughput *S* vs. the traffic load *G*. These values were normalized to the data rate in the radio channel *R*, which was set to 2Mbit/s. The network throughput was measured by the BPCP component. Assuming the Poisson process, the traffic load was defined as follows:

$$G = G_{STA\_1} + G_{STA\_2} + \ldots + G_{STA\_N-1} + G_{STA\_N} \quad (3)$$

where G_{STA_N} means the traffic load coming from Station number N.

G_{STA_N} can be calculated from the formula given below.

$$G_{STA\ N} = (UDP\_Payload + OH) \cdot n \quad (4)$$

UDP_Payload is the product of the voice codec, e.g., 160B for G.711 codec, while *n* denotes the number of frames sent per second.

OH represents the total overhead and consists of the UDP, IP, MAC and PHY overheads [9].

$$OH = H_{UDP} + H_{IP} + H_{MAC} + H_{PHY} \quad (5)$$

For the purpose of analysis, the PHY overhead is defined as the sum of the following elements: *DIFS*, average size of *Contention Window*, *PLCP* header and *Preamble*.

$$H_{PHY} = DIFS + CW_{Avg} + H_{PLCP} + preamble \quad (6)$$

Typically, the bitrate of the G.711 voice codec is equal to 64kbit/s. The bitrate at the PHY layer increases up to 96,8kbit/s, when the overheads are taken into account.

Fig. 6 and Fig. 7 shows the relation between the network throughput and the traffic load when packets are generated with the interval of 20ms and 40ms respectively. The shape of the curves is very characteristic. After linear growth of the network throughput the effect of saturation is observed for G/R equal to 0,5. If the traffic load exceeds 0,5 G/R, then the decrease of the throughput is observed, which means the decrease of the amount of data transferred across the network due to collisions and retransmissions. The delay and packet loss ratio exceeds the allowable level. Eventually, if the traffic load is still increasing, the network can be blocked completely. As a result, almost no data is transferred across the network.

If the packets interval increases, e.g., up to 40ms, the effect of saturation occurs for values greater than 0,5 G/R. When UDP Payload is set to 320B, the network is stable up to 0,75 G/R. If 160B UDP Payload is considered, the network throughput amounts to 0,65 for the traffic load reaching about 0,78 G/R.

Fig. 8 and Fig. 9 show the collision index vs. the normalized traffic load for the packets interval of 20ms and 40ms respectively. The value of collision index rises when the traffic load increases. The change of collision index is more rapid for packet of smaller size. For example, for the packets of 40B the collision index reaches the critical level 0,3 for 0,38 G/R. If 160B packets are considered, the critical level is observed for 0,6 G/R.

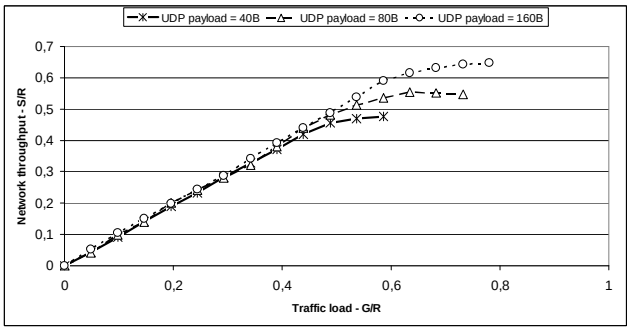


Figure 6. Normalized network throughput – packets generated with the interval of 20ms.

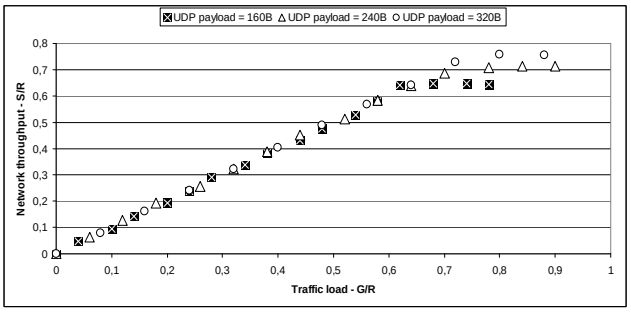


Figure 7. Normalized network throughput – packets generated with the interval of 40ms.

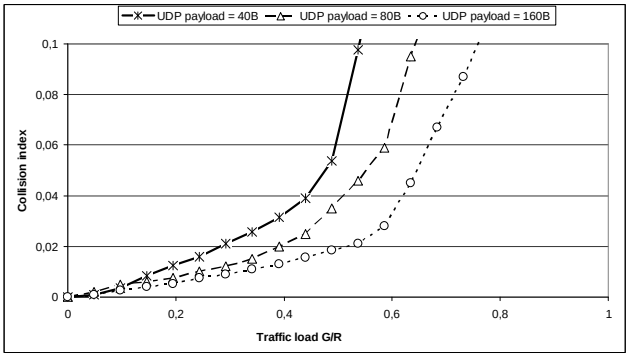


Figure 8. Collision index – packets generated with the interval of 20ms.

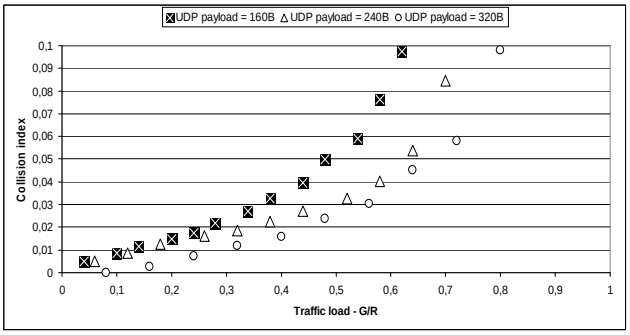


Figure 9. Collision index – packets generated with the interval of 40ms.

Higher traffic load causes sudden growth of collisions, which leads to network inefficiency.

The results presented in Fig. 6 - Fig. 9 are very consistent when comparing packets of the same size and packetization interval. The effect of the network saturation as well as the critical level of collisions is observed for the same value of the traffic load.

C. MAC Protocol States

At the beginning of the operation, special stations can cooperate with commercial and use standard access schemes, until BPCP detects insufficient bandwidth and initiates the MAC AM.

During AM mode, the Backoff interval is minimized according to the rank of the user. As a result, special stations prevail over the network. Only a small part of the bandwidth can be hard-won by remaining users. To determine the Backoff interval, the Contention Window parameter is used, however different values have been introduced, depending on the rank of the user and the type of frame (Control, Data, Broadcast or RTData). It is assumed, that the high rank users use special equipment with modified MAC parameters, while the low rank users are equipped with commercial devices operating with standard configuration, e.g., CW.

If the available bandwidth is still too small, BPCP triggers a mechanism called the Network Self-Organizing Mechanism, which is responsible for creating a Closed Network Mode. From this moment on, Wi-Fi network operates in a point-coordinated mode. Fig. 10 presents the states of MAC protocol for the proposed protocol extension.

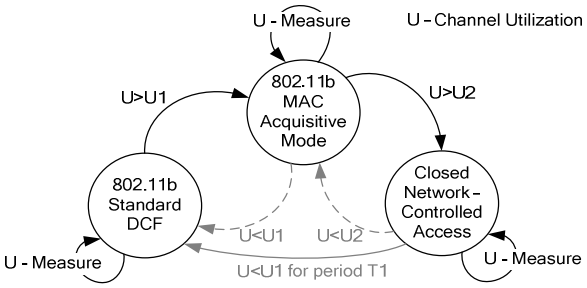


Figure 10. MAC protocol states.

An important issue is to determine the proper level of channel utilization for triggering between MAC AM and Closed Network Mode. To resolve this problem, the authors applied the Pareto optimization approach. Simulation results obtained for 2Mbit/s data rate and G.711 voice codec are presented below.

Fig. 11 shows the network throughput vs. traffic load. Triggering levels are denoted by A, B and C. If the throughput reaches limit denoted by B, the station switches from standard mode to AM. If it reaches another limit denoted by C, the station switches to Closed Network Mode. Fig. 12 presents collisions vs. traffic load. The critical level of collisions is denoted by A. This information may be used additionally by BPCP.

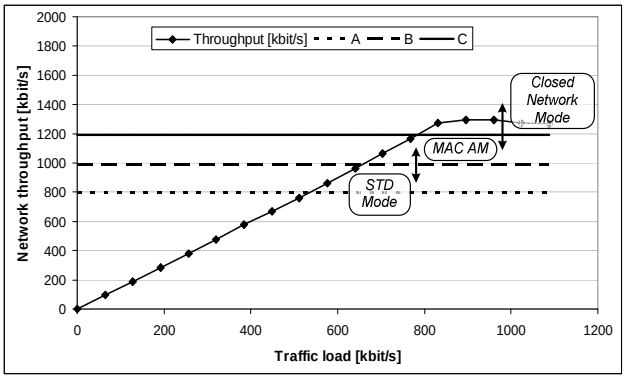


Figure 11. Network throughput vs. traffic load (where triggering levels are denoted as follows: A - switch from AM Mode to std., B - switch from std. to AM Mode, C - switch to Closed Network Mode).

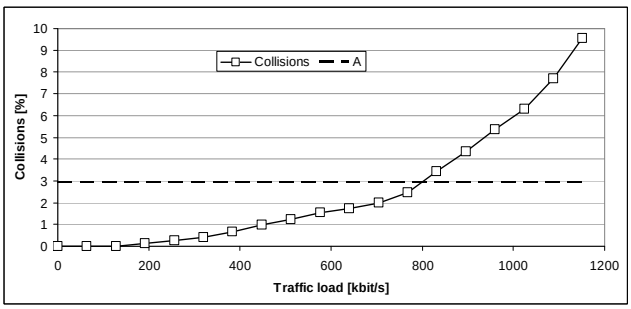


Figure 12. Collisions vs. traffic load (where A denotes the critical level of collisions).

Results of simulations performed in order to estimate the acceptable number of VoIP connections, depending on the type of voice codec and MAC protocol parameters in a contention mode, were widely discussed in literature [10][11][20][21]. However, the question where and how to implement the AC mechanism and how to manage the traffic still remains open. The AC mechanism is necessary to prevent new calls if there is not enough bandwidth. In a contention mode, stations are not aware of the traffic load and try to transmit frames every time they have a packet to send. For this reason, a Closed Network Mode was proposed with a station named the Network Resource Manager (NRM) that manages the network.

IV. CLOSED NETWORK MODE

In this section, the Closed Network Mode was described. This mode enables the ad hoc network to self-organize and to determine the Resource Manager. Comparing to the PCF method, the infrastructure is not required, since the standard ad hoc node is selected to play the Resource Manager role. A set of new procedures was proposed and new frames were introduced as presented in next subsections.

A. Network Self-Organizing Mechanism

Network Self-Organizing Mechanism (NSOM) enables nodes to recognize the neighborhood and to determine the Resource Manager.

At the beginning, all stations work in a contention mode with standard parameters, Fig. 9. In the background, Neighbor Discovery Procedure is performed, which is based on broadcasting Neighbor Request and Neighbor Response frames [22]. This procedure allows recognition of the surroundings by collecting data from other nodes, namely: received signal strength and noise, battery level and rank of the station. Based on this information, each station determines its own *NRM Readiness* coefficient, which describes whether the station is ready to play a network manager role. This mechanism is still under implementation in OMNET++ v4.0.

If BPCP again detects the insufficient bandwidth coincidence, a station changes the mode to AM, while Neighbor Discovery Procedure is still in the background, Fig. 13.

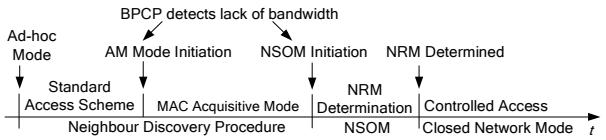


Figure 13. Network Self Organizing procedure.

When a first station detects the insufficient bandwidth, it initiates NSOM. Only stations with a certain *NRM Readiness* coefficient are allowed to participate in this phase.

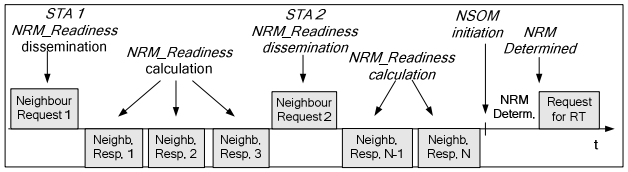


Figure 14. Neighbour Discovery procedure.

If necessary, information on the network topology is refreshed by sending Neighbour Request frame, which contains the last *NRM Readiness* coefficient of the sending station, Fig. 14.

B. Real-time Traffic Management

When the NRM station is determined, it sends a NRM Request broadcast frame informing that nodes are allowed to call for a bandwidth reservation, Fig. 15. Some stations respond with RT Confirm frames if they have RT packets to send, Fig. 16. The NRM Request frame is sent periodically to disseminate the list of queued stations and also the current queue limit. A more detailed description of the algorithm can be found in [19].

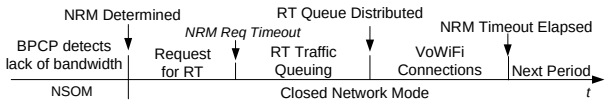


Figure 15. Channel reservation procedure.

If the queue limit is reached or *NRM Req Timeout* has elapsed, the NRM station sends a RT Queue frame containing:

- queue size: number of STAs in queue,
- number of cycles: number of queue repetition,
- voice codec type,
- data rate,
- MAC address and order of stations in the queue.

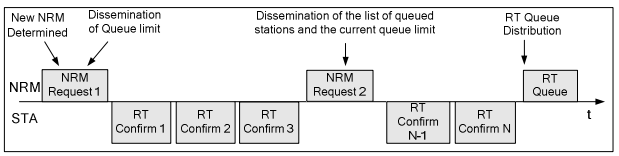


Figure 16. RT traffic management.

After receiving the RT Queue frame, the first station on the list is allowed to transmit after DIFS and receives an ACK frame after SIFS, Fig. 17. The next station in queue transmits data frame after DIFS. The number of cycles describes how long nodes will transmit data in a given order.

After each transmission of DATA and ACK, stations decrease their *TransmissionIndex* and are allowed to send after it reaches zero. After a predefined number of cycles, the NRM station again sends a NRM Request frame to give a chance to transmit for stations that were out of queue during the preceding period.

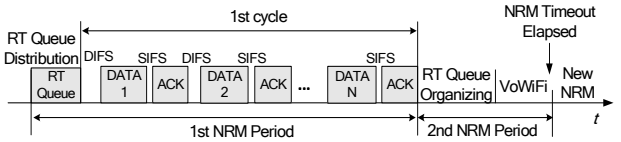


Figure 17. RT traffic queue.

An unpredictable NRM termination may occur, e.g., as a result of depletion of the battery, which should be taken into account. In such a situation nodes will detect a lack of frames from NRM for the assumed timeout. Since this moment on, the station with the second highest *NRM Readiness* coefficient starts playing this role.

In order to organize a closed network and manage RT traffic, the following management frames were introduced:

- Neighbor Request and Neighbor Response - for neighborhood discovering,
- NRM Request - for initiation of the RT traffic queuing phase,
- RT Confirm - for the bandwidth reservation,
- RT Queue - distribution of RT traffic queue.

The detailed description of the management frames structure can be found in [19]. Because all of these frames are of a broadcast type, all receiving stations are forced to process it in the data link layer, although acknowledgement is not sent. The structure of new frames is the same as defined in the IEEE 802.11 standard for management frames and consists of MAC Header and Frame Body containing information fields. The maximum size and capacity of frames are presented in Table IV.

TABLE IV. MANAGEMENT FRAMES SIZE AND CAPACITY

Frame Type	Frame max size [B]	Number of addresses
NRM Request	240	35
RT Confirm	40	1
RT Queue	280	35

V. VOIP CAPACITY ANALYSIS

In order to assess the time required to organize the RT traffic, analytical investigations were performed. It was assumed that NRM is determined, avg. Backoff is equal to 100μs and 10 nodes compete for bandwidth reservation, Fig. 18.

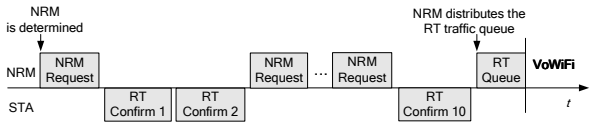


Figure 18. RT traffic scheduling procedure.

The size and the amount of frames exchanged in this procedure are presented in Table V.

TABLE V. AVERAGE SIZE AND NUMBER OF EXCHANGED FRAMES

Frame type	Frame average size [B]	Frames number
NRM Request	100	5
RT Confirm	40	10
RT Queue	100	1

If the data rate is set to 1Mbit/s, one cycle required to schedule the RT traffic takes approximately 14ms and this period reaches 10ms if the data rate increases to 2Mbit/s. Even if some level of collisions is assumed, this duration should not exceed 20ms.

Synchronous RT data transmission in a Closed Network Mode can be verified by using an analytical as well as simulation model. For the sake of convenience, e.g., in order to apply different input parameters, the authors used COMNET 3 simulation tool.

The aim of simulations was to assess the channel utilization and the number of possible simultaneous VoIP calls as a function of the data rate. The following assumptions were made:

- network stations with commercial voice codec (G.711) with attributes defined in Table I,
- MAC/PHY parameters: SIFS = 10μs, DIFS = 50μs, PLCP Header + Preamble = 192μs,
- packets with standard protocol headers: MAC = 30B, IPv4 = 20B, UDP = 8B.

The channel utilization vs. the number of VoIP calls and various data rates was shown in Fig. 19 - Fig. 21.

In the phase of synchronous RT data transmission, there are only two cases when the channel is idle: DIFS, which precedes data frame transmission and SIFS between data frame and ACK frame.

An increasing number of VoIP connections leads to a linear growth of channel utilization, up to 90%. Better channel utilization is unachievable. This is a result of the fact that although the number of frames sent in a given period increases for higher data rates, there are still constant idle periods that separate frames.

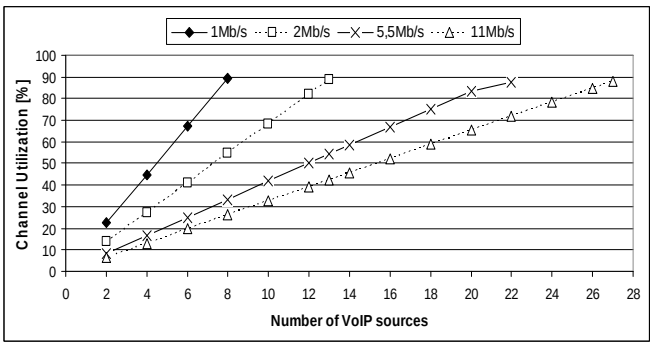


Figure 19. Channel utilization vs. number of VoIP streams for G.711 voice codec (64kbit/s).

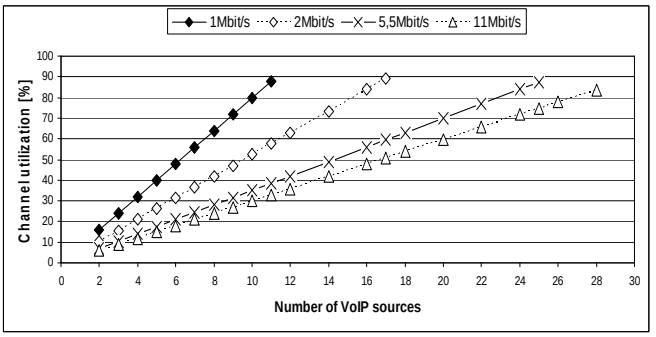


Figure 20. Channel utilization vs. number of VoIP streams for G.726 voice codec (32kbit/s).

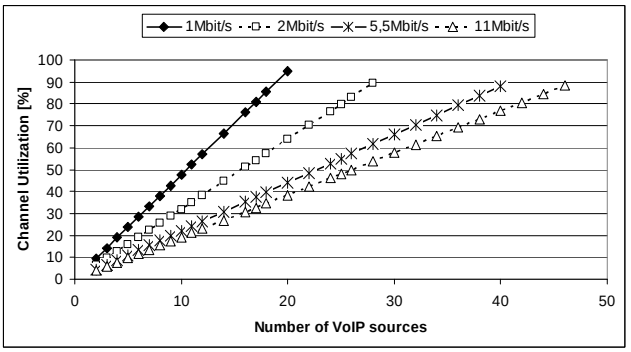


Figure 21. Channel utilization vs. number of VoIP streams for G.728A voice codec (16kbit/s).

The delay of RT packets results from the data rate and the sequence number of a given station in the whole queue. Thus, this delay does not exceed two dozens of milliseconds. When the data rate grows, the time needed for transmission of one frame becomes shorter, while DIFS and SIFS remain on the same level. Therefore it is possible to set up more VoIP connections, however the channel utilization cannot exceed 90%.When G.711 codec is used and the data rate is set up to 11Mb/s, up to 27 VoIP calls are available.

Table VI summarizes the results of experiments showing the maximum number of voice streams sent across the network when synchronous transfer is applied.

TABLE VI. NUMBER OF VOICE STREAMS

Codec	Data rate [Mbit/s]			
	1	2	5,5	11
G.711	8	13	22	27
G.726	11	17	25	28
G.728A	20	28	40	46

VI. CONCLUSIONS

We have presented the concept of the resource management strategy in ad-hoc networks for rescue operations. This strategy is a result of the new outlook on the 802.11 WLANs capabilities and performance enhancement.

A set of novel procedures was developed with a view of organizing the network and managing the real-time traffic. These procedures were validated analytically and by simulations, and results were included. The proposed method of the available bandwidth measurement and estimation works correctly. The triggering levels were determined to switch the MAC protocol mode operation: Standard, MAC Acquisitive Mode or Close Network Mode.

The procedure of RT data synchronous transmission in a Closed Network Mode was verified by simulation. Results of tests allowed estimating the channel utilization achieving over 80% when synchronous transmission was applied. If the number of stations in a queue is set correctly, the delay of the RT data frame transmission is limited to two dozens of milliseconds and is determined mainly by the data rate.

The proposed mechanisms were developed as a result of a completely new approach to the support of RT data transmission in 802.11 ad-hoc network. They enrich standard procedures and enable an efficient utilization of the channel.

VII. FUTURE WORK

In this article, we have only presented the resource management strategy to support VoIP traffic. We described the procedures enabling the organization of network and real-time traffic management.

The presented results were obtained under the assumption that only UDP traffic is transferred across the network. For future research it would be interesting to study the effect of the TCP traffic on the network capacity for VoIP. Based on this work, we are going to investigate how to efficiently manage the network where VoIP streams are combined with the TCP flows.

The issue of nodes mobility is crucial for NRM determination and will be the topic of further study.

Furthermore, we would like to devote attention to the aspect of the distributed network management. This includes optimization of the scheme for determining the secondary resource manager when the first manager terminates unpredictably.

ACKNOWLEDGEMENT

This work was supported by the Polish Ministry of Science and Higher Education under grant number 6266/B/T00/2010/39.

REFERENCES

[1] J. Romanik, P. Gajewski, and J. Jarmakiewicz "A Resource Management Strategy to Support VoIP across Ad hoc IEEE 802.11 Networks," The Fourth International Conference on Communication Theory, Reliability and Quality of Service - CTRQ, Budapest, Hungary, April 17-22.2011

[2] A. F. da Conceicao, J. Li, D. A. Florencio, and F. Kon "Is IEEE 802.11 ready for VoIP," IEEE Workshop on Multimedia Singal Processing, October 2006, pp. 108-113

[3] T. Maseng, "Wireless Tactical Local Area Network," Multinational CDE, Pre-Symposium Workshop, Sundvollen, Oslo, Norway 2002

[4] J. Lopatka and R. Krawczak, "Military Wireless LAN Based on IEEE 802.11b Standard", in Military Communications, Meeting Poceedings RTO-MP-IST-054, Poster 2, France 2006, pp. 21-28

[5] S. Srivathsan, N. Balakrishnan, and S. S. Iyengar, Guide to Wireless Mesh Networks - Scalability in Wireless Mesh Networks,Springer 2009

[6] S. Choi and J. Yu, QoS Provisioning in IEEE 802.11 WLAN, John Wiley & Sons, Inc,2006.

[7] H. Yoon, J. W. Kim, and D. Y. Shin, "Dynamic Admission Control in IEEE 802.11e EDCA-based Wireless Home Network," IEEE Consumer Communication and Networking Conference, January 2006, pp. 1-5

[8] P802.11e – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, Nov. 2005

[9] W. Wang and S. C. Liew, "Solutions to Performance Problems in VoIP over 802.11 Wireless LAN," IEEE Transaction on Vehicular Technology, 54(1), 2005, pp. 336-384

[10] N. Hegde, A. Proutiere, and J. Roberts, "Evaluating the Voice Capacity of 802.11 WLAN under Distributed Control," In Proc of LanMan, 2005, pp. 1-6

[11] L. Gannoune, "A Comparative Study of Dynamic Adaptation Algorithms for Enhanced Services Differentiation in IEEE 802.11 Wireless Ad-Hoc Networks," Proc. of AICT/ICIW, February 2006

[12] L. Cai, Y. Xiao, X. Shen, and L. Cai "VoIP over WLAN: Voice capacity, admission control, QoS and MAC," International Journal of Communication Systems, 2006, (19) pp. 491-508

[13] J. Liu and Z. Niu, "A Dynamic Admission Control Scheme for QoS Supporting in IEEE 802.11e EDCA," Proc. WCNC 2007, pp. 3697-3702

[14] P. Wang, H. Jiang, and W. Zhuang, "A New MAC Scheme Supporting Voice/Data Traffic in Wireless Ad Hoc Networks," IEEE Transactions on Mobile Computing, vol. 7, no. 12, December 2008, pp. 1491-1503

[15] J. Z. Haas and J. Deng, "Dual Busy Tone Multiple Access (DBTMA) - A Multiple Access Control Scheme for Ad Hoc Networks," IEEE Trans. Comm., vol. 50, no.6, June 2002, pp. 975-985,

[16] R. Prasad, M. Murray, C. Dovrolis, and K. Claffy, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools," IEEE Network, vol. 17, no. 6, pp. 27-35, 2003

[17] G. Chelius and I. G. Lassous, "Bandwidth Estimation for IEEE 802.11-Based Ad Hoc Networks," IEEE Transactions on Mobile Computing, vol. 7, no. 10, 2008, pp. 1228-1241

[18] C. Sarr, C. Chaudet, G. Chelius, and I. G. Lassous, "A node-based available bandwidth evaluation in IEEE 802.11 ad-hoc networks," International Journal of Parallel, Emergent and Distributed Systems, vol. 21, 2006, pp. 423-440

[19] J. Romanik, P. Gajewski, and J. Jarmakiewicz, "Performance enhancement of Wi-Fi ad-hoc network for VoIP support," Proc. of Military Communications and Information Systems Conference, Wroclaw, 2010, pp. 525-535

[20] S. Garg and M. Kappes, "An experimental study of throughput for UDP and VoIP traffic in IEEE 802.11b networks," WCNC, March 2003, pp. 1748-53,

[21] F. Anjum, et al., "Voice performance in WLAN networks - an experimental study," Proc. IEEE Globecom, 2003, pp. 3504-3508

[22] M. Bednarczyk and M. Amanowicz, "Wireless Relay Control Protocol (WRCP)," Proc. of Military Communications and Information Systems Conference, Bonn, Germany, September 2007, ISBN 978-3-934401-16-7

[23] H. Zhang, J. Zhao, and O. Yang, "Adaptive Rate Control for VoIP in Wireless Ad Hoc Networks," Proc. of IEEE International Conference on Communications, 2008, pp. 3166-3170

[24] J. Barcelo, B. Ballalta, and C. Cano, “VoIP Packet Delay in Single-Hop IEEE 802.11 Networks,”WONS 2008, Barcelona, pp. 77-80

[25] S. Sangho and H. Schulzrinne, “Measurement and Analysis of the VoIP Capacity in IEEE 802.11 WLAN,” IEEE Transactions on Mobile Computing, vol. 8, 2009, pp. 1265-1279

[26] J. Romanik, P. Gajewski, and J. Jarmakiewicz, “An adaptive MAC scheme to support VoIP across ad hoc IEEE 802.11 WLANs,” Military Communications and Information Systems Conference, Amsterdam, 17-18.10.2011, in: Military Communications and Information Technology: A Comprehensive Approach Enabler, ISBN 978-83-62954-20-9, pp. 496-505.

Real-Time Packet Loss Probability Estimates from IP Traffic Parameters

Ahmad Vakili
Institut national de la recherche scientifique
(INRS-EMT)
Montreal, Canada
vakili@emt.inrs.ca

Jean-Charles Grégoire
Institut national de la recherche scientifique
(INRS-EMT)
Montreal, Canada
gregoire@emt.inrs.ca

***Abstract*—For network service providers, assessing and monitoring network parameters according to a Service Level Agreement and optimal usage of resources is important. Packet loss is one of the main factors to be monitored, especially when IP networks carry multimedia applications. Measuring network parameters is more valuable when it is accurate and online. In this paper, we propose an accurate approximation for packet loss probability at an intermediate high speed node with finite buffer, where a large number of sources are expected to be aggregated. In this method, based on *Large Deviation Theory*, estimation of packet loss probability at the intermediate nodes is based on the input stochastic traffic process. In accordance with *Central Limit Theorem* arguments, the input process is modelled as a general Gaussian process. Different traffic situations and node buffer sizes are simulated (with NS-2 software) and the effectiveness of the method is examined via a detailed numerical investigation. The simulation results show that our proposed method significantly improves the quality of packet loss probability estimate compared to other recently introduced estimators.**

***Keywords*—Packet loss probability, estimation, stochastic traffic process.**

I. Introduction

In telecommunications, performance is assessed in terms of quality of service (QoS). QoS, in turn, is measured either in terms of technology (e.g., for ATM, cell loss, variation, etc.) or at some protocol level (e.g., packet loss, delay, jitter, etc.) [1]–[3].

Today, increased access to Internet networks as well as broadband networks have made possible and affordable the deployment of multimedia applications such as Internet telephony, video conferencing, and IP television (IPTV) by academia, industry, and residential communities. Therefore the quality assessment of media communication systems and the parameters, which affect this quality have been an important field of study for both academia and industry for decades. Due to the interactive or online nature of media communications and the existence of applicable solutions to deduce the effect of delay and jitter (e.g., deployment of a jitter buffer at the end user node [4][5]), data loss is a key issue, which must be considered. If there is a possibility for online accurate measurement of packet loss, then the network service providers can take the appropriate action to satisfy the contractual Service Level Agreement (SLA) or to improve and troubleshoot their service without receiving end user feedback.

Packet loss often happens because of congestion. In other words, buffer overflow at the outgoing interface in intermediate network nodes causes packet loss. Since measuring packet loss ratio at the intermediate nodes in high speed networks does not seem applicable in real time, some recent research has focused on estimation of packet loss probability (plp) [1][6]–[9].

According to central limit theory, the aggregated input traffic at intermediate nodes in the network core can be described with a Gaussian model [10][11]. Based on the Large Deviation Theory (LDT) and the large buffer asymptote approach, the plp can be estimated by a stochastic process considering the probability of buffer overflow in a finite buffer system where b is the buffer size (or tail probability $\mathbb{P}\{Q > b\}$ in a infinite buffer system). Since the input traffic is described by a Gaussian process, the latter can be identified by an online measure of the mean and variance of the input traffic.

In this paper, we propose a tighter approximation of plp based on the input traffic process and the information, which was measured in the past. In other words, we use some online measures and historical data for accurate estimation and thus improve on earlier proposed estimates. Our plp estimation method can also compose with systems whose buffer size is not large enough to meet the assumptions of the large buffer asymptote approach.

Furthermore, this estimate can integrate well with a quality control architecture. Using the online estimated plp as feedback information, a control system could properly throttle the ingress traffic rate and keep the plp below some target upper bound value of packet loss in an SLA. An overall architecture of measurement, estimation, and control loop to keep the quality of service/experience within the SLA bounds is shown in Fig. 1. In this figure, the estimated plp is used as an online transducer in a control loop of packet loss.

The paper continues in Section II by reviewing prior bodies of work on measuring or estimating the packet loss probability. Section III provides some useful definitions, which are employed in this paper. In Section IV, we develop a new plp estimator. Section V presents the testbed and the simulations used to assess the quality of our estimator. Numerical results and comparison that demonstrate the effectiveness of our proposed estimator are presented in Section VI. Section VII concludes the paper and points to our future work.

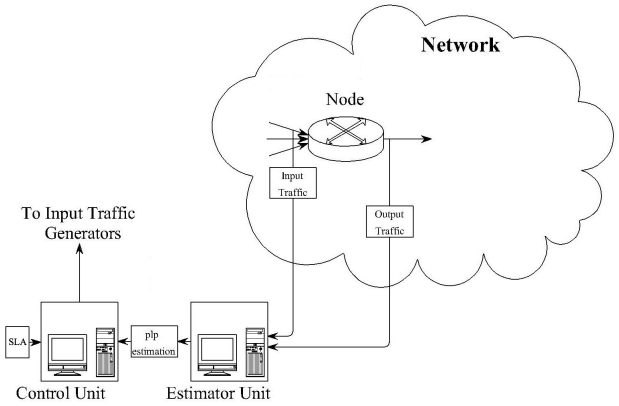


Fig. 1. Measurement, estimation, and control loop schematic.

II. Previous work

In our observations, earlier research on measuring and modelling the packet loss would generally either increase the burden of probe packets' bit rate to the available bandwidth [6][12][13] or not provide real time information [14]–[16]. For example, [14] and [15] have characterized loss traces by identifying mathematical models. Yin Zhang et al. in [17] and [18] have analyzed the stationarity of the loss process on the Internet paths and studied its predictability. Although these studies are undoubtedly useful to understand the general loss characteristics, they cannot be used in real time performance estimation and consequently online control systems.

To obtain real time network performance information such as available bandwidth, delay, and loss, various probing techniques have been recently used by researchers. For instance, [15], [19], and [20] have employed packet pair and packet train techniques, respectively, to measure bottleneck bandwidth. He et al. in [21] have used probing method to explore end-to-end traffic by exploiting the long range dependence nature of Internet traffic. The authors of [12] and [13] have measured the loss rate on individual links by end-to-end multicast/unicast probes and different inference techniques. Further, Tao and Guérin in [6] have used a probing method to construct a Hidden Markov Model (HMM) [22] to capture the main characteristics of loss process such as loss length distribution, loss distance, etc. The disadvantage of these methods is to increase the burden of probe packets' bit rate to the available bandwidth when better accuracy is required.

To cope with the shortcomings of the aforementioned methods, many researchers have tried to link the input process to the loss probability at intermediate nodes. Behavior of the FIFO scheduler fed by many on-off sources was investigated by Anick et al. in [23]. Elvalid et al. and Stern et al. in [24] and [25], respectively, extended Anick's work by presenting a simple approximation of the loss for a very large buffer size system whose input can be modelled with Markov Modulated Rate Processes (MMRP). Their mathematical models are derived from large deviation theory (LDT).

Studies, which estimate loss probability based on input traffic process generally fall into one of the following methodological categories given their underlying assumptions:

- *Large buffer asymptote*: In this approach, the intermediate node's buffer size is assumed to be large. The value of overflow and consequently loss attained in the case of small buffer size is extrapolated using the large buffer asymptote. Chang in [26] and the references therein review this topic comprehensively. Zhang and Ionescu in [8][9][27][28] have extended this research to estimate the loss probability.
- *Large number of sources asymptotic*: This method is based on the homogeneity of n identical sources that feed the intermediate node's input buffer. Likhanov and Mazumdar in [29] used this methodology to estimate the loss probability.
- *Aggregate traffic approximation*: This approach is used to reduce the computational complexity of input traffic model estimation. It is employed when an intermediate high-speed node's input traffic consists of a large number of individual user traffic flows with unique characteristics, in which case the large number of sources asymptotic method is not applicable [30]. The main justification for a packet loss probability estimation based on aggregate traffic approximation is the Bahadur-Rao Theorem, which computes the asymptotic tail distribution of the sum of n identically non-lattice random variables when $n \rightarrow \infty$ [31].

In this paper, we use *large buffer asymptote* approach for online packet loss estimation. Our work revisits Zhang and Ionescu's research [8][9][27][28] (i.e., recent work on this topic); we will review their method and explain how we overcome its shortcomings at the end of Section IV.

III. Definitions

The input traffic model and packet loss probability are explained in this section. All the definitions are related to a high speed intermediate node in which the received packets are served with First In First Out (FIFO) scheduling.

A. Input traffic model

According to the Central Limit Theorem (CLT), the aggregated traffic at an intermediate link in a high-speed network can be well approximated by a Gaussian process [32][33][34]. Moreover, characterizing the input process of a large number of sources with the traditional Markovian models seems infeasible. Therefore, in our study the input process λ_n is characterized by a Gaussian process and presented by

$$\lambda(t) = \mu t + \sigma Z(t), \tag{1}$$

where μ and σ^2 are the mean and variance of arrival rate (i.e., $\lambda(t)$), respectively. $Z(t)$ is a centered Gaussian process when $Var\{Z(t)\} = 1$ [35].

B. Packet loss probability

The packet loss probability, P_{loss}, is defined as the long term ratio of the number of lost packets to the number of input packets. It is expressed by the following formula:

$$P_{loss} = \lim_{N\to\infty} \frac{\sum_{k=1}^{N}(q_{k-1}+\lambda_k - c - b)^+}{\sum_{k=1}^{N}\lambda_k} = \frac{\mathbb{E}[l_k]}{\mathbb{E}[\lambda_k]}, \quad (2)$$

where $(x)^+$ denotes $max\{x, 0\}$, b is buffer size, c is output link capacity, and q_k and l denote the number of packets that occupy the buffer in the time interval $[k, k+1)$ and the number of lost packets, respectively. $\mathbb{E}[x]$ is the expected value of variable x.

The packet loss ratio, $plr(k)$, is defined as the short term ratio of the amount of packets lost to the amount of input packet. It is expressed by the following formula:

$$plr(k) = \frac{l(k)}{\lambda(k)}, \quad (3)$$

where $l(k)$ is the number of lost packets during the time slot $[k, k+1)$ and $\lambda(k)$ is the number of packets that arrive during the time slot $[k, k+1)$.

Kim and Shroff in [32] showed that the plp in a buffer of size x can be well approximately mapped from the tail probability in the infinite buffer system. Tail probability also called the *overflow probability* $\mathbb{P}\{Q > x\}$ is expressed as

$$\mathbb{P}\{Q > x\} = \lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} I(Q_k > x), \quad (4)$$

where $I(A)$ is an identification function, which is equal to 1 if A is true and equal to 0 otherwise, and Q is the dynamic queue size. Although $\mathbb{P}\{Q > x\}$ is averaged by time and plp is averaged by the input, [32] shows the following relationship between $\mathbb{P}\{Q > x\}$ and plp:

$$P_{loss}(x) = \alpha\mathbb{P}\{Q > x\}, \quad (5)$$

where α is constant and equal to $P_{loss}(0)/\mathbb{P}\{Q > 0\}$ and $P_{loss}(0)$ denotes the packet loss probability in a bufferless system.

C. Effective bandwidth

The *effective bandwidth* of arrival traffic process $A(t)$ is defined as

$$\omega(\theta, t) = \frac{1}{\theta t} ln\mathbb{E}[e^{\theta A(t)}] \quad 0 < \theta, t < \infty, \quad (6)$$

where θ and t are system parameters determined by the channel capacity and buffer size, the QoS requirement, and the characteristics of the multiplexed sources [36]. Based on Gärtner-Ellis theorem [37][38], $\omega(\theta, \infty)$ exists when the input traffic is Gaussian. So,

$$\omega(\theta^*, \infty) = \lim_{t\to\infty} \frac{1}{\theta^* t} ln\mathbb{E}[e^{\theta^* A(t)}] = c, \quad (7)$$

where c is link capacity. Glynn and Whitt in [39][40] have proved that overflow probability can be related to θ^*, which is calculated from (7) as following

$$\lim_{x\to\infty} \frac{1}{x} ln\mathbb{P}\{Q > x\} = -\theta^*. \quad (8)$$

IV. Packet Loss Probability Estimator

There are several approaches to estimate packet loss probability. Sending probe packets periodically through the path and processing the returned signals for predicting the performance of path (e.g., packet loss ratio, delay, etc.) is one of the recent methods for estimating the plp [6][41]. The disadvantage of this method is to increase the burden of probe packets' bit rate to the available bandwidth when greater accuracy is requested.

Estimation of plp based on stochastic input traffic process is another approach in this field [8][9][42]. In this method some important assumptions are made as follows: 1) Measurement and estimation take place at intermediate nodes in high-speed network core links, therefore the input traffic is a mix of a large number of individual traffics and thus the Gaussian process model is considered to represent the stochastic input traffic process [10][11]; and 2) the size of the buffer should be large, otherwise the queue process is not exponential and the behaviour of the traffic in small buffers cannot be approximated by a logarithmically linear behavior [43][44][26], so the input traffic process cannot estimate plp.

Following the Gaussian model assumption for the input traffic, the effective bandwidth in this model [36] is given by:

$$\omega(\theta, t) = \mu + \frac{\theta}{2}\sigma^2 t^{(2H-1)} VarZ(t), \quad (9)$$

where θ is the *space* parameter, t is the *time* parameter, which corresponds to the most probable duration of the buffer congestion period prior to overflow, μ is defined as the *traffic mean*, *Var* represents the second moment of *Z(t)*, which is equal to 1 (see (1)), σ^2 is the *variance* of the input traffic random variable, and *H* is the *Hurst* parameter.

The Hurst parameter H shows the degree of self-similarity in the traffic. $H = 0.5$ corresponds to a well behaved Gaussian traffic while any value larger than 0.5 indicates a self-similar traffic source. Based on the classical assumption for input traffic [42][45], the H parameter is set to 0.5. So the effective bandwidth is finite, independent of time, and can be simplified into:

$$\omega(\theta, t) = \mu + \frac{\theta}{2}\sigma^2. \quad (10)$$

Further, if μ and σ exist, effective bandwidth, in case of $t \to \infty$, is equal to link capacity (see (7)). Therefore,

$$\omega(\theta^*, \infty) = \mu + \frac{\theta^*}{2}\sigma^2 = c. \quad (11)$$

Based on our second assumption of large buffer asymptotic approach for packet loss estimation, the overflow probability for the large buffer size can be approximated by a logarithmically behavior as follow [39][40]:

$$\exists\kappa \in \mathbb{R}^+, \mathbb{P}\{Q > x\} = \kappa e^{-\theta^* x}, \quad (12)$$

where θ^* is the solution of (11). Note that such an approximation in (12) is more precise when the buffer size x is large [26]. Therefore, $\mathbb{P}\{Q = x\}$ can be defined by

$$\mathbb{P}\{Q = x\} = \kappa(e^{\theta^*} - 1)e^{-\theta^* x}. \quad (13)$$

To estimate the packet loss probability, $\mathbb{E}[l_k]$ of (2) is defined as follows (recall that b is buffer size):

$$\mathbb{E}[l_k] = \int_b^\infty (x - b)\mathbb{P}\{Q = x\}\, dx. \quad (14)$$

From (13) and (14), we have

$$\mathbb{E}[l_k] = \kappa(e^{\theta^*} - 1)\frac{e^{-\theta^* b}}{{\theta^*}^2}, \quad (15)$$

where θ^* calculated from (11) is

$$\theta^* = 2\frac{c - \mu}{\sigma^2}. \quad (16)$$

Solving (11) in θ^* and replacing in (15) define P_{loss} by the following equation:

$$P_{loss} = \frac{\mathbb{E}[l_k]}{\mathbb{E}[\lambda_k]} = \kappa(e^{2\frac{(c-\mu)}{\sigma^2}} - 1)\frac{e^{-2b\frac{(c-\mu)}{\sigma^2}}}{4\mu\frac{(c-\mu)^2}{\sigma^4}}. \quad (17)$$

Applying the natural logarithm (ln) to (17), we derive the following estimator:

$$\ln(P_{loss}) = \ln(e^{2\frac{(c-\mu)}{\sigma^2}} - 1) - 2b\frac{(c - \mu)}{\sigma^2} - \ln\left(4\mu\frac{(c - \mu)^2}{\sigma^4}\right) + \ln(\kappa). \quad (18)$$

In line with other similar studies [8][9], we change the base of the logarithm function from e to 10. Thus, (18) can be replaced by:

$$\log(P_{loss}) = \log(e^{2\frac{(c-\mu)}{\sigma^2}} - 1) - 2b\frac{c - \mu}{\sigma^2}\log(e) - \log\left(4\mu\frac{(c - \mu)^2}{\sigma^4}\right) + \log(\kappa). \quad (19)$$

Replacing μ and σ with their measurement value $\bar{\mu}(k)$ and $\bar{\sigma}(k)$ changes (19) into the following equation:

$$\log(P_{loss}) = \log(e^{2\frac{(c-\bar{\mu}(k))}{\bar{\sigma}^2(k)}} - 1) - 2b\frac{c - \bar{\mu}(k)}{\bar{\sigma}^2(k)}\log(e) - \log\left(4\bar{\mu}(k)\frac{(c - \bar{\mu}(k))^2}{\bar{\sigma}^4(k)}\right) + \kappa', \quad (20)$$

where $\kappa' = \log(\kappa)$ and $\bar{\mu}(k)$ and $\bar{\sigma}(k)$ are defined as:

$$\bar{\mu}(k) = \frac{1}{N}\sum_{i=0}^{N-1}\bar{\lambda}(k - i), \quad (21)$$

and

$$\bar{\sigma}^2(k) = \frac{1}{N - 1}\sum_{i=0}^{N-1}\left[\bar{\lambda}(k - i) - \bar{\mu}(k)\right]^2, \quad (22)$$

where $\bar{\lambda}(k)$ is the measured input packet rate in the kth time interval and N is the number of time intervals for calculating the average of the mean and variance of the packet rate.

In the rest of the paper let $epl(k)$ denote the $\log(P_{loss})$, which is estimated by

$$epl(k) = \log(e^{2\frac{(c-\bar{\mu}(k))}{\bar{\sigma}^2(k)}} - 1) - 2b\frac{c - \bar{\mu}(k)}{\bar{\sigma}^2(k)}\log(e) - \log\left(4\bar{\mu}(k)\frac{(c - \bar{\mu}(k))^2}{\bar{\sigma}^4(k)}\right), \quad (23)$$

and let $plp(k)$ denote the logarithm of real packet loss probability during the time slot $[k, k + 1)$, which can be expressed by:

$$plp(k) = \log\left(\frac{l(k)}{\lambda(k)}\right), \quad (24)$$

where $l(k)$ is the number of packets lost during the time slot $[k, k+1)$ and $\lambda(k)$ is the number of packets that arrive during the time slot $[k, k + 1)$.

Some estimation errors are expected due to the assumption made for the stochastic traffic process and the simplifications and approximations employed in (23) (e.g., κ' is eliminated from (20)). Numerical results in the next section show that estimating the plp with (23) completely follows the variation of plp, although there is an almost constant offset between the real plp value and epl, which is best explained from ignoring the constant κ' in (20).

To eliminate this difference it is proposed to use the offline measured plp and compare it with the estimated one to obtain the offset. We therefore present an improved estimator, iep, defined as:

$$iep(k) = epl(k) + \frac{1}{n}\sum_{l=1}^{n}\left[plp(k - l - m) - epl(k - l - m)\right], \quad (25)$$

where m is the number of interval periods after which the data of plp is available and $epl(k)$ and $plp(k)$ are calculated via (23) and (24), respectively.

With this improved estimator, the required time for measuring and calculating the plp is represented by m in (25), where the mean of errors between epl and plp during a moving window (i.e., n time intervals) in the past (i.e., m time intervals ago) is added to epl to estimate the new plp. Note that the duration of the time interval is independent from the measurement and calculation speed of plp. In other words, the estimator depends on m, in (25), only for the duration of the measuring time interval.

As we have mentioned in Section II, Zhang and Ionescu [8][27][28] also have proposed a packet loss probability estimator based on LDT and buffer asymptote approach. Their estimator describes packet loss probability by:

$$epl' = \log(P_{loss}) = -2b\frac{c - \mu}{\sigma^2}\log(e) - \log\left(2\mu\frac{(c - \mu)}{\sigma^2}\right). \quad (26)$$

To cope with their estimator's error, they have introduced a Reactive Estimator (re) [9], which is defined as:

$$re(k) = epl'(k) + \frac{1}{n}\sum_{l=1}^{n}\left[plp(k - l) - re(k - l)\right], \quad (27)$$

where epl' is packet loss probability estimated by (26).

A careful examination of (27) reveals that the error re attempts to correct will decrease to the amount of difference between re and plp, whereas the error really is the difference between epl' and plp.

We thus claim that our proposed estimator, iep, does a better job at tracking plp. To investigate the accuracy and applicability of the aforementioned estimators and to compare their performance with that of our estimator, we propose to conduct simulations. In these simulations, the effects of different configurations of network traffic and packet loss ratio on estimators' performance are examined, and then will be discussed in detail in Sections V and VI.

V. Simulation Testbed

The NS-2 software [46] is used to simulate the network. The network topology, which is simulated is shown in Fig. 2.

An MPEG2 traffic flow is generated by node 1 and the Real-time Transport Protocol (RTP) is deployed for transferring video data to node 4. Node 2 generates the voice traffic flow, which is coded by G.729 [47]. This data is transferred to node 5. Node 3 and node 6 are designed to generate the common Internet traffic flow for background traffic and make the aggregated traffic situation closer to the Gaussian distributed traffic for stochastic input traffic process. The Tmix module in NS-2 is utilized in node 3 and 6 in order to generate realistic Internet network traffic [48]. The protocol deployed for communications between nodes 3 and 6 is TCP. Since the background traffic is TCP-based, congestion (i.e., buffer overflow and loss) affects traffic flows, which leads to a situation similar to that of a real Internet network traffic. Nodes 7 and 8 generate the on-off traffic to randomly increase the probability of packet loss. Measurement of the input and output traffics is performed at node 9. Since the focus is on node 9, the bandwidth of all links except link A is set to 100 Mbps and the buffer size of all nodes except node 9 is set to 500 packets. We vary the size of the node 9 buffer from 5 packets to 100 packets to examine different router configurations. To generate different amounts of packet loss, the bandwidth of link A varies between 7.4 Mbps and 7.8 Mbps. With these settings loss takes place only in node 9. When the bandwidth of link A is set to 7.8 Mbps and nodes 7 and 8 do not generate any traffic, the packet loss probability is about 0.1 percent and when the bandwidth is decreased to 7.4 Mbps, the packet loss probability in node 9 increases to about 1 percent, which is closer to the amount where effect of loss on media communication quality becomes annoyingly noticeable. By turning on the traffic of nodes 7 and 8 at some short periods of time, the packet loss probability reaches 7 percent, which is an unacceptable amount of packet loss for media communications. In the next section the numerical values of the different estimators in these situations will be examined.

VI. Numerical Results Analysis

This section presents the experimental results of the evaluation of the performance of the proposed estimator for the different types of traffic generated in the simulation testbed. The accuracy of the loss probability predicted by our proposed estimator is compared to that of a couple of other recent estimators.

A. Input traffic

The crucial assumption in estimating loss probability based on input traffic process is the Gaussian behavior of the aggregated input traffic. Therefore, the verification of this statement (i.e., the aggregated input traffic process is a Gaussian process) is the first test, which should be performed. So, the received times of all packets for aggregated traffic are measured, while node 1 generates MPEG2 traffic flow, a voice traffic is generated by node 2, and node 3 sends an approximate common Internet traffic mix through the core of testbed.

In this paper the graphical technique is used for normality testing, although, the Chi-Square test [49] could also be used to verify the assumption of Gaussian behavior of input traffic in our simulations. Fig. 3, which shows the instantaneous input traffic bit rate and the distribution of input traffic visually, verifies that in our simulations the aggregated traffic in core link can be approximated by Gaussian traffic and consequently, the main assumption of proposed estimator is met.

B. Individual flow loss

To satisfy the SLA and to take the appropriate action on each flow's source, a control system needs to be aware of the packet loss probability of each flow. However, only the aggregated traffic loss probability can be estimated by the proposed estimator.

The simulation results show that the loss ratio of each flow (e.g., MPEG2 flow) is very close to loss ratio of the aggregated traffic. Therefore, it can be concluded that the estimated loss probability of aggregated traffic can be used as the individual probability of packet loss. Fig. 4 verifies this statement by showing that the measured MPEG2 flow's packet loss ratio is very close to the packet loss ratio of aggregated traffic in node 9.

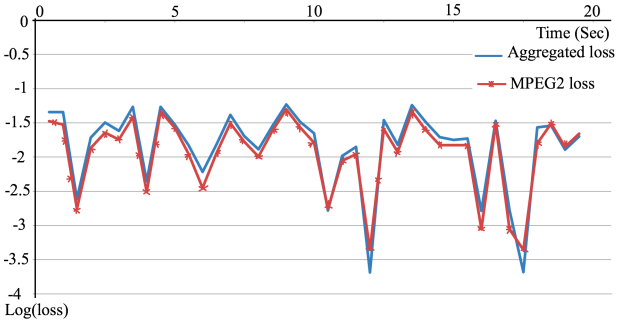


Fig. 4. Comparison of the MPEG2 loss ratio with the aggregated traffic loss ratio.

C. Estimator performance

First, to evaluate that if the epl from (23) follows the plp variation with an almost constant offset, a situation has been

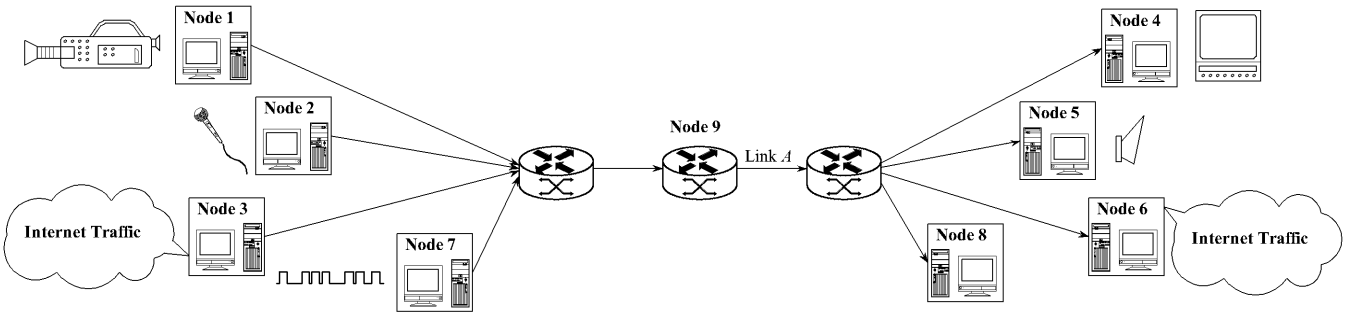
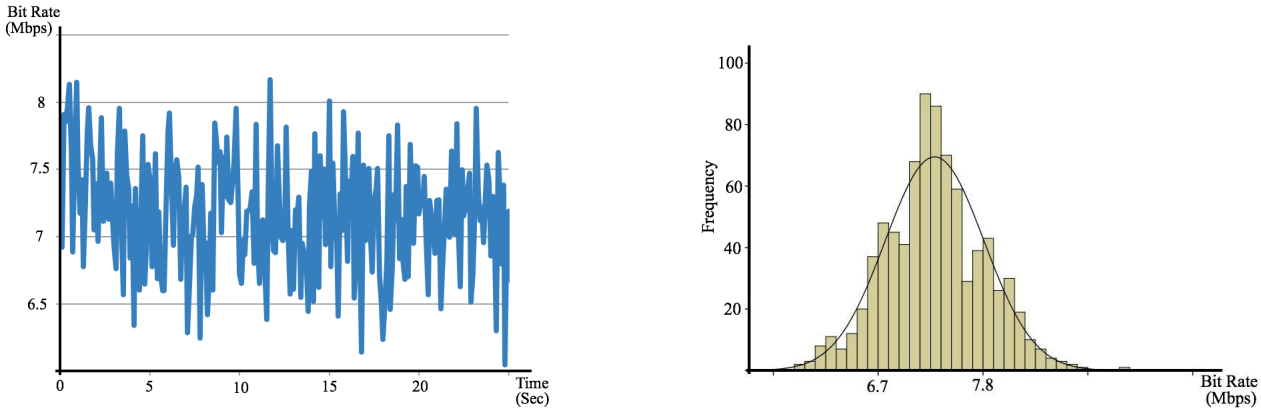


Fig. 2. Testbed topology.



(a) Instantaneous input traffic bit rate.

(b) Histogram of input traffic distribution.

Fig. 3. Aggregated input traffic characteristics in network core.

investigated in which the bandwidth of link A was 7.4 Mbps and there was no traffic coming from nodes 7 and 8. As shown in Fig. 5, although there is an offset between plp and epl, epl follows the variation of plp thoroughly and this can be seen as a clear sign of soundness of the use of epl as the main part of proposed estimator.

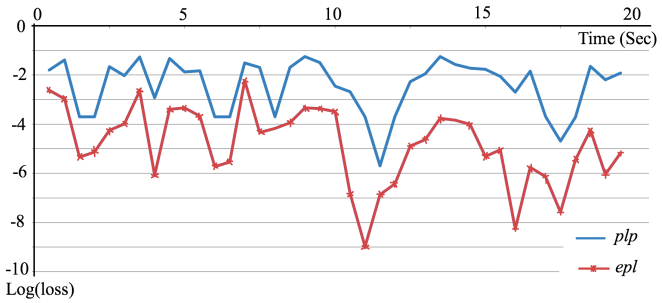


Fig. 5. Comparison of plp (measured loss) and epl (estimated loss with offset).

Next, all the mentioned estimators (i.e., epl', re, and our proposed estimator, iep) are evaluated and their performance compared in different situations.

Fig. 6 shows the performance of the different estimators in a situation where the bandwidth of link A is 7.8 Mbps and there is no traffic coming from nodes 7 and 8. The accuracy of proposed estimator (iep) to estimate the plp compared to the other estimators is demonstrated in this figure.

In all experiences the time interval is 100 ms. In Fig. 6 iep is calculated according to (25) where m is 5. This means that iep uses plp data measured up to 500 ms earlier.

Since the amount of loss in the former example might be negligible for media communications, we change the network conditions to increase the loss ratio and then re–evaluate the accuracy of estimators. To achieve this situation, the buffer size of node 9 is decreased to 10 packets. Fig. 7 shows the results of this experience: during the time periods of $[10, 15]$, nodes 7 and 8 add network traffic and bring the loss ratio close to 7 percent ($log(plp) = -1.5$). As Fig. 7 shows, the effect of simplification and approximation in (26) and (27) on the operation of epl' and re methods is more apparent at this larger loss ratio.

Tables I and II summarize the statistics for the different estimators with varying loss ratio. In all comparisons the error is defined as the difference between estimated and measured plp.

As mentioned before, the buffer size affects the plp and the accuracy of estimators [43][44]. The larger the buffer size, the lesser plp and the better the accuracy of the estimation. The

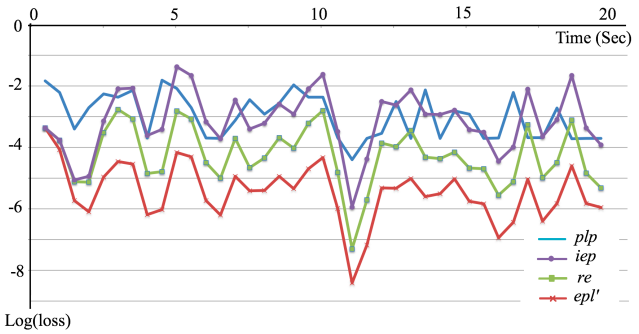


Fig. 6. Measurement and estimation of packet loss probability when plp is about -2.5.

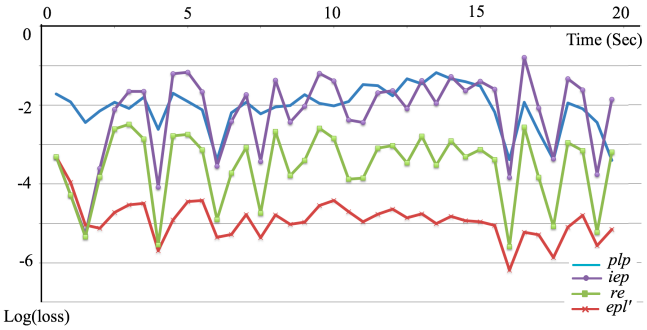


Fig. 7. Measurement and estimation of packet loss probability when plp is about -1.5.

effect of buffer size on estimation methods re and epl' has been examined in [8] and [27], respectively. The value of m, in (25), also affects the accuracy of iep estimation.

To examine the accuracy of the proposed estimator in different configurations (i.e., buffer size and m), we introduce a new variable, $error$. Given that the errors of logarithmic variables (plp's) are not easily comparable, $error$ is defined as follows to make it more sensible to small variations:

$$error = 10^{estimation} - 10^{plp}. \quad (28)$$

Fig. 8 shows the probability density function of $error$ when buffer size is 10, 30, and 100 packets, and m is 5, 10, and 20 ($m = 10$ means using a plp measured 1 s before), and the effect of buffer size on estimation. Fig. 8-(a),(b), and (c) show that our proposed estimator has better performance in the case of a larger buffer. Note that a larger buffer size causes more latency, which is not suitable particularly for multimedia transmission; hence, it should be set carefully. However, in our simulations, the buffer size of 100 packets causes only a 15 ms delay, which could be even lower in real high speed intermediate networks.

It can be also shown by Fig. 8 that the offline measuring speed affects the accuracy of our proposed estimator: the faster the measurement, the more accurate the estimation.

Further considering the effect of buffer size on estimations derived from (23), it appears that the accuracy of estimation

TABLE I
STATISTICS SYNOPSIS ON LOSS PROBABILITY ESTIMATION FOR DIFFERENT ESTIMATORS WHEN plp IS ABOUT -2.5.

Estimator	Error* Mean	Error Variance	Error Min	Error Max
iep	0.16	0.77	0.016	2.24
epl'	2.47	0.60	0.88	4.22
re	1.27	0.70	0.25	2.98

Error* is equal to difference between estimations (iep, epl', and re) and plp.

TABLE II
STATISTICS SYNOPSIS ON LOSS PROBABILITY ESTIMATION FOR DIFFERENT ESTIMATORS WHEN plp IS ABOUT -1.5.

Estimator	Error Mean	Error Variance	Error Min	Error Max
iep	0.19	0.45	0.016	2.8
epl'	2.86	0.50	1.59	3.8
re	1.49	0.24	0.20	2.93

(iep) will improve if the role of the measured plp is increased. Therefore, (25) is changed to:

$$iep(k) = p \times epl(k) + \frac{1}{n}\sum_{l=1}^{n}[plp(k-l-m) - p \times epl(k-l-m)], \quad (29)$$

where p is the proportional coefficient and is less than 1. To increase the importance of the second term in (29), n is increased from 3, which is recommended in [9], to 10 and to decrease the effect of first part, p is set to $\frac{2}{3}$. For a smaller p, when a considerable variation happens to plp, the estimator (iep) cannot follow the plp properly and the value of $error$ will be significant.

Fig. 9 shows the value of $error$ when buffer size is 10 and (29) is used for estimation. Comparing Fig. 9 and Fig. 8(a), the effectiveness of the changes in estimation is clear.

To conclude, the advantages of our proposed estimator are: 1) an increase in the accuracy of estimation by using the measured parameters properly, 2) flexibility on the duration of measuring time interval, and 3) an estimate of plp reasonably accurate in the case of a small buffer.

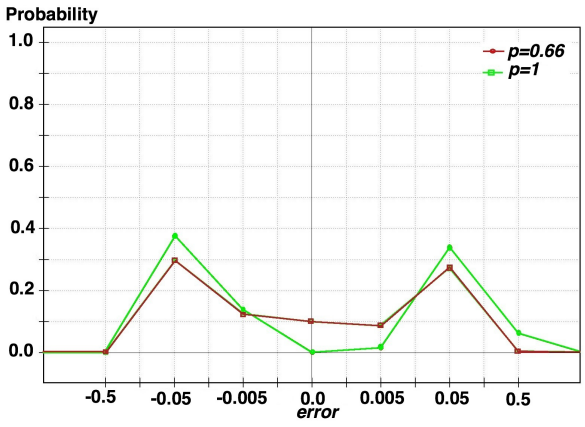
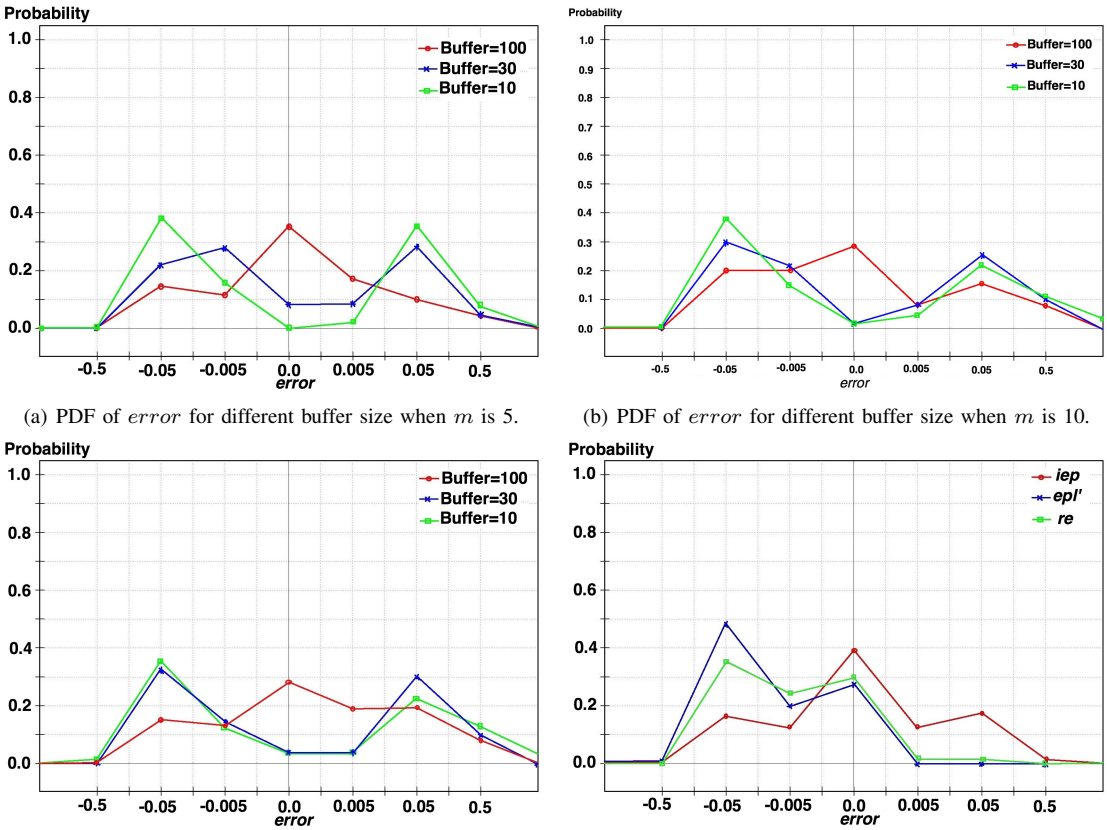


Fig. 9. PDF of $error$ for estimator, which uses (29) when buffer size is 10.



(a) PDF of $error$ for different buffer size when m is 5.

(b) PDF of $error$ for different buffer size when m is 10.

(c) PDF of $error$ for different buffer size when m is 20.

(d) PDF of $error$ for different estimators when buffer size is 100, m=5, and there is no random traffic.

Fig. 8. The comparison of PDF of $error$ for different conditions.

VII. Conclusion

One of the most important issues in multimedia quality of experience is packet loss, which has an especially critical role in interactive communications. Accurate online network-based measurements of loss are necessary to give service providers the means to estimate the quality received by a user and to give them an opportunity to take remedial actions to satisfy the contractual SLA. Increased use of multimedia communications in the Internet has led to a renewed interest in the measure and estimation of loss, in the form of the plp, in modern communication networks. More specifically, recent studies have focused on estimation of the plp by measurement of input traffic based on LDT and the large buffer asymptote. In this paper, we have reviewed the theory behind the finite buffer overflow probability (tail probability in infinite buffer) estimation. Based on *central limit theory*, by modelling the input traffic of an intermediate high speed node as a Gaussian process, we have introduced a new approximation for plp. Combining this online approximation with the offline output traffic measurement, we have proposed an accurate plp estimator, which significantly improves the quality of the estimate compared to the recent proposed plp estimators [27][28], which have used similar theoretical basis.

To study the accuracy of the estimates, we have used the NS-2 simulator with the input traffic, which is very similar to the Internet traffic at the measurement node. Overall, the simulation results demonstrate the effect of different configurations, such as buffer size, on the estimates. The analysis of the results shows the improvement of accuracy in plp estimation achieved by our new calculation method.

For future research, we plan to investigate how it is possible to estimate the end user's perception, aka the Quality of Perception (QoP), based on the effect of loss. Along this line of research, we plan to study the methods of estimation of other network parameters (e.g., delay and jitter) to utilize them as the input of QoP measurement.

References

[1] A. Vakili and J. C. Grégoire, "Estimation of packet loss probability from traffic parameters for multimedia over IP," Proc. of the Seventh International Conference on Networking and Services, ICNS 2011, pp. 44–48, May 2011.

[2] D. McDysan, *QoS & traffic management in IP & ATM networks*, McGraw-Hill, 2000.

[3] W. C. Hardy, *VoIP service quality: measuring and evaluation packet-switched voice*, McGraw-Hill, 2003.

[4] B. Oklander and M. Sidi, "Jitter buffer analysis," Proc. of 17th IEEE International Conference on Computer Communications and Networks, pp. 1–6, August 2008.

[5] H. Hata, "Playout buffering algorithm using of random walk in VoIP," Proc. of IEEE International Symposium on Communications and Information Technology, pp. 457–460, October 2004.

[6] S. Tao and R. Guerin, "On-line estimation of internet path performance: an application perspective," Proc. of 23rd IEEE Conference on Computer Communications, Vol. 3, pp. 1774–1785, March 2004.

[7] R. Serral-Gracia, A. Cabellos-Aparicio, and J. Domingo-Pascual, "Packet loss estimation using distributed adaptive sampling," Proc. of IEEE Workshop on End-to-End Monitoring Techniques and Services, pp. 124–131, April 2008.

[8] D. Zhang and D. Ionescu, "A new method for measuring packet loss probability using a Kalman filter," IEEE Transaction on Instrumentation and Measurement, Vol. 58, No. 2, pp. 488–499, February 2009.

[9] D. Zhang and D. Ionescu, "Reactive estimation of packet loss probability for IP-based video services," IEEE Transaction on Broadcasting, Vol. 55, No. 2, pp. 375–385, June 2009.

[10] R. van de Meent and M. Mandjes, "Evaluation of user-oriented and black-box traffic models for link provisioning," Proc. of the 1st EuroNGI Conference on Next Generation Internet Networks Traffic, pp. 380–387, April 2005.

[11] J. Kilpi and I. Norros, "Testing the Gaussian approximation of aggregate traffic," Proc. of the 2nd ACM SIGCOMM Workshop on Internet Measurement, pp. 49–61, November 2002.

[12] R. Caceres, N. G. Duffield, J. Horowitz, D. Towsley, and T. Bu, "Multicast-based inference of network-internal characteristics: Accuracy of packet loss estimation," Proc. IEEE INFOCOM, New York, March 1999.

[13] N.G. Duffield, F. Lo Presti, V. Paxson, and D. Towsley, "Inferring link loss using striped unicast probes," Proc. of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, Vol.2, pp. 915–923, 2001.

[14] M. Yajnikk. S. Moon. J. Kurose, and D. Towsley, "Measurement and modeling of the temporal dependence in packet loss," Proc. of IEEE INFOCOM. New York, March 1999.

[15] V Paxon. "End-to-end Internet packet dynamics," IEEE/ACM Transaction on Networking. Vol. 7. No. 3, pp. 277–292. June 1999.

[16] J. Bolot, "End-to-end packet delay and loss behavior in the Internet," Proc. ACM SIGCOMM, San Francisco, CA, September 1993.

[17] Y. Zhang, V. Paxson, and S. Shenker, "The stationarity of Internet path properties: routing, loss, and throughput," ACIRI Technical Report, 2000

[18] Y. Zhang, N. Duffield, V. Paxan, and S. Shenker. "On the constancy of Internet path properties," Proc. of ACM SIGCOMM Internet Measurement Workshop, San Francisco. CA. November 2001.

[19] V. Jacobson, "Pathchar: a tool to infer characteristics of Internet paths," Mathematical Sciences Research Institute MSRI Workshop, April 1997.

[20] M. Jain and C. Dovrolis, "End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput," Proc. of ACM SIGCOMM, pp. 295–308, 2002.

[21] G. He and J. C. Hou, "On exploiting long range dependence of network traffic in measuring cross traffic on an end-to-end basis," Proc. of IEEE lNFOCOM, March, 2003.

[22] K. Salamatian and S. Vaton, "Hidden Markov modeling for network communication channels," ACM SIGMETRICS Cambridge, 2001.

[23] D. Anick, D. Mitra, and M.M. Sodhi, "Stochastic theory of a data handling system with multiple sources," Bell Sys. Tech Journal, Vol. 61, No. 8, 1982.

[24] A. Elwalid and D. Mitra, "Effective bandwidth of general Markovian traffic sources and admission control of high speed networks," IEEE Transactions on Networking, Vol. 3, pp. 329–343, 1993.

[25] T.E. Stern and A.I. Elwdid, "Analysis of a separable Markov modulated rate models for information handling systems," Advances in Applied Probability, Vol. 23, pp. 105–139, 1992.

[26] C. Chang, *Performance guarantees in communication networks*, New York: Springer-Verlag, 2000.

[27] D. Zhang and D. Ionescu, "Online packet loss measurement and estimation for VPN-based services," IEEE Transactions on Instrumentation and Measurement, Vol. 59, No. 8, pp. 2154–2166, Aug. 2010.

[28] D. Zhang and D. Ionescu, "On packet loss estimation for virtual private networks services," Proc. of 13th IEEE Conference on Computer Communications and Networks, pp. 175–180, October 2004.

[29] N. Likhanov and R.R. Mazumdar, "Cell loss asymptotics in buffers fed with a large number of independent stationary sources," Proc. of INFOCOM, Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1, pp. 339–346, 1998.

[30] C. Lambiri, "On the estimation and control of packet loss for VPN services," Ph.D. dissertation, University of Ottawa, Ottawa, 2003.

[31] R. R. Bahadur and R. Ranga Rao, "On deviations of the sample mean," Ann. Mathematical Statistics, Vol. 31, No. 23, pp. 1015–1027, 1960.

[32] H. S. Kim and N. Shroff, "Loss probability calculations and asymptotic analysis for finite buffer multiplexers," IEEE/ACM Transaction on Networking, Vol. 9, No. 6, pp. 755–767, Dec. 2001.

[33] J. Choe and N. Shroff, "A central-limit-theorem-based approach for analyzing queue behavior in high-speed networks," IEEE/ACM Transaction on Networking, Vol. 6, No. 5, pp. 659–671, Oct. 1998.

[34] K. Debicki and M. Mandjes, "Exact overflow asymptotics for queues with many Gaussian inputs," Journal of Applied Probability, Vol. 40, pp. 704–720, 2003.

[35] A. Leon-Garcia, *Probability, Statistics, and Random Processes for Electrical Engineering*, Prentice Hall, 1994.

[36] F. Kelly, "Notes on effective bandwidths," in Stochastic Networks: Theory and Applications, Oxford University Press, pp. 141–168, 1996.

[37] J. Gärtner, "On large deviations from invariant measure," Theory of Probability and Its Applications, Vol. 22, pp. 24–39, 1977.

[38] R.S. Ellis, "Large deviations for a general class of random vectors," Ann. Probability, Vol. 12, pp. 1–12, 1984.

[39] W. Whitt, "Tail probabilities with statistical multiplexing and effective bandwidths in multi-class queues," journal of Telecommunication Systems, Vol. 2, pp. 71–107, 1993.

[40] P.W. Glynn and W. Whitt, "Logarithmic asymptotics for steady-state tail probabilities in a single-server queue," Journal of Applied Probability, Vol. 31, 1994.

[41] S. Tao, K. Xu, A. Estepa, T.F.L. Gao, R. Guerin, J. Kurose, D. Towsley, and Z.L. Zhang, "Improving VoIP quality through path switching," Proc. of 24th IEEE Conference on Computer Communications, Vol. 4, pp. 2268–2278, March 2005.

[42] C. Lambiri, D. Ionescu, and V. Groza, "A new method for the estimation and measurement of traffic packet loss for virtual private networks services," Proc. of the 21st IEEE Conference on Instrumentation and Measurement Technology, Vol. 1, pp. 401–406, May 2004.

[43] D. P. Heyman and T. V. Lakshman, "What are the implications of long-range dependence for VBR-video traffic engineering?," IEEE/ACM Transactions on Networking, Vol. 4, No. 3, pp. 301–317, June 1996.

[44] B. K. Ryu and A. Elwalid, "The importance of long-range dependence of VBR video traffic in ATM traffic engineering: myths and realities," Proc. of ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp. 3–14, August 1996.

[45] X. Yu, I. L. Thng, and Y. Jiang, "Measurement-based effective bandwidth estimation for long range dependent traffic," Proc. of IEEE Region 10 International Conference on Electrical and Electronic Technology TENCON, Vol. 1, pp. 359–365, 2001.

[46] UC Berkeley, LBL, USC/ISI, and Xerox PARC, *Network simulator NS-2*, http://www.isi.edu/nsnam/ns, [Accessed: 14 June 2012].

[47] ITU-T Recommendation G.729, "Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)," 2007.

[48] M.C. Weigle, P. Adurthi, F. Hernandez-Campos, K. Jeffay, and F.D. Smith, "Tmix: A tool for generating realistic application workloads in NS-2," ACM SIGCOMM Computer Communication Review, Vol 36, No 3, pp. 67–76, July 2006.

[49] P.E. Greenwood and M.S. Nikulin, *A guide to chi-squared testing*, Wiley-Interscience, 1996.

Resource Management in Multi-Domain Content-Aware Networks for Multimedia Applications

Eugen Borcoci, Mihai Stanciu, Dragoş Niculescu, Şerban Obreja

Telecomunications Dept.
University POLITEHNICA of Bucharest
Bucharest, Romania
{eugenbo, ms, dniculescu, serban}@elcom.pub.ro

***Abstract*—The significant orientation of the current Internet towards information/content determined the appearance of new solutions and concepts among which the Content Aware Networking is a significant one. Virtual Content Aware Networks (VCAN) constructed as overlays over IP network substrate is considered an efficient solution to incrementally introduce content awareness at network level. This paper continues a previous effort to define and develop a new framework for connectivity resources management in overlay VCANs, built over multi-domain, multi-provider IP networks. The VCANs are created and managed by novel business entities called CAN Providers and they offer enhanced connectivity services to high level Services Providers (SP), including unicast, multicast, and P2P in a multi-domain networking context. The paper develops the management system and procedures to negotiate and allocate the connectivity resources in different network domains, independently managed, but cooperating to create VCANs. The management framework is based on vertical and horizontal Service Level Agreements (SLA) negotiated and concluded between providers and possibly also on content/service description information (metadata) inserted in the media flow packets by the servers.**

Keywords*—*Content-Aware Networking; Network Aware Applications; Connectivity services; CAN Management; Multimedia distribution; Future Internet

I. INTRODUCTION

The Future Internet has a strong orientation towards services and content [1][2][3]. A new solution to make the Future Internet more content oriented, is to create virtualized *Content-Aware Networks* (CAN) and *Network-Aware Applications* (NAA) on top of the flexible IP [3][4][5][6]. Additionally to routing, the CAN routers are optimized for filtering, forwarding, and transforming inter-application messages on the basis of their content and context.

The work of this paper is part of an activity performed in the framework of a new European FP7 ICT research project, "Media Ecosystem Deployment Through Ubiquitous Content-Aware Network Environments", ALICANTE [7] [8][9] and is a continuation and extension of the work presented in [10] and [11]. The following inter-working multi-actor environments are defined: *User Environment (UE)*, to which some end users belong; *Service Environment (SE)*, to which Service Providers (SP) and Content Providers (CP) belong; *Network Environment (NE)*, to which the Network Providers (NP) belong. *Environment* is a generic name for a grouping of functions defined around the same common goal and which possibly vertically span one or more several architectural layers.

Note that in this text the Service provider is actually a High Level Service Provider, offering high level services (Video on demand, VoIP, conference services, etc.). It is not mandatory the owner of the network and transport resources, but may uses such capabilities hired from the network owners named here Network Providers (NP). This approach defines a flexible business model. In practice the same commercial entity can play several roles (e.g. CP+SP+ NP), but they can be as well, separated.

We propose a new framework, for management of the resources necessary for connectivity services management in overlay VCANs built over multi-domain, multi-provider IP networks. The VCANs are created and managed by a CAN Provider (CANP), at the request of high level Services Providers which exploit these networks to the benefits of their individual users. These requests are actually "provisioning actions" in the sense that the SP have some forecasted traffic and services data on the future needs and decides to construct some new "networks", for the future.

The traffic and services forecast is not in the scope of this paper. However our solution is neither a static provisioning nor an over-provisioning one; the VCANs can be established, modified in terms of their capabilities and terminated dynamically, given 1) the support of several negotiation protocols existing between the managers (i.e dynamic SLAs (Service Level Agreements) can be established any time, by negotiation); 2) an integrated monitoring system exist covering all environments, capable to offer measurement data on traffic load and thus permitting to the managers to take appropriate decision about resource (re)allocation for different VCANs.

The CANP offers to SPs enhanced connectivity services including unicast, multicast and peer-to-peer (P2P). The management framework is based on vertical and horizontal SLAs defined in the Management and Control (M&C) Plane negotiated and concluded between providers and possibly also on content/service description information (metadata) inserted in the media flow packets (Data Plane) by the servers.

The paper continues the starting work on VCAN management presented in [8][10][11].

Note that this complex system is still under development, therfore some final and evaluation results will be presented in future papers. This paper is organized as follows. Section II presents samples of related work. Section III summarizes the overall ALICANTE architecture. Section IV presents the content awareness features of the system and QoS assurance solutions. Section V describes the peering approach to extend a VCAN over several domains. The proposed CAN management architecture and functionalities is presented in Section VI. Section VII discusses the scalability aspects of the system. Section VIII contains some conclusions and future work outline.

II. RELATED WORK

A higher coupling between the Application and Network layers was recently proposed as a new approach in order to make the IP network more adapted to content and services. In the framework of rethinking the architecture of the Future Internet, the concepts of CAN and NAA are proposed. CAN adjusts network layer processing based on limited examination of the nature of the content, and NAA implies processing the content based on limited understanding of the network conditions. The work presented in [1] emphasizes the strong orientation of the Future Internet (FI) towards content and services and shows the importance of management. CAN/NAA can offer a way of evolution of networks beyond IP, as presented in [6]. The implementation of such an approach can be supported by virtualization as a strong method to overcome the ossification of the current Internet [2][3][4][5].

The work in [12] discusses the content adaptation issues in the FI as a component of CAN/NAA approach. The CAN/NAA approach can also offer QoE (Quality of Experience) and QoS capabilities of the future networks, [6] [13]. Context awareness is added to content awareness in [14]. However, the CAN approach requires a higher amount of packet header processing, similar to deep packet inspection techniques. The CAN/NAA approach can also help to solve the current networking problems related to the P2P traffic overload of the global Internet [15]. The Application Layer Traffic Optimization (ALTO) problem studied by the IETF can be solved by the cooperation between the CAN layer and the upper layer. The management architecture of the CAN/NAA oriented networks is still an open research issue.

Virtualization, including its management and control is seen as a key method in the FI, to increase the flexibility and collaboration capabilities among network and SPs. The challenges are to develop:

- Virtual networks creation, abstracting the subset of network resources (link bandwidth, element processing power, etc.). Parallel logical slices can be defined, based on mechanisms independent or dependent on technology [16][24]. Virtualization based on overlays have been proposed in [3][4][5][21][25].
- Flexible management to create virtual network services on-demand (e.g., security, content-awareness) offered to upper layers, i.e., in [25], by defining a VNet Provider and VNet Operator. Such entities provide the VNet planning / advertising/ discovery/offering, negotiation, provisioning, operation (installation, modification, manipulation, monitoring, termination) while cooperating with IP network layer, [16][23][24][25].
- Support for VNets across multiple network domains based on inter-domain peering conforming to certain SLA/SLSs (Service Level Agreement/ Service Level Specifications), while preserving each domain's resource management independency [16][23][24][25][26]. Inter-domain QoS-enabled routing based on Virtualization is proposed [21][22][30].
- Support of unicast and multicast services on top of the virtual networks. The CURLING [23] architecture is content-centric using a multicast-style receiver-driven service model, but does not address content adaptation, mapping to native IP multicast, or QoS. In [27] a support for multicast streams adapted to each terminal's needs is proposed, by encoding media in multiple Scalable Video Coding (SVC) layers, and defining independent multicast trees for each layer, but it only supports overlay multicast.

Multi-domain Network Resource Management and QoS Support: there are limitations of existing work that are related to management and control. The *Management and Service-aware Networking Architectures* (MANA) Group [28] evaluated several issues either not yet solved, or having limitations. Among them, one can identify some issues related to the area of CAN/Network Environment:

- Guaranteeing availability of service according to Service Level Agreements (SLAs) and high-level objectives; facilities to support Quality of Service (QoS) and SLAs;

- Mobility of services;
- Facilities for the large scale provisioning and deployment of both services and management; support for higher integration between services and networks;
- Facilities for the addition of new functionality, capability for activating a new service on-demand, network functionality, or protocol (i.e., addressing the ossification bottleneck);
- Support of security, reliability, robustness, context, service support, orchestration and management for communication and services' resources.
- Multi-domain QoS support: the [16][17][24][26] projects are examples of architectures supporting end-to-end QoS across multiple domains. However, they do not specifically address media content.
- Dynamic assignment, provisioning and interfacing of customizable multi-domain network services to upper layers (e.g. SPs): this challenge is tackled in [16][24][26]. However, the aforementioned works do not address the cross-layer optimisation between the network layer and upper layers.

Specific comparisons between previous work presented in various research project are given below. The ALICANTE approach for network management versus other research project solutions is compared (The list is not exhaustive). Correlating their scope with ALICANTE's objectives, the selected projects' solutions are (partially) media and content oriented, including end-to-end QoS, and consider multi-provider, multi-domain, multi-technology architectures; they also cover (partially) the integrated management of both high-level services and networking resources. The scope and limitations of the proposed solutions are identified, in order to clarify the ALICANTE design choices and/or progress with respect of these solutions.

The FP6 project MESCAL "Management of End-to-end Quality of Service Across the Internet at Large" project [17][18][19], proposed an evolutionary, scalable architecture, enabling flexible delivery flows over multi-domains, with QoS. The main actors are: Service Providers (SPs), IP Network Providers (INPs), Physical Connectivity Providers (PCPs) and Customers. MESCAL has a complex management system mainly focused on resource management and traffic engineering (offline and online) intra and inter-domain. It does not have a multimedia orientation as a main design direction.

While applying sophisticated techniques for traffic engineering intra and inter-domain MESCAL has no concept of parallel planes as ALICANTE VCANs. However ALICANTE will use the MESCAL concepts of QoS classes (local, extended, meta-QC) in a multi-domain environment, but fitted to VCAN oriented architecture. ALICANTE proposes a joint algorithm for QoS constrained routing , admission control and resource mapping and reservation, both in inter and intra-domain.

The FP6 project ENTHRONE "End-to-End QoS through Integrated Management of Content, Networks and Terminals" [26][29], proposed an evolutionary architecture on top of IP, to cover an entire Audio/Video (A/V) service distribution chain (content generation, protection, distribution across QoS-enabled heterogeneous networks and delivery at user terminals). ENTHRONE targeted primarily multimedia distribution services.

ALICANTE offer as well as Enthrone QoS enabled paths on top of a multi-domain, but its management at network level is more powerful, being able to create VCAN parallel planes.

The FP6 project AGAVE "A liGhtweight Approach for Viable End-to-end IP-based QoS Services" [16][24], aims to solve the E2E provisioning of QoS-aware services over multi-domain IP networks. The Service Providers (SPs) and the IP Network Provider (INPs) cooperates. The INP offer enhanced connectivity services across multiple domains, by extending a Network Plane (NP) to A Parallel Internet (PI) spanning several domains
In AGAVE the NPs implement local virtual network segments while PIs can be seen as end-to-end "virtual network segments", each PI exposing specific performance characteristics. The NPs are built by specific Traffic Engineering (TE) techniques applied in each INP domain. AGAVE suggests an incremental solution for network virtualization. However, it does not create virtual network segment as slices "for sale" to SPs or peer network providers. It manages the complexity of performing the Connectivity Provisioning Agreements (CPA) concluded between SPs and INPs, aiming to the provisioning and delivering of different types of traffic in multi-domain context. The NP and PI notions are said to be internal to INPs, and their definition and realization, through TE, while the SPs sees only the CPA. The AGAVE authors state that "the definition of NPs and PIs and their engineering are hidden from SPs. AGAVE does not consider content-aware aspects at network level.

The novelty in ALICANTE is that it creates VCANs which are known to SPs. However, ALICANTE benefits from, and extends the AGAVE concepts of PIs, by offering the VCAN as an enhanced equivalent of Network Planes. This is done in the framework of a more complete architecture, of the proposed Media Ecosystem.

This paper further develops previous work of the same group of authors. The work in [8] is only a first description of concepts and high level description of the ALICANTE system architecture, with no details on functional capabilities. The paper [10] is the first approach description of the CAN management architecture. The work in [11] is focused only on QoS aspects of the system. While parts of these works are present or referenced here, this work is a step forward in integrating the various components into the system assembly.

III. ALICANTE SYSTEM ARCHITECTURE

The main concepts and general ALICANTE architecture are defined in [7][8][9]. The business model is defined, composed of traditional SP (Service Provider), CP (Content Provider), NP (Network Provider) and End-Users (EU). A new actor is the CAN Provider (CANP) offering virtual layer connectivity services. A new entity is also defined: Home-Box (HB) - partially managed by the SP, the NP, and the end-user, located at the end-user's premises and gathering content/context-aware and network-aware information. The HB can also act as a CP/SP for other HBs, on behalf of the EUs. Two novel virtual layers exist: the CAN layer for network level packet processing and the HB layer for the actual content delivery, working on top of IP. The virtual CAN routers are called Media-Aware Network Elements (MANE) to emphasize their additional capabilities: content and context - awareness, controlled QoS/QoE, security and monitoring features, etc.

The SE uses information from the CAN layer to enforce NAA procedures, in addition to user context-aware ones [8]. Apart from VCANs provisioning, per flow adaptation can be deployed at both HB and CAN layers, as additional means for QoS, by making use of scalable media resources.

The management and control of the CAN layer is partially distributed; it supports CAN customization as to respond to the upper layer needs, including 1:1, 1:n, and n:m communications, and also allow efficient network resource exploitation. The rich interface between CAN and the upper layer allows cross-layer optimizations interactions, e.g., including offering distance information to HBs to help collaboration in P2P style. At all levels, monitoring is performed in several points of the service distribution chain and feeds the adaptation subsystems with appropriate information, at the HB and CAN Layers. Figure 1 presents a partial view on the ALICANTE architecture, with emphasis on the CAN layer and management interaction. The network contains several Core Network Domains (CN); each of them can be extended up to Autonomous System – (AS), the main idea being an unified management of each domain. Therefore, each domain is supposed to have an Intra-domain Network Resource Manager (IntraNRM), as the authority actually configuring the network nodes. Access Networks (ANs) also exists, connected to the core domains; however the ALICANTE VCANs do not cover the ANs. This design decision has been taken because the heterogeneity of AN technologies in terms of managing and guaranteeing the QoS capabilities. On the other side, from the business point of view, the Access Providers have complete independence on "if" and "how" to control the access network resources. The CAN layer cooperates with HB and SE by offering them CAN services. One CAN Manager (CANMgr) exists for each IP domain to assure the consistency of CAN planning, provisioning, advertisement, offering, negotiation installation and exploitation. However, autonomous CAN-like behavior of the MANE nodes can be also offered in a distributed way by processing individual flows.

The following contracts/interactions of SLA/SLS types performed in the Management and Control Plane and the appropriate interfaces are shown in Figure 1:

SP-CANP(1): the SP requests to CANP to provision/ modify/ terminate new VCANs and the CANP to inform SP about its capabilities; *CANP-NP(2)* - through which the NP offers or commits to offer resources to CANP (this data is topological and capacity-related); *CANP-CANP(3)* - to extend a VCAN upon several NP domains; *Network Interconnection Agreements (NIA) (4)* between the NPs or between NPs and ANPs; these are not new ALICANTE functionalities but are necessary for NP cooperation.

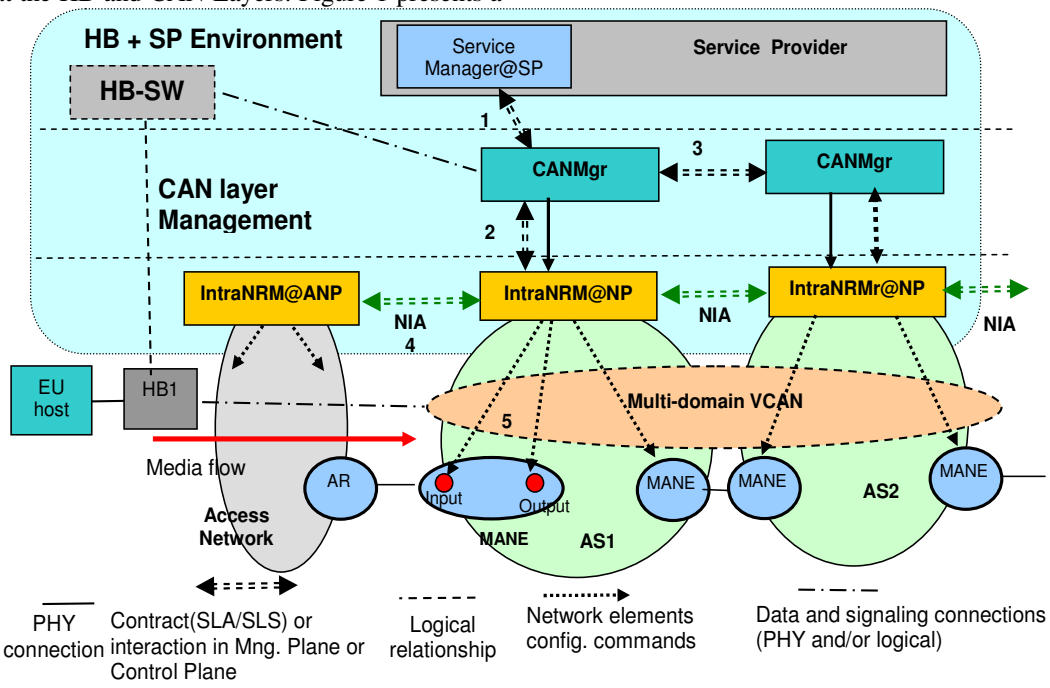


Figure 1. ALICANTE architecture: CAN management interactions

After the SP negotiates a desired VCAN with CANP, it will issue the installation commands to CANP, which in turn configures via IntraNRM (5) the MANE functional blocks (input and output).

IV. CONTENT AWARENESS AND QOS ASSURANCE AT CAN LAYER

The content awareness (CA) is realized in three ways:

- by concluding an SLA between SP and CANP, concerning different VCAN construction. The content servers are instructed by the SP to insert some special Content Aware Transport Information (CATI). This simplifies the media flow classification and treatment by the MANE.
- the SLA is concluded, but no CATI information is inserted in the data packets. The MANE applies deep packet inspection for data flow classification and assignment to VCANs. The treatment of the flows is based on VCANs characteristics defined in the SLA.
- no SLA exists between SP and CANP. No CATI is inserted in the data packets. The treatment of the data flows can still be CA, but conforming to the local policy established at CANP and IntraNRM.

An important issue related to multimedia flow transportation is the QoS assurance. The DiffServ philosophy can be applied to split the sets of flows in QoS classes (QC), with a mapping between the VCANs and the QCs.

Several levels of QoS granularity can be established when defining VCANs. The QoS behavior of each VCAN is established inside the SLA between SP and CANP.

Actually, the CAN layer may offer to the SP, several Parallel Internets (PI), specialized in different types of application content [16]. We adopt the PI concept, enriching it with content awareness. A PI enables end-to-end service differentiation across multiple administrative domains. The PIs can coexist, as parallel logical networks composed of interconnected, per-domain, Network Planes. A given plane is defined to transport traffic flows from services with common connectivity requirements. The traffic delivered within each plane receives differentiated treatment, so that service differentiation across planes is enabled in terms of edge-to-edge QoS, availability and also resilience.

In ALICANTE, generally a one-to-one mapping between a VCAN and a network plane will exist. Specialization of CANs may exist in terms of QoS level of guarantees (weak or strong), QoS granularity, content adaptation procedures, degree of security, etc. A given network plane or VCAN can be realized by the CANP, by combining several processes, while being possible to choose different solutions concerning some dimensions: route determination, data plane forwarding, packet processing, and resource management.

The definitions of local QoS classes (QC) and extended QCs were adopted, to allow us to capture the notion of QoS capabilities across several domains [17][18][19]. For a simplified design, we also used the concept of Meta-QoS-Class [17]. A meta-QC captures a common set of QoS ranges of parameters spanning several domains. It relies on a worldwide common understanding of application QoS needs. For example, VoD service flows need similar QoS characteristics whatever AS they transit. The meta-QC concept offers the advantage that the existence of well known classes greatly simplifies the inter-domain signaling in the sequence of actions needed to establish domain peering in the multi-domain context. This concept simplifies the peering of different domains inside the same VCAN.

The types of VCANs for different QoS granularities based on QCs are described in [9]. In short, the following use cases have been defined for multi-domain VCANs: VCANs based on meta-QC, VCANs based on local QC composition, hierarchical CANs based on local QC composition.

The last case is the most efficient but also the most complex. Each domain may have its local QoS classes. Several local QCs can be combined to form an extended QC. Inside each CAN, several QCs are defined corresponding to platinum, gold, silver, etc. In such a case, the mapping between service flows at SP level and CANs can be done per type of the service: VoD, VoIP, Video-conference, etc.

Note that in ALICANTE architecture, apart from resource provisioning at CAN layer, there is another subsystem, performing per flow adaptation (e.g. for flows generated by Scalable Video Codecs in several layers) [8][9]. This adaptation can adjust the numbers of layers received by a given HB or EU terminals depending on terminal capabilities and network status. For reasons of dimension and focus, this adaptation subsystem is not described in this paper.

V. CAN MULTI-DOMAIN PEERING

A VCAN may span one or several IP domains. In a multi-domain context, one should distinguish between two topologies (in terms of how the domains are linked with each others): *Data Plane Topology* and *Management* and *Control (M&C) topology.* The first can be of any kind, e.g a heterogeneous graph representing a partial mesh (depending on SP needs and including the domains spanned by a given VCAN). The *M&C topology* defines how the CAN Managers associated to different domains inter-communicate for multi-domain VCANs construction. The VCAN initiating CANMgr has to negotiate with other CAN Managers. There exist two main models to organise this communication at management level: *hub* model and *cascade* model [16][17][18][24][26].

A. VCAN Negotiations

The hub model supposes that an initiator VCAN Manager is discussing in hub style with other managers in order to negotiate multi-domain CANs. With this respect, the *CAN Manager is supposed to have inter-domain topology information.* The advantage is that allows a complete control of the VCAN because the CANMng initiating the VCAN knows all network domains participating to this CAN. A drawback is that each CAN Manager should know the complete graph of domain candidates to participate in every possible VCAN, which creates a signaling overhead. However, the number of domains (ASes) involved in a VCAN communication is rather low, given the hierarchical

tiered structure of the Internet [23]. Usually a group of domains of interest for a VCAN are localized in an Internet region, so the scalability problem is not so stringent.

The initiator CAN Manager should discuss/negotiate with all other CAN Managers in order to establish the VCAN = {VCAN1 U VCAN2 U VCAN3 …}, where U represents union action. Split of the SLS parameters should be done at the initiator (e.g. for delay).

Two functional components are needed: (1) inter-domain topology discovery protocol; (2) overlay negotiation protocol for SLA/SLS negotiations between CAN Managers.

The *cascade* model would be more advantageous for initiating CAN Manager if a chain of domains is to form the VCAN [16][26][30]. However, for an arbitrary mesh topology of the NDs composing the VCAN, and for multicast enabled VCAN, this model offers less efficient management capabilities.

Figure 2 shows an example for hub-style signaling adopted in ALICANTE for a multi-domain VCAN. The overall infrastructure is supposed to have four core network domains CNDk, CDNj, CNDn, CNDm, each having a CAN Manager. Several Access Networks are connected to these domains, containing Home-Boxes or/and Content Servers (CS). The latter are controlled by the Content Provider (CP). The SP is requesting a *CANMgr_k* to construct a VCAN, spanning several domains, e.g CNDk, CNDj, CNDn, and CNDm. It is supposed that SP knows the edge points of this VCAN, i.e. the MANEs where different sets of HB are (currently) or will be connected.

The *CANMgr_k* determines (based on its inter-domain topology knowledge) that the components of the VCANs are CNDn, CNDj, CNDm. Therefore, it negotiates SLSss in actions represented by 2.1, 2.2 and 2.3 notations. The negotiations target to achieve appropriate VCAN capabilities from *CANMgr_j CANMgr_n* and respectively *CANMgr_m.* Each CANMgr has to check in its domain if sufficient resources are available (by negotiating with Intra-NRM and concluding an SLS). These actions are not represented in Figure 2. In a successful scenario, the multi-domain VCAN is agreed on (logical resource reservation only) and then it is installed in the network upon request of the SP and executed by CANMGr_k (at its turn it requests this to *CANMgr_j*, *CANMgr_n* and *CANMgr_m).* Then, each CANMgr instructs its associated Intra-NRM to install the appropriate configurations in the edge MANE routers and interior core routers.

B. *Overlay Virtual Inter-domain Topology*

The problem leading to consideration of the inter-domain topology comes from the following needs:

- a multi-domain VCAN should be constructed by the initiator CAN Manager spanning several core network domain CNDs;
- each CND has complete autonomy w.r.t. its network resources including off-line network dimensioning, traffic engineering (TE) and also internal routing. Each CND can assure QoS enabled paths towards some destination network prefixes, by using its own network layer technology like DiffServ, MPLS, etc. and also can control the QoS on its out links. Consequently, each CAN Manager associated to a CND will decide upon accepting or rejecting a proposed SLS for this domain;

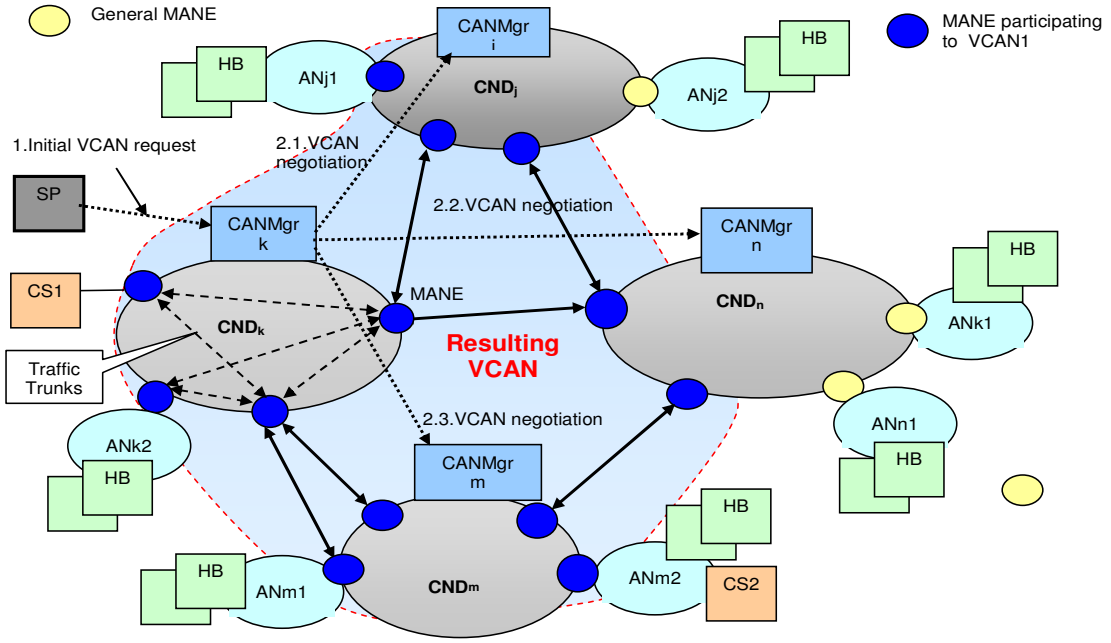


Figure 2. Example of a multi-domain VCAN and hub model for management communication between CAN Managers

- inter-domain QoS- enabled routing should be solved;
- internal topological and capacities information (real or even in abstracted form) of a CND can be non-public for other CNDs;
- the VCAN initiator CAN Manager should determine which CNDs will compose a requested VCAN and split the connectivity requirements among these CNDs, in order to prepare the negotiations described in the previous section. Therefore, the multi-domain VCANs deployment needs knowledge on multi-domain topology and has to solve also a constrained inter-domain routing problem;
- the solution should be scalable, by avoiding one CAN manager be burdened with computations which are related actually to other CND internal business.

The ALICANTE solution is to develop a special service, *Overlay Network Topology Service (ONTS*), able to support multiple VCAN construction while meeting the above constraints. This is explained shortly below and more detailed in Section V.

Note: the subsystem composed by a CAN Manager and its corresponding Intra-NRM will be optionally called CND Manager (CNDMgr). This will simplify the description of overlay topology concepts applied to ALICANTE (without considering the amount of information given by the Intra-NRM to CAN Manager about its topology and capacities).

Each CNDMgr has at least an abstract view of its network and output links towards neighbors, in a form of a set of virtual pipes (called *Traffic Trunks*). A set of such pipes can belong to a given QoS class. Each multiple domain VCAN should also belong to some QoS class and therefore inter-domain QoS aware routing information is necessary in order to construct this VCAN, i.e. to establish SLSs, when negotiating the multi VCAN. Usage of the standard *Border Gateway Protocol (BGP)* to provide knowledge on inter-domain paths would require no modifications in the edge routers, however no QoS information is carried in BGP advertisements. Therefore the establishment of the SLSs between CANMgrs, tried on BGP-indicated routes, might have high probability of failure.

A solution which better fits the ALICANTE purposes is to determine an inter-domain Overlay Network Topology (ONT) by developing a special ONT Service (ONTS), running at the level of CAN Managers. This is partially similar to those described in [21][22], while abstracting the physical network details of each CND. The ONTS delivers to the CAN Manager the information on the inter-domain graph linking different CNDs in a zone and capacities of the inter-domain links. Using the information on this topology (which abstracts the domains), the initiator CAN Manager can determine which domains could compose a requested VCAN. Then, one can apply an inter-domain QoS-aware and constrained routing algorithm to find inter-domain paths satisfying the SLS constraints.

Actually, the determination of the inter-domain ONT can be split in two parts: (1) determination of the inter-domain connectivity graph; (2) determination of the capacities of the inter-domain links. The algorithms and mechanisms to determine the ONT constitute the subjects of other work.

VI. CAN Resource Management Architecture at Service Provider and CAN Provider

Figure 4 presents the proposed architecture for CAN Management. This is a continuation and development of the one presented in [11]. At the Service Manager SM@SP level, the CAN Network Resources Manager (CAN_RMgr) component performs all the actions needed to assure the CAN support to the SP, in order to deploy its high level services in unicast or multicast mode. It is responsible to negotiate with CANP on behalf of the SP and to perform all actions necessary for VCAN planning, VCAN provisioning and VCAN operation.

CNMgr@CANP performs, at the CAN layer, all actions related to VCAN provisioning and operation. The two entities interact based on the SLA contract initiated by the SP. The technical part of an SLA contract is the Service Level Specification (SLS).

Several points of view should be considered when defining/planning the services, planning the CAN and respectively when defining CAN_RMgr functionalities: the commercial optimization needs of the SP, CANP resources, CAN network engineering and implementation.

A. CAN Management at Service Provider

The CAN_RMgr@SP interacts with the following modules supposed to exist and belonging to the SM:

Service Forecast and Planning - an *offline process* performing service predictions and their associated plans of deployment, considering the business as input.

Service Deployment Policy - can contain (in a data base) predefined rules for service planning. This information is derived from the high-level business interests of the SP and significantly influences the planning.

CAN_RMgr@SM contains the following functional blocks: CAN Planning, CAN Provisioning and CAN Operation and Maintenance, as main functional blocks. A CAN Repository data base keeps all data related to VCAN provisioning, installation and current status. Policies can intervene to guide the other blocks through the module *CAN Deployment and Operation Policies*.

Figure 5 also shows the interfaces, defined below. Where possible, the interface implementation will be based on SOAP/Web Services, used for SOAP requests and responses.

1. CAN Planning at *CAN_RMgr@SM* - to - *Service Forecast and Planning@SM* at Service Life Cycle block. This input interface to CAN_RMgr delivers information from the service forecast module and from the policy block, to allow the high level CAN Planning.

2. CAN Operation and Maintenance at *CAN_RMgr@SM* - to - *Service Life Cycle* block. This interface delivers the current status data on active CANs to the Service Life Cycle block.

3. *CAN_RMgr@SM* – to – *CAN Manager@CANP*. This is a multiple interface necessary for CAN_RMgr at SM@SP to perform the following:

- request the CAN Manager VCANs, and to this aim it performs negotiation (SLS contracts will be concluded for VCAN subscription, based on a negotiation protocol);
- command VCAN installation (invocation)
- receive advertisement information about available VCANs constructed at the CANP's initiative
- request modification and/or termination of a VCAN: according to the current situation and the evolution of the forecast, the SP can re-negotiate the network resources with CANP, which will imply to add/modify/delete VCANs;
- receive status and monitoring information about the active VCANs.

B. CAN Provisioning at Service Provider

The functional block for this is the CAN Provisioning Manager at SM@SP. The *CANProvMng@SM* has several main functions shortly presented below.

It performs all *sp_CANpSLS* processing - subscription (unicast/multicast mode) in order to assure the VCAN transport infrastructure for the SP. For VCAN subscription, the *CANProvMng@SM* receives requests for a *sp_CANpSLS* contract dedicated to a given VCAN from *CAN Planning*. Then, it requests to the CAN Manager associated with its home domain, to subscribe for a new CAN. It negotiates the subscription and concludes an SLS denoted by: *SP-CAN_SLS-uni_sub* for unicast, or *SP-CAN_SLS-mc_sub for* multicast. The results of the contract are stored in the *CAN repository*. Note that CAN subscription only means a logical resource reservation at the CAN layer, not real resource allocation and network node configuration.

The CAN subscription action may or may not be successful, depending on the amount of resources demanded by the SP and the available resources in the network. Note that at its turn the CAN Manager has to negotiate the CAN subscription with IntraNRM, and overbooking is an option, depending on the SP policy.

C. Negotiation Protocol

This section will define the specifications for a general SLA/SLS negotiation service and protocol, *AL-SLA/SLS-NP*, valid for several ALICANTE usages and actor pairs. Negotiation protocols must be available at the interfaces: CANMgr – SP to establish SP-CANP SLA; CANMgr – CANMgr to negotiate CAN extension to other NP domains). Negotiation is also needed between CANP and Intra-NRM to negotiate resource commitments by IntraNRM. The *AL-SLA/SLS* is partially a new protocol, i.e. the service primitives and negotiation styles are designed for ALICANTE purposes. This protocol will be implemented over Web Services framework.

The *AL-SLA/SLS-NP* runs at the subscription time to negotiate an agreement between two parties: a customer (requesting the SLA/SLS) and a provider (offering the SLA/SLS). The negotiation can also happen in practice at service invocation periods, provided that subscriptions are immediately followed by invocations. The qualitative and quantitative parameters of the SLS can be specified in a special data structure, known as the *Service Subscription Data Structure (SSDS).* This is to be defined for different usages of the protocol, depending on the type of the SLS required.

The *AL-SLA/SLS-NP* has features of a general negotiation protocol. It has enhanced/new features if compared to other negotiation protocols like SrNP (MESCAL, [18][19]) while being adapted to the ALICANTE environment. It is a client-server half-duplex negotiation protocol between two entities. The cases considered in ALICANTE are:

(1) Service Provider = client, CANP = server- for VCAN contracts

(2) CANP = client, Intra-NRM = server- for contracting the VCAN network resources of core network domain

(3) CANP = client, other CANP = server - for contracting the VCAN resources from other domains (the negotiating entities are CAN Managers and they might belong to the same CANP)

(4) HB = client, Service Provider = server – for individual contracts between HB and SP in order to access media services form SP.

In a particular negotiation session, one party can only be a client or a server but not both at the same time. Concerning the reliability and security of the services offered by the *AL-SLA/SLS-NP*, several choices have been considered: UDP fast transport, having the drawback of non-reliability, or reliable and secure negotiation service offered to the negotiation entities. Given the importance of this signaling in ALICANTE, a reliable and secure negotiation service has been adopted by implementation on top of Web services.

The following assumptions (these can be considered also as detail design decisions) are valid: the parties (Negotiation Logic – NL modules) are the "users" of the protocol. They know the identity of each other; the objects under negotiation can be described as a document whose syntax and semantics is known by the NLs; the NLs know to build, extract, and manipulate the information in the document; the negotiation objective is to conclude a contract between the parties regarding the document content (negotiation is performed upon the values of the parameters in the document and not upon the types of these parameters); AAA processing related to negotiation aspects is not performed by *AL-SLA/SLS-NP* (it concerns the NL); *AL-SLA/SLS-NP* uses the services of a reliable and secure transport protocol; the *AL-SLA/SLS-NP* is transparent to the policy used by NL to make decisions on the negotiated objects (therefore the NL logic complexity is irrelevant for the negotiation protocol).

D. Negotiation Functional Blocks

AL-SLA/SLS-NP is a *transactional protocol* (1-to-1), between two negotiation service interfaces (AL-SLS-NP/NL). Figure 3 shows the negotiation functional architecture. It is seen that the NL can have several active transactions in the same time interval. Figure 3 also presents the generic interfaces involved in negotiations.

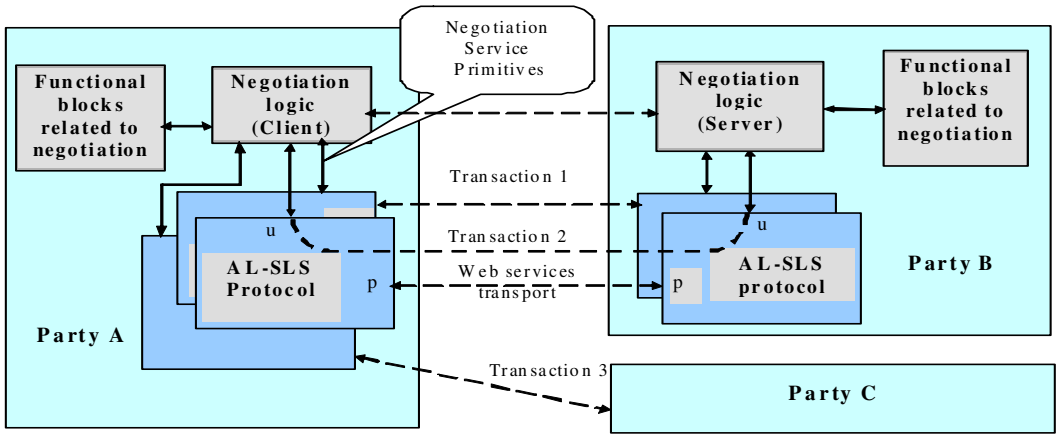


Figure 3 AL-SLA/SLS negotiation architecture

Note: AL-SLS is a short notation for *AL-SLA/SLA-NP* protocol.

AL-SLA/SLS-NP is an *application-layer*, *negotiation session/transaction-oriented* protocol. Each transaction allows to establish/modify/delete an SLS related contract (agreement). It performs contract establishment/modification or termination session. For the negotiation itself, several styles may be applied: simple two steps negotiation (one negotiation object); multiple steps negotiation (one negotiation object). An advanced feature could be: multiple steps negotiation with several negotiation objects in responses.

E. CAN Management at CAN Provider

The Functional architecture of the CAN Management and Control along with CAN Resource Management at CANP is illustrated in Figure 4.

a) Static CAN Services Management

The static CAN services management means that the VCANs containing aggregated multi-domain pipes are subscribed in advance (statically) to actual data transfer, based on SP forecasted data. The VCANs spans from CS locations (known) up to the regions of the access routers in the AN where potential HBs are located, based on a non frequent planning, at SP initiative; the network dimensioning is done infrequently in so called Resource Provisioning Cycles; the multicast trees are established statically over multiple domains. The above functions might be policy based influenced.

b) Dynamic CAN Services Management

The dynamic characteristics of the CAN service management are related to: policy based SLS dynamic invocation handling; possibility of modifying the VCAN invocation; multiple invocation per the same subscription; inter-domain dynamic resource optimization; multicast trees dynamically adjusted at the edges; cooperation between adaptation and provisioning.

c) CAN Manager Main Functions

The main functions of the CAN Manager are: VCANs resources planning, negotiation and provisioning (at SP request or at its own initiative); VCANs advertisement, offering; VCANs installation and exploitation. The mapping CANMgr – per NP domain – allows the horizontal architectural structuring. A single CANMgr can control one or several VCANs deployed in its domain or, initiate the construction of a multi-domain VCAN.

One CAN Manager (CANMgr) is associated to each core IP domain This per-domain mapping exhibits important technical and business advantages: (i) allows for a horizontal structuring of the architecture and creates the possibility of horizontal negotiations between CAN Managers; (ii) creates the possibility that CAN Manager SW to be an extension of the Intra-NRM; (iii) enables each NP to become a CAN Provider; (iv) limits the network area controlled by a single CAN Manager, thus contributing to the scalability of the solution, as the management and control overhead is concerned.

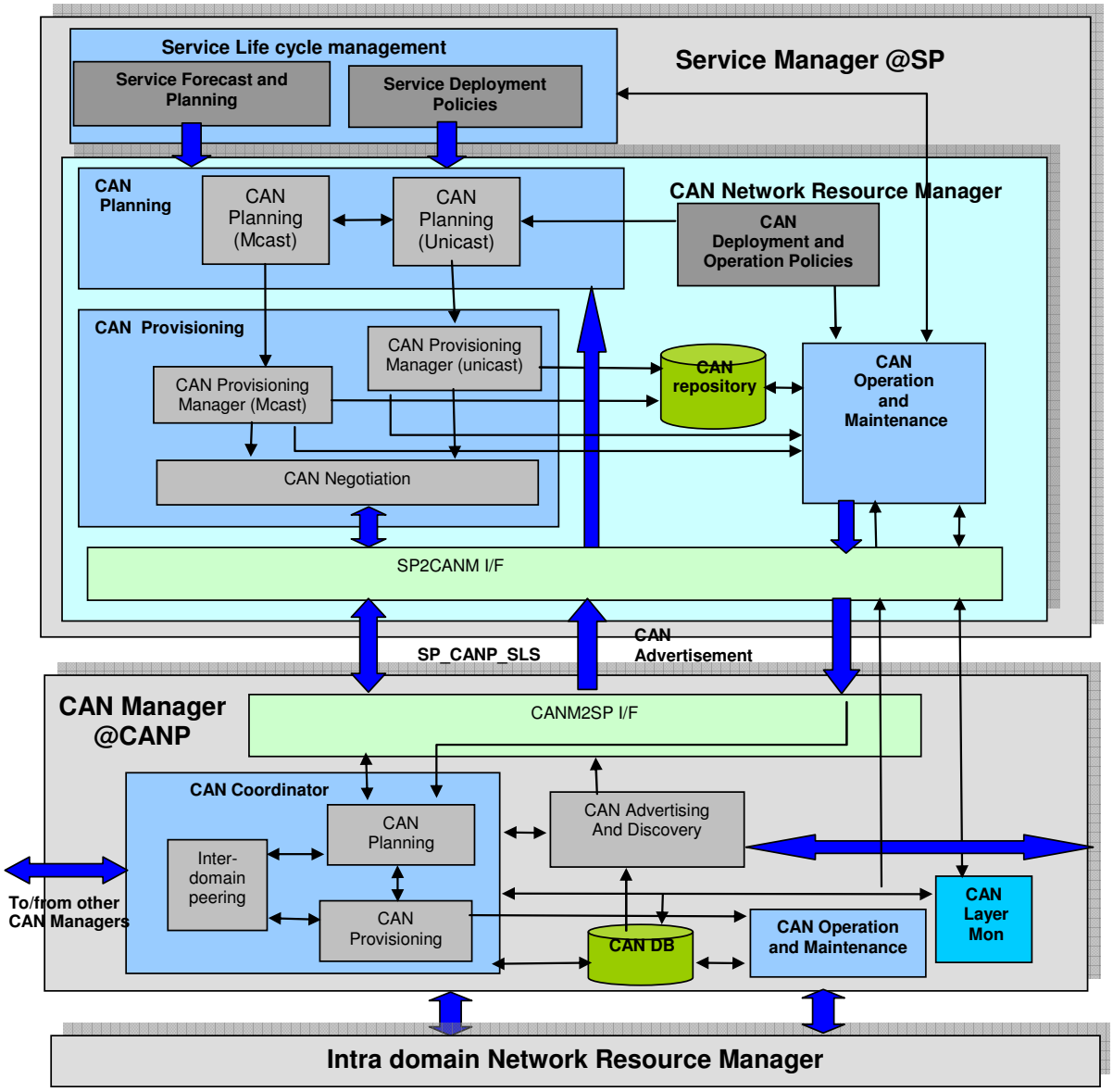


Figure 4. Architecture of the CAN Network Resource Manager at SP and CAN Manager at CAN layer

d) CAN Management Layer External Interfaces

- **North (Upper) I/Fs**

CAN Provider – Service Provider: This I/F of CAN Manager to CAN Network Resource Manager at SP (component of the Service Manager) has the role to assure cooperation between SP and CANP in order to negotiate VCANs (at SP initiative) and install, maintain and exploit, modify, and terminate the VCANs. The second role of this I/F is to assure vertical communication for the cross-layer monitoring framework. Moreover, this I/F will support advanced features like VCAN advertisements made by the CANP to SPs, in order to offer them the existing VCAN resources.

- **Horizontal I/Fs**

CAN Manager – Home Box: this I/F has the role to instruct HBs how to use the VCANs and also to deliver to HBs network distances (based on static and dynamic monitored information) between different edge MANEs.

- **South (Lower) I/Fs**

CAN Manager- Intra-domain Network Resource Manager: this I/F has the role to support negotiation between CANP and NP related to assuring network resources for VCANs and to transport the messages to control the installation and operation of the VCANs; to exchange of monitoring information between the Monitoring module at

network layer and Monitoring module at CAN layer; to assure the transport of flow adaptation policies from CAN Manager to the MANEs via Intra-NRM.

f) Horizontal Interfaces between CAN Managers

The following I/Fs are defined between the CAN Managers associated to different Core Network Domains: negotiation I/F for SLSs between the CAN Managers in order to extend the VCANs over multi-domains; I/F to exchange messages for inter-domain peering,

g) CAN Manager Functional Modules

- CAN Planning, Provisioning and Security
  - *CAN Planning:* this is the highest level decision block at CANMgr, determining what, when and where the VCANs are to be constructed and controls all VCAN life cycle for unicast/multicast VCANs. It cooperates vertically with SP and horizontally with other CAN Managers. Internally to CAN Manager, the CAN Planning instructs the CAN Provisioning block about VCAN negotiations and subsequent related actions.
  - *CAN Provisioning:* performs the lower level actions related to VCAN life cycle by preparing the individual SLSs and conducts horizontal negotiations with other CAN Managers and also with local Intra-NRM.
  - *CAN Security:* controls and performs authentication and authorization functions at CAN Layer including the relationship with SP, and manages the security related policy distribution. The security architecture and functions will not be discussed in this paper
- *Inter-domain Peering:* determines the inter-domain topology and capacities by using an overlay topology service developed among the CAN Managers. This service gets the Overlay Network Topology (ONT), in cooperation with other CAN Managers and then is used by the CAN Planning in order to determine which domains can belong to a given VCAN to be constructed.
- *CAN Operation and Maintenance (CAN_OM):* commands the VCANs installation upon request of the CAN Operation and Maintenance @SP (if the VCAN has been ordered by SP) or from the CAN Provisioning (if the VCAN has been ordered by local CANMgr). This is called *CAN invocation*. The installation actions are implemented as commands given by CAN_OM to the Intra-NRM of the local domain and also to other CAN Managers (horizontally) through the path: CAN_OM –> CAN_Prov –> other CAN Managers. The CAN_OM controls the modification and termination of the VCAN. CAN_OM controls the Monitoring operations at CAN layer related to the SLSs associated to a given VCAN or to the discovery of the Network Distance when requested by the HB.
- *CAN Layer Monitoring:* performs measurements at CAN layer, conforming to the instructions given by the CAN_OM; returns reports on traffic measured and stores them in the CAN DB; communicates with the upper layer of monitoring.
- *CAN Data Base:* contains all data on static and dynamic information related to the CAN Layer. CAN Manager modules read and write information in this database which is the main component through which all other functional blocks interface.
- *Advanced functionalities*:
  - *CAN Policies:* local policies defined at level of this CAN Manager will be defined and guide the VCAN planning and deployment.
  - *CAN advertisement and discovery:* informs horizontally other CAN Managers about existing VCANs and respectively discover other CAN Managers VCAN resources.

F. Basic Signaling for VCAN Resource Provisioning

Figure 5 shows the signaling diagram at CAN layer, in order to construct a multi-domain VCAN, spanning three core network domains CND1, 2, 3. The picture presents a case of successful establishment of a SLS and finally the installation of the VCAN in the network. This is considered as step 0 in a sequence of steps during VCAN cycle. The other steps are not presented in this paper.

The messages exchanged are generically described without details on parameters. The initiator of this VCAN construction is SP which issues a request to CANMgr1.

The latter determine the other CAN Managers involved, i.e. associated to other domains, splits the SLS in particular SLSs particular to each domain involved (these details are not shown in the diagram) and then negotiates with them. In the example, CANMgr2 and CANMgr3 are dialogue partners for CANMgr1. Each CANMgr at its turn negotiates resources with its associated Intra-NRM. Finally, in a success scenario, the SP receives a confirmation about VCAN resource reservation, via VCAN_rsp_neg (ok). Later, at SP will the VCAN is installed in the network by the respective CAN Mangers and Intra-NRMs.

G. CAN Planning at CAN Provider

Before performing VCAN signaling with other CAN Managers, the VCAN initiator CAN Manager should perform the VCAN planning, done by the CAN Planning functional block. Details on the planning algorithm will be presented in another work. Here a summary is presented. The objectives of this planning are: 1. to determine the domains participating to a given VCAN requested by SP; 2. inter-domain (links) resource management; 3. apply a constrained routing algorithm based on ONT acquired by the Initiator CAN Manager; 4. based on routing information, the SLS splitting between domains is computed.

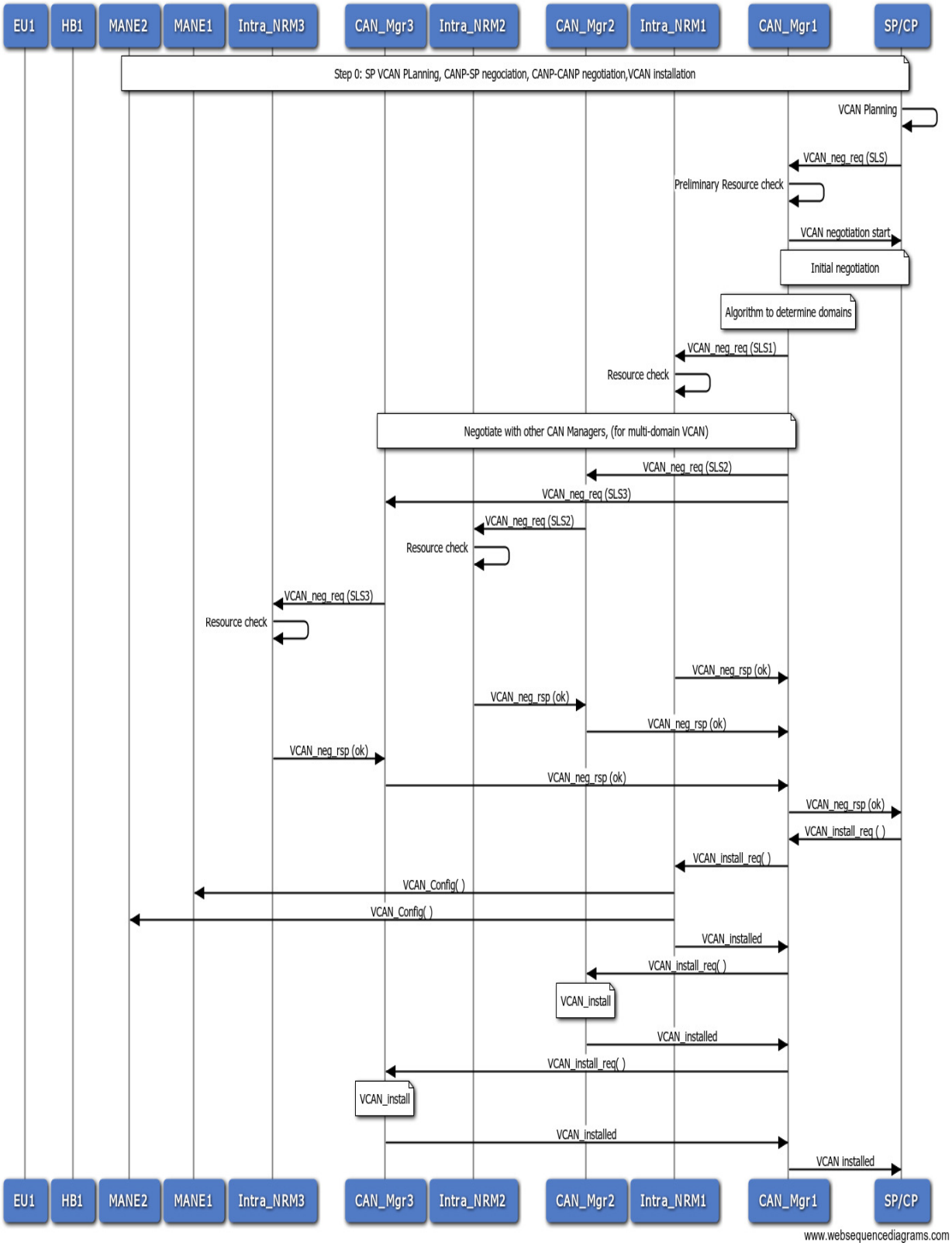


Figure 5. Basic signaling diagram for VCAN establishment (multiple network domain case)

Inter-domain planning

A summary of actions set is the following:

1. SP issues a VCAN-0 request (this will be mapped onto a given QoS class), i.e. an SLS request (topology, traffic matrix, QoS guarantees, etc.)
2. The initiator CANMgr obtains from ONTS (this service is assured by the Inter-domain Peering block) the inter-domain level ONT (topology graph, inter-domain link capacities, etc.). The ONT is sufficiently rich to cover the required VCAN.
3. The initiator CANMgr determines the involved domains in VCAN by using the border ingress-egress point's knowledge (actually MANE addresses) indicated in the SLS parameters
4. The initiator CANMgr determines a contiguous inter-domain connectivity graph (each CND is abstracted as a node) resulting in an extended VCAN-1. In VCAN-1 graph, some additional transit core network domains need be included (it is supposed in the most simple version that these new core network domains added are also VCAN capable). Therefore a contiguous new VCAN-1 is defined. Optimization techniques can be applied in this phase.
5. The initiator CAN Manager should make the first split of the initial SLS among core network domains. This means to produce the set of SLS parameters valid to be requested to each individual CND. The inputs: are: ONT graph, abstracting each CND (domain) by a node; QoS characteristics of the inter-domain links (bandwidth, delay); Traffic Matrix (and other QoS information) of the SLS proposed by SP. The outputs are the Traffic matrices for each CND composing the VCAN.

In order to perform this, the Initiator CAN Manager will run a constrained routing (modified Dijkstra algorithm) based on a composite additive QoS-aware metric. This will be described in a future work. Finally the CAN Planning has determined the sets of SLS parameters to be negotiated with each CAN Manager of the domains participating to VCAN.

H. Intra-domain CAN Resources Provisioning

An essential functional aspect related to the VCAN mapping onto network resources in a core network domain is the relationship between CANMgr and Intra-NRM with respect to:

(1) the style for Intra-NRM to upload information to CANMgr about its available resources : on demand (OD) or in proactive (P) style (at Intra-NRM initiative);

(2) amount and depth of information uploaded by Intra-NRM on network resources (graph, capacities, etc.).

Note that for every variant, and depending on monitoring information at network level the Resource Availability Matrix (RAM) uploaded can be adjusted by Intra-NRM to improve the traffic engineering performances.

These variants are shortly discussed below.

Proactive style*:* At initiative of Intra-NRM, (periodically or event triggered) the RAM, i.e., either full connectivity graph and capacities or only a summary similar to ONT information is uploaded to CANMgr. Advantages are that the Intra-NRM is the most qualified to know when it is appropriate to deliver network information to CANMgr, e.g. every time when network re-dimensioning is performed. Also, CANMgr has at every moment all information about network resources.

The disadvantage is that CANMgr can be overloaded with more information than it really needs at a given time; it may keep or discard some information, depending on its local policy at CANMgr level.

On demand style: the RAM of the Intra-NRM is obtained on demand of the CANMgr when it needs it, in order to appropriately answer SP requests. The advantages are that the CANMgr decides when it wants RAM information from Intra-NRM, and it is a possibly better usage of CANMgr DB space. Another plus is that Intra-NRM is released from informing the CANMgr.

The cons are that this approach incurs a higher delay in servicing the SP requests, because CANMgr should first acquire RAM in order to respond appropriately based on updated RAM information.

In the real networks world the NPs are usually reluctant to disclose information on their networks (topologies, traffic load, etc.) to external parties. However the approach of this paper supposes a strong cooperation betwen the CAN Procvider (CANP) and Network Provider (NP) in order to construct the VCANs. Therefore, several degrees of "trust" between such entities should be analysed.

The depth of information uploaded by the Intra-NRM to CAN Manager depends on the degree of trust between these two entities and is of course, policy determined. We might have several situations:

- High trust (HT): Intra-NRM uploads to CANMgr its full connectivity graph;
- Medium trust (MT): Intra-NRM uploads to CANMgr an overlay RAM based on traffic trunks (similar to ONT);
- Low trust (LT): Intra-NRM does not upload/disclose any topology and resources to CANMgr, but only ingress-egress points Ids and Yes/No answers to a SLS request

Depending on the actual routing and mapping algorithm (to map the matrix traffic requested for this domain), the real graph will be placed at the level of CAN Manger (case HT) or Intra-NRM (case MT or LT).

From the architectural and also business point of view, the MT and LT solutions are more appropriate, in order not to outsource the important task of configuring the network elements to third parties. In such a case, the CAN Manager has only to decide on mapping of the Traffic Matrix onto TTs reported by the Intra-NRM. Decision P/OD can be an implementation option. The solution HT is actually one in which the CAN Manager functionalities are constructed as an additional software on top of the IntraNRM, and in such a case one has a single integrated entity (IntraNRM +

CANMgr) belonging to the business actor which is now (NP + CANP).

VII. SCALABILITY ASPECTS OF MANAGEMENT AND CONTROL

The ALICANTE system targets large network configurations. Scalability in such cases is important and is shortly discussed in this section, with focus on M&C.

A. VCAN Planning and Provisioning

The following features assure a good scalability of the architectural solution:

- The full centralized solution for VCAN management is avoided, given that each Core Network Domain has associated a CAN Manager; the initiator CAN Manager should negotiate in hub style with other CAN Mangers. However, this approach does not create difficult scalability problems, given that the number of domains actually involved in a multi-domain chain is rather small (less than 10) due to tiered structure of the Internet. On the other side these signaling are not real time ones. The advantage of this solution is that the initiator CAN Manager has always an overall image of a multi-domain VCAN and can respond to some SP possible complaints about different events. No per-flow signaling between CAN Managers exist in M&C.
- The VCAN SP–CANP negotiation are done per VCAN, described in terms of aggregated traffic trunks
- The SP negotiates its VCAN(s) with a single CAN Manager, irrespective if it wants a single or a multi-domain spanned VCAN
- A hierarchical overlay solution is applied for inter-domain peering and routing, where each CAN Manager knows its inter-domain connections. The CAN Manager initiating a multi-domain VCAN is the coordinator of this hierarchy, without having to know details on each domain VCAN resources
- The monitoring at CAN layer and network layer will be performed at an aggregated level.

B. Multicast Management and Control

- The management system described work as well for unicast or multicast capable VCANs. However, the multicast detailed management is not described in this paper, given that the general signaling actions are the same in unicast or multicast case. Multicast hybrid solution has been envisaged, with usage of IP level multicast intra-domain wherever is possible;
- VCAN multicast capable, i.e., multicast aggregated trees can be constructed, usable by multicast sessions having similar QoS characteristics;
- The multicast solution is combined with P2P (used by the HBs), thus assuring a better scalability for multicast distribution.

C. Routing and Forwarding

In the case of multi-domain VCANs, the broadest paths will be selected, thus optimizing the network resource usage. The length of the paths can be minimized by using higher layer tiers domains when necessary.

The length of the paths between HBs working in P2P mode, or between HBs and CSs, will be minimized by delivering network distances information to HBs to help the peering process.

D. Management of Configurable Types of VCANs

The amount of processing in the Data Plane affects the scalability of the system. In order to be flexible with respect to different SP needs, and not to reach a very rich granularity if no need for it exists, the CAN layer may offer several types of VCANs seen as parallel planes. The M&C can configure the VCANs (at request of the SP), to offer gradual scalability properties and QoS differentiation capabilities:

- VCANs based on Meta-QoS-Classes – mostly scalable (lower processing tasks for the data flows) but with rough granularity in terms of VCAN QoS properties
- Multi-domain VCANs based on an inter-domain combination of local (per-domain) QoS classes (LQC) – having medium scalability and higher degree of service/flows differentiation
- Multi-domain hierarchical VCANs based on local QC composition, but where each domain may have its local QoS classes.

VIII. CONCLUSIONS AND FUTURE WORK

This paper proposed an architectural solution for connectivity services management in Content Aware Networks for a multi-domain and multi-provider environment. The management is based on vertical and horizontal SLAs negotiated and concluded between providers (SP, CANP, NP), the result being a set of parallel VCANs offering different classes of services to multimedia flows, based on CAN/NAA concepts. The approach is to map the QoS classes on virtual data CANs, thus obtaining several parallel QoS planes. The system can be incrementally built by enhancing the edge routers functionalities with content awareness features. Further work is going on to design and implement the system in the framework of the FP7 research project ALICANTE. A preliminary implementation and performance evaluation of the main network element (MANE router) supposed to be managed by the described framework of this papaer appeared in [20].

Future work is also necessary to solve the mapping of the overlay VCANs (as requested by SP) onto real network resources in a multi-domain context, while satisfying QoS constraints. The VCAN resources are first logically reserved; later when installation is requested by the SP, they will be really allocated in routers.

Finally use cases (Video on demand, IPTV media streaming will be experimented on four testbeds (Portugal, Bordeaux, Bucharest, Beijing) in order to to validate the

overall functionality. These results will be presented in future papers.

ACKNOWLEDGMENTS

This work was supported partially by the EC in the context of the ALICANTE project (FP7-ICT-248652) and partially by the project POSDRU/89/1.5/S/62557.

REFERENCES

[1] Schönwälder, J., Fouquet, M., Dreo Rodosek, G., and Hochstatter, I.C., "Future Internet = Content + Services + Management", IEEE Communications Magazine, vol. 47, no. 7, Jul. 2009, pp. 27-33.

[2] Baladrón, C., "User-Centric Future Internet and Telecommunication Services", in: G. Tselentis, et. al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 217-226.

[3] Turner, J. and Taylor, D., "Diversifying the Internet," Proc. GLOBECOM '05, vol. 2, St. Louis, USA, Nov./Dec. 2005, pp. 760-765

[4] Anderson, T., Peterson, L., Shenker, S., and Turner, J., "Overcoming the Internet Impasse through Virtualization", Computer, vol. 38, no. 4, Apr. 2005, pp. 34–41.

[5] Chowdhury, N. M. and Boutaba, R., "Network Virtualization: State of the Art and Research Challenges", IEEE Communications Magazine, vol. 47, no.7, Jul. 2009, pp. 20-26.

[6] Kourlas, T., "The Evolution of Networks beyond IP", IEC Newsletter, vol. 1, Mar. 2007. Available at http://www.iec.org/newsletter/march07_1/broadband_1.html (last accessed: Mar. 2010).

[7] FP7 ICT project, "MediA Ecosystem Deployment Through Ubiquitous Content-Aware Network Environments", ALICANTE, No248652, http://www.ict-alicante.eu/ (last accessed: Dec. 2010).

[8] Borcoci, E., Negru, D. and Timmerer, C., "A Novel Architecture for Multimedia Distribution based on Content-Aware Networking" Proc. of. CTRQ 2010, Athens, June 2010, pp. 162-168

[9] ALICANTE, Deliverable D2.1, ALICANTE Overall System and Components Definition and Specifications, http://www.ict-alicante.eu, Sept. 2010

[10] Borcoci, E. and Iorga, R., "A Management Architecture for a Multi-domain Content-Aware Network" TEMU 2010, July 2010, Crete.

[11] Borcoci, E., Stanciu, M., Niculescu, D., and Xilouris, G., "Quality of Services Assurance for Multimedia Flows based on Content-Aware Networking", CTRQ2011, April 2011 - Budapest, Hungary

[12] Zahariadis, T., Lamy-Bergot, C., Schierl, T., Grüneberg, K., Celetto, L., and Timmerer, C., "Content Adaptation Issues in the Future Internet", in: G. Tselentis, et al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 283-292.

[13] Liberal, F., Fajardo, J.O., and Koumaras, H., "QoE and *-awareness in the Future Internet", in: G. Tselentis, et al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 293-302.

[14] Baker, N., "Context-Aware Systems and Implications for Future Internet", in: G. Tselentis et. al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 335-344.

[15] Aggarwal, V., Feldmann, A., "Can ISPs and P2P Users Cooperate for Improved Performance?", ACM SIGCOMM Computer Communication Review, vol. 37, no. 3, Jul. 2007, pp. 29-40.

[16] Boucadair, M. et al., "A Framework for End-to-End Service Differentiation: Network Planes and Parallel Internets", IEEE Communications Magazine, Sept. 2007, pp. 134-143

[17] Levis, P., Boucadair, M., Morrand, P., and Trimitzios, P., "The Meta-QoS-Class Concept: a Step Towards Global QoS Interdomain Services", Proc. of IEEE SoftCOM, Oct. 2004.

[18] Howarth, M.P. et al., "Provisioning for Interdomain Quality of Service: the MESCAL Approach", IEEE Communications Magazine, June 2005, pp. 129-137

[19] MESCAL D1.2: "Initial Specification of Protocols and Algorithms for Inter-domain SLS Management and Traffic Engineering for QoS-based IP Service Delivery and their Test Requirements", January 2004, www.mescal.org (last accessed: Dec 2010)

[20] Niculescu, D., Stanciu, M., Vochin, M., Borcoci, E., Zotos, N., Implementation of a Media Aware Network Element for Content Aware Networks, CTRQ 2011, April 2011 - Budapest, Hungary

[21] F. Verdi, M. F. Magalhaes, "Using Virtualization to Provide Interdomain QoS-enabled Routing", Journal of Networks, April 2007, pp. 23-32.

[22] Zhi Li, P. Mohapatra, "QRON: QoS-Aware Routing in Overlay Networks", IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 22, NO. 1, January 2004, pp.29-39.

[23] W. K. Chai, N. Wang, I. Psaras G. Pavlou, C. Wang, G. García de Blas, F.J. Ramon Salguero, L. Liang, S. Spirou, A. Beben, E. Hadjioannou, "CURLING: Content-Ubiquitous Resolution and Delivery Infrastructure for Next-Generation Services", IEEE Communications Magazine, March 2011, pp.112-120.

[24] M. Boucadair (ed) et al., AGAVE Public Deliverable "D1.1: Parallel Internets Framework", September 2006, www.ist-agave.org.

[25] 4WARD, "A clean-slate approach for Future Internet," http://www.4ward-project.eu/.

[26] ENTHRONE, End-to-End QoS through Integrated Management of Content, Networks and Terminals, FP6 project, available on-line: http://www.ist-enthrone.org/.

[27] Open ContEnt Aware Networks (OCEAN), http://www.ict-ocean.eu/.

[28] A. Galis et al., "Management and Service-aware Networking Architectures (MANA) for Future Internet Position Paper: System Functions, Capabilities and Requirements", http://www.future-internet.eu/home/future-internet-assembly/prague-may-2009

[29] T.Ahmed, A. Asgari, A.Mehaoua, E. Borcoci, L.B Équille, K. Georgios "End-to-end quality of service provisioning through an integrated management system for multimedia content delivery" Computer Communications, Special Issue: Emerging Middleware for Next Generation Networks, Volume 30, Issue 3, 2 February 2007, Pages 638-651.

[30] Z. Wang and J. Crowcroft, "Quality-of-service routing for supporting multimedia applications", *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 7, pp. 1228—1234, 1996.

LTE Performance Evaluation Based on two Scheduling Models

LTE downlink and uplink analysis

Oana Iosif
Faculty of Electronics, Telecommunications and Information Technology
"Politehnica" University of Bucharest
Romania
e-mail: oana_iosif@yahoo.com

Ion Bănică
Faculty of Electronics, Telecommunications and Information Technology
"Politehnica" University of Bucharest
Romania
e-mail: banica@comm.pub.ro

***Abstract*—This paper presents a detailed analysis on the Long Term Evolution performance in both downlink and uplink directions emphasizing the most important aspects that influence the performance indicators. Round Robin and Weighted Round Robin scheduling strategies, in time domain and time-frequency domain, are used in different scenarios concerning antenna configuration, number of users and types of services in order to evaluate cell throughput, average user throughput and cell capacity. The control channels bring some limitations in the number of users served and on the actual transmission bandwidth when time-frequency domain packet scheduling is implemented and all these are reflected in the simulation results. This paper offers an image of the LTE network performance in various scenarios, the most important aspect being the cell capacity evaluation with a certain minimum or expected service throughput.**

Keywords – LTE; OFDMA; SC-FDMA; scheduling; control channel; Round Robin.

I. INTRODUCTION

In the context of a continuous mobile traffic growth along with the high requirements of users and operators, 3GPP (3rd Generation Partnership Project) has standardized a new technology called Long Term Evolution (LTE) as the next step of the current 3G/HSPA (High Speed Packet Access) networks to meet the needs of future broadband cellular communications. It may be considered as a milestone towards 4G (Fourth Generation) standardization. The requirements set for LTE that are specified in [1] envisage high peak data rates, low latency, increased spectral efficiency, scalable bandwidth, optimized performance for mobile speed, etc. In order to fulfill this extensive range of requirements several key technologies have been considered for LTE radio interface of which the most important are: multiple-access through Orthogonal Frequency Division Multiple Access (OFDMA) in downlink and Single Carrier - Frequency Division Multiple Access (SC-FDMA) in uplink and multiple-antenna technology.

Packet Scheduling is one of LTE Radio Resource Management (RRM) functions, responsible for allocating resources to the users and, when making the scheduling decisions, it may take into account the channel quality information from the user terminals (UE), the QoS (Quality of service) requirements, the buffer status, the interference situation, etc. [2]. Like in HSPA or WiMAX, the scheduling algorithm used is not specified in the standard and it is eNodeB (Evolved NodeB) vendor specific.

The LTE downlink has been previously analyzed in several papers like [3], [4], [5] and [6]. The authors evaluated the system and/or user throughput and the fairness of the scheduling algorithms used in their simulations, but the work was restricted either to SISO (Single Input Single Output) antenna technology, or the users experiencing the same radio conditions. Very few papers considered the PDCCH (Physical Downlink Control Channel) limitation in the number of users served and the terminal category impact. For LTE uplink there are fewer papers, some examples being [7], [8] and [9]. As for downlink, the control channels limitation is scarcely mentioned and evaluated and none of them analyzes the priority set for a specific type of users and its impact on cell capacity and throughput.

In this paper, we evaluate the performance of packet scheduling in downlink and uplink LTE using the Round Robin and Weighted Round Robin strategies through the results obtained for the average cell throughput, the achieved user throughput and the system capacity. These results may be considered in the LTE network design, in order to approximate the number of users that can be served with a certain throughput in a commercial LTE network.

The remainder of this paper is organized as follows. Section II discusses several aspects on scheduling and assigned resources in downlink LTE system followed by an insight on resource allocation in LTE uplink presented in Section III. Section IV describes the Round Robin and Weighted Round Robin scheduling models used in the simulations and Section V depicts the results of the simulated scenarios. The conclusions are driven in Section VI.

II. SEVERAL ASPECTS ON RESOURCE ALLOCATION IN LTE DOWNLINK

The LTE downlink is mainly characterized by OFDMA as multiple access scheme and MIMO (Multiple Input Multiple Output) technology. The benefit of deploying OFDMA technology on downlink LTE is the ability of allocating capacity on both time and frequency, allowing

multiple users to be scheduled at a time. The minimum resource that can be assigned to a user consists of two Physical Resource Blocks (PRBs) and it is known as chunk or simply Resource Block (RB) [2],[10]. In downlink LTE one PRB is mapped on 12 subcarriers (180 kHz) and 7 OFDM symbols (0.5 ms) and this is true for non-MBSFN (Multimedia Broadcast multicast service Single Frequency Network) LTE systems and for normal CP (Cyclic Prefix). Scheduling decisions can be made each TTI (Time Transmission Interval) that in LTE is equal to 1 ms.

For non-real time services dynamic scheduling is usually used as it provides flexible and even full utilization of the resource. This scheduler performs scheduling decisions every TTI by allocating RBs to the users, as well as transmission parameters including modulation and coding scheme. The latter is referred to as link adaptation. The allocated RBs and the selected modulation and coding scheme are signaled to the scheduled users on the PDCCH (Physical Downlink Control Channel). The dynamic packet scheduler also interacts closely with the HARQ (Hybrid Automatic Repeat Request) manager as it is responsible for scheduling retransmissions and it may also take into account the QoS attributes and buffer information [6], [11].

The schedulers in the eNodeB may or may not take into consideration the channel information when making scheduling decisions. An alternative to channel-dependent scheduling is Round Robin strategy that serves the users in cyclic order, regardless the channel information.

Although OFDMA technology allows the users to be multiplexed in time and frequency, the scheduler, according to the implemented algorithm, may choose to allocate the entire bandwidth to a single user, reducing the scheduling to be done only in time domain. The channel-sensitive scheduling done in time domain only is called Non-Frequency Selective Scheduling (NFSS) and the scheduling exploiting the channel variations in both time and frequency is known as Frequency Selective Scheduling (FSS) as specified in [12]. Fig. 1 illustrates an example of FSS for two users [6], [13].

When scheduling is done in time and frequency domain, independently if it is channel-aware or not, the number of multiplexed users in each TTI is limited by the number of

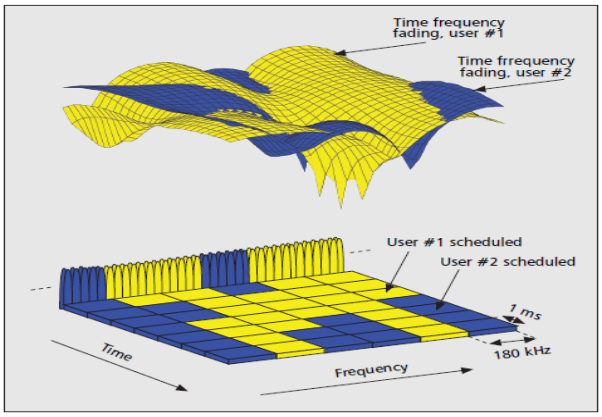


Figure 1. Frequency selective scheduling illustration for two users in downlink LTE

PDCCHs that can be configured. This depends on the system bandwidth, the number of symbols signaled for PDCCH allocation, the PDCCH format number, etc. [10], [11], [14], [15].

The PDCCHs are intended to provide both uplink and downlink scheduling information and in the assumption of half of the users making downlink transmissions, the maximum number of scheduled users per TTI in downlink LTE is half of the number of PDCCHs available. The authors from [11] discussed this constraint and proposed a three-step packet scheduling algorithm as it is depicted in Fig. 2 [11].

The highest number of PDCCHs is obtained with PDCCH format 0 (excellent radio conditions), but in real scenarios there will be a mix of PDCCH formats in order to realize link adaptation [11].

From all the multiple antenna techniques that can be used in downlink LTE the most performance improvements in terms of cell/user throughput and cell capacity are reached with MIMO (Multiple Input Multiple Output). The baseline antenna configuration for MIMO and antenna diversity is two transmit antennas at the cell site and two antennas at the terminal. The higher-order downlink MIMO and antenna diversity (four TX and two or four RX antennas) is also supported. The basic MIMO schemes applicable to the downlink are illustrated in Fig. 3.

These schemes can be applied depending on the scenario (indoor, urban and rural coverage) and the UE capability.

The multi-antenna technology brings a new dimension on mobile radio – SPACE – and its implementation is based on three fundamental principles:

- Diversity gain – Use of the space-diversity provided by the multiple antennas to improve the robustness of the transmission against multipath fading (Fig. 3A).
- Array gain – Concentration of energy in one or more given directions via precoding or beamforming. This also allows multiple users located in different directions to be served simultaneously (so-called multi-user MIMO) (Fig. 3B and Fig. 3D).
- Spatial multiplexing gain – Transmission of multiple signal streams to a single user on multiple spatial layers created by combinations of the available antennas (Fig. 3C) [16].

III. SEVERAL ASPECTS ON RESOURCE ALLOCATION IN LTE UPLINK

The high PAPR (Peak to Average Power Ratio) of the transmitted signal in OFDMA and the limited power of the mobile terminal determined 3GPP to choose a different scheme for LTE uplink – SC-FDMA – in order to optimize the power consumption of mobile handsets.

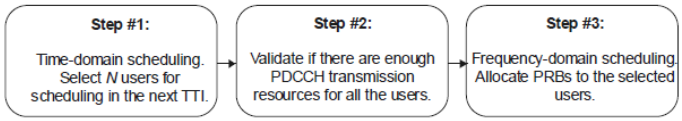


Figure 2. Illustration of a three step scheduling algorithm framework

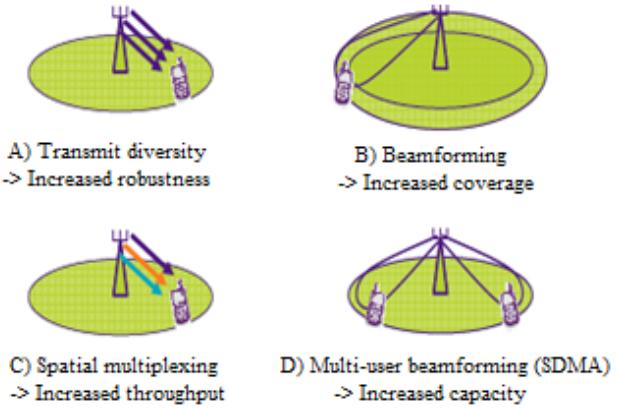


Figure 3. MIMO schemes for LTE downlink

This multiple access technology is a variation of OFDMA, but with initial precoding stage using DFT (Discrete Fourier Transform), which results in each subcarrier carrying a linear combination of data symbols instead of each data symbol being mapped to a separate subcarrier. This results in a single-carrier waveform that exhibits a significantly lower PAPR than OFDMA, but keeps the multipath resistance and the inter-user orthogonality [11].

The smallest resource that can be assigned to a user also consists of two PRBs adjacent in time and for simplicity of expression, in the rest of the paper we will use the term resource block (RB). In uplink LTE one PRB is mapped on 12 subcarriers, each of 15 kHz, and 7 SC-FDMA symbols, with 0.5 ms time duration and this is true for non-MBSFN LTE systems and for normal CP [2], [10]. As well as in downlink, SC-FDMA allows multiple users to be scheduled at a time and the scheduling decisions can be made each TTI.

Unlike OFDMA, SC-FDMA constrains transmission to occur only on adjacent subcarriers in order to maintain its single carrier property. This means that RBs cannot be allocated freely and must be contiguous, limiting both frequency and multi-user diversity.

LTE defines both localized and distributed scheduling in the downlink direction, but only localized scheduling in the uplink direction in order to keep the PAPR small in the SC-FDMA symbols of each user. Fig. 4 compares the localized and the distributed scheduling [17].

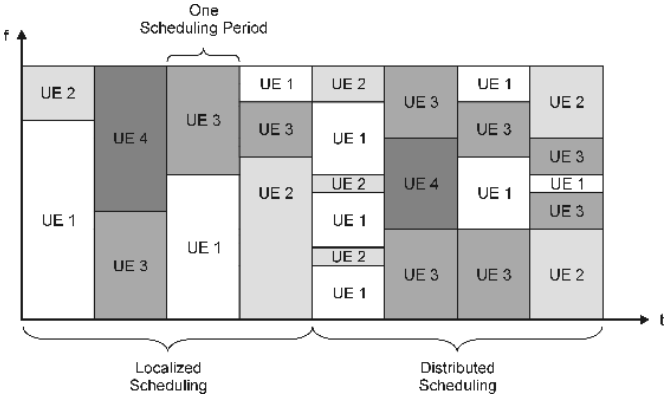


Figure 4. Localized vs. distributed scheduling in LTE

Taking into account that the PDCCH limitation applies also in LTE uplink, the scheduling framework from Fig. 2 can be used in LTE uplink too.

The LTE uplink is more impacted by the control information than the downlink. The actual transmission bandwidth in uplink is limited by the PUCCH (Physical Uplink Control Channel) regions and some typical expected number for different LTE bandwidths are presented in [16] and shown in Table I. PUCCH carries scheduling requests, ACK/NACK information related to downlink data packets, CQI (Channel Quality Information) etc. The number of PUCCH RBs per slot is the same as the number of PUCCH regions per sub-frame.

IV. ROUND ROBIN AND WEIGHTED ROUND ROBIN SCHEDULING MODELS IN LTE

As mentioned in Section II, Round Robin (RR) scheduling is a channel non-aware scheduling scheme that lets users take turns in using the shared resources (time and/or RBs), without taking the instantaneous channel conditions into account. Therefore, it offers great fairness among the users in radio resource assignment, but degrades the system throughput. Weighted Round Robin (WRR) is a variation of RR with priorities defined for different service categories. Time Domain (TD) RR and WRR, as well as Time and Frequency (FD) RR and WRR scheduling models are described in this Section.

A. Time Domain Round Robin and Weighted Round Robin scheduling model

In TD RR the first reached user is served with the whole frequency spectrum for a specific time period (1 TTI), not making use of the information on his channel quality. Then these resources are revoked back and assigned to the next user for another time period. The previously served user is placed at the end of the waiting queue so it can be served in the next round. This algorithm continues in the same manner [18]. Fig. 5 illustrates the resource sharing between two users with TD RR algorithm. The colors and the line orientation make the difference between the users. In this example, every user is allocated 100% of the RBs and 50% of the time resource, so each gets 50% of the global resource [6]. The TD WRR differentiates from TD RR in the number and the type of users served.

Let us suppose a CBR (Constant Bit Rate) service of 500 kbps and a SNR (Signal to Noise Ratio) throughput per RB given by the radio conditions of 1 Mbps. Assuming there is one static user making the service and the same SNR is experienced in each RB and in all TTIs, the maximum amount of data that can be sent during 1 TTI per RB is 1 kb.

TABLE I. TYPICAL NUMBER OF PUCCH REGIONS

Bandwidth (MHz)	Number of 0.5 ms RBs sub-frame	Number of PUCCH regions
1.4	2	1
3	4	2
5	8	4
10	16	8
20	32	16

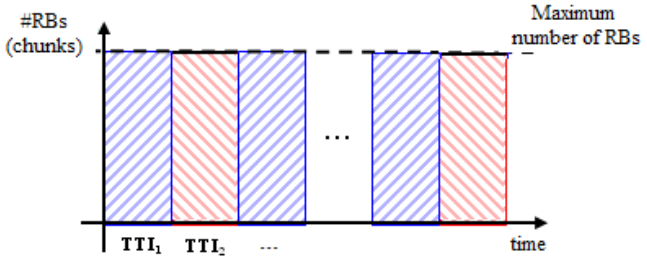


Figure 5. Resource sharing between two users with TD RR

Considering the system bandwidth of 20 MHz, which consists of 100 RBs, the user needs to be allocated all resources for five TTIs to reach his service throughput. Therefore the user must be allocated 1/200 of the total resource in order to be served. This ratio is equal to service throughput / (SNR throughput * total number of RBs given by the system bandwidth). This represents the main idea in the TD RR model.

B. *Time and Frequency Domain Round Robin and Weighted Round Robin scheduling model*

The FD RR allows multiple users to be scheduled within one TTI in cyclic order. Keeping in mind the PDCCH limitation discussed in Section II, the scheduling framework from Fig. 2 can be applied. The TDPS (Time Domain Packet Scheduling) may select N users in RR fashion to be scheduled in one TTI, but the PDCCH resources (M) must be checked in order to see if all users selected by the TDPS can be simultaneously scheduled. M users at most can be the input of FDPS (Frequency Domain Packet Scheduling), which schedules each user with RR strategy across different RBs. In the next TTI the users that were not selected in the previous one will be scheduled in the same manner and so on [6].

The FD RR is briefly presented in [19] where PDCCH constraint is not considered. The authors propose that all users be allocated one RB before reallocating to the same user. If the number of users waiting to be scheduled is less than the number of PDCCHs per TTI, this approach is correct, but only for LTE downlink (as in uplink the RBs must be adjacent). But if the number of users selected within one TTI is greater than the number of configurable PDCCHs and if the idea of allocating one RB to each user is maintained, the result will be a waste of resources [6].

The resource sharing between two users with FD RR, assuming a hypothetical system bandwidth of two RBs, is depicted in Fig. 6. As in Fig. 5, each user is allocated 50% of the global resource.

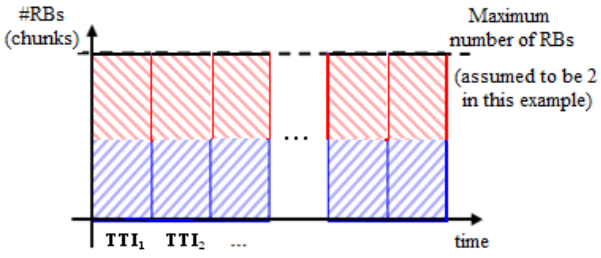


Figure 6. Resource sharing between two users with FD RR

Taking the example given in Section III.A, but considering the limitation of 20 PDCCHs per TTI for downlink LTE as it is concluded from [11], [14] and [15] and 40 users having the same radio conditions and making the same service, one user needs to be allocated 1 RB for 500 TTIs [6]. The global resource in this case is reduced due to PDCCH constraint i.e. the maximum throughput given by the radio conditions * number of PDCCHs. The radio resource ratio assigned to each user is 1/40, higher than in TD RR example, so the capacity will be smaller.

A solution to address this problem would be the allocation of more RBs at once to each user in order to exploit all transmission bandwidth [6].

Knowing that for 20 MHz band in downlink LTE 20 users can be simultaneously scheduled at most, each user can be allocated 5 RBs before assigning resources to another one. In this case, the FD RR cell throughput in LTE downlink will be the same as for TD RR, with the only advantage of being more suited to services with small packets and some delay requirements [6].

The FD RR cell throughput in LTE uplink will be less than that in TD RR due to the limitation in the actual transmission bandwidth brought by PUCCH.

As it was previously mentioned for TD WRR, the FD WRR has an impact on the number and types of users served, but the main principle is that from FD RR.

V. SIMULATION SCENARIOS AND RESULTS

A computer simulation using C++ platform is conducted to evaluate the performance of RR and WRR scheduling in downlink and uplink LTE, based on the mathematical modeling of these scheduling strategies, along with the basic network parameters. For the simulations performed a single cell eNodeB is considered, with a carrier frequency of 2.6 GHz FDD (Frequency Division Duplex) and a system bandwidth of 20 MHz.

Besides SISO (Single Input Single Output) antenna configuration used in [6], in this paper we also consider MIMO 2x2 and we present several simulation results for LTE uplink using SIMO (Single Input Multiple Output) 1x2. Moreover, in several scenarios the users are uniformly distributed in the cell compared to the results presented in [6], which treated the case of all users experimenting the same radio conditions.

In the simulations considering SISO in LTE downlink category 1 terminals are used with ~10 Mbps, in those with SIMO 1x2 in LTE uplink it is assumed that all users have category 3 terminals with ~50 Mbps, while in those with MIMO 2x2 category 3 terminals with ~100 Mbps are chosen.

In order to reduce the complexity of the system simulations, we assume that equal downlink transmit power is allocated on each RB, all transmitted packets are received correctly and the users are static. For LTE uplink scenarios, we also assume that the UE transmit power can sustain the entire bandwidth allocation to a single user during 1 TTI.

The downlink SNR values for SISO case used in this paper, resulting from pathloss, shadow fading, multipath fading, eNodeB transmit power and thermal noise, are listed

in Table II, along with the corresponding modulation and coding schemes and data rates. The downlink SNR values for MIMO 2x2 are listed in Table III and those for SIMO 1x2 for LTE uplink in Table IV.

The following sub-sections present the simulation results for cell throughput, average user throughput and system capacity in downlink and uplink LTE with RR and WRR scheduling models. There are two categories of users considered: the first makes a CBR streaming service (e.g. video streaming) with a certain expected throughput (under this value the users cannot be served) and the second makes a VBR best effort service (e.g. data transfer using File Transfer Protocol) with a defined minimum accepted throughput, but it can reach more. The maximum best effort throughput reached is limited by the minimum between the data rate corresponding to the SNR experienced and the maximum throughput given by the user terminal category.

For all simulation scenarios, the FD RR scheduling model considered is the one with 1 RB allocation to each user before reallocating another one to other user. The reason for this choice stands in emphasizing the PDCCH impact on simultaneously served users that also leads, in certain situations, in cell throughput limitation.

TABLE II. DOWNLINK SNR TO DATA RATE MAPPING FOR SISO

Minimum downlink SNR values (dB)	Modulation and coding scheme	Data rate (kbps)
1.7	QPSK (1/2)	138
3.7	QPSK (2/3)	184
4.5	QPSK (3/4)	207
7.2	16 QAM (1/2)	276
9.5	16 QAM (2/3)	368
10.7	16 QAM (3/4)	414
14.8	64 QAM (2/3)	552
16.1	64 QAM (3/4)	621

TABLE III. DOWNLINK SNR TO DATA RATE MAPPING FOR MIMO 2X2

Minimum downlink SNR values (dB)	Data rate (kbps)
3	207.8
9	383.6
12	518.2
16	734.9
19	898.6
21	992.0
24	1086.0
26	1124.4

TABLE IV. UPLINK SNR TO DATA RATE MAPPING FOR SIMO 1X2

Minimum uplink SNR values (dB)	Data rate (kbps)
1	88.5
3	177.0
6	265.6
8	354.2
10	425.0
12	487.0
14	499.0
17	506.6

A. Cell throughput results for LTE downlink with SISO

These results have been previously presented in [6].

A 2 Mbps expected throughput is chosen for streaming users and the same value is considered as the minimum throughput for best effort users. It is assumed that all users experience the same radio conditions.

Fig. 7 and Fig. 8 show the cell throughput with TD RR and FD RR for streaming users and best effort users.

The dependence of the cell throughput on the SNR values with 30 users in the cell is depicted in Fig. 7. An interesting evolution is shown by the cell throughput in FD RR for streaming service, where the cell saturation is reached. The explanation lies in both PDCCHs limitation of 20 per TTI and the CBR service of 2 Mbps. Despite the PDCCH limitation in FD RR for best effort users, cell saturation is not reached due to their capability of achieving a higher throughput compared to their service throughput. All 30 users are served only in TD RR for the last SNR throughput value.

Considering that the users experience only the last SNR value from Table II, the cell throughput is evaluated with the number of users in the cell trying to reach their service. When comparing TD RR with FD RR based on the results illustrated in Fig. 8 it can be concluded that for best effort users they show the same cell throughput evolution. Despite the PDDCH limitation, the best effort users may achieve a higher throughput than the minimum defined one.

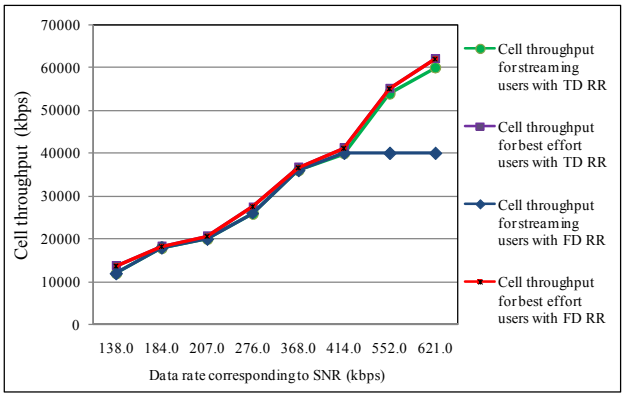


Figure 7. Cell throughput vs. SNR in LTE downlink with SISO for TD RR and FD RR

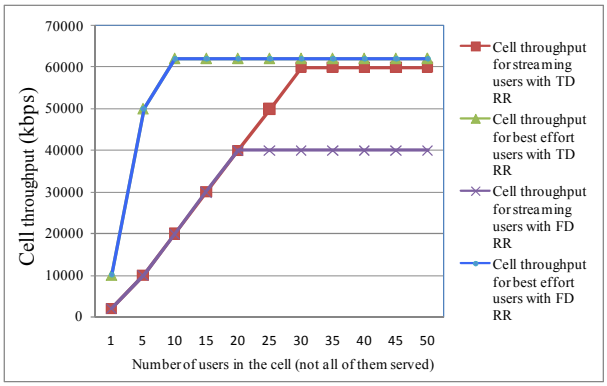


Figure 8. Cell throughput vs. the number of users in the cell in LTE downlink with SISO for TD RR and FD RR

This is not the case for streaming users because in TD RR the cell throughput is higher due to a higher number of users served. From the cell throughput saturation it can also be seen that in TD RR there are 31 streaming users served, while in FD RR only 20 users reach their service requirements (the maximum 20 PDCCHs that can be configured within 1 TTI does not necessarily limits the number of served users in the cell to 20; for a lower expected throughput, the number of users served is more than 20 in FD RR with one 1 RB allocated to each user, as it will be presented in the scenarios concerning MIMO 2x2 in LTE downlink and SIMO 1x2 in LTE uplink).

B. *Average user throughput results for LTE downlink with SISO*

Fig. 9 shows the evolution of average user throughput with the number of users in the cell (experiencing the same radio conditions as in Fig. 8). For streaming service the user throughput is constant at 2 Mbps, while for best effort users it varies until the cell saturation is reached, the saturation point being the maximum number of users served. The maximum best effort user throughput in the case of 1 and 5 users in the cell is limited by the terminal category at 10 Mbps. The achievable best effort user throughput is higher in FD RR than in TD RR for more than 20 users in the cell because there are fewer users served and the cell resource is shared between a smaller number users.

All the results presented so far were obtained considering separately streaming and best effort users, not mixed. The following Section presents the case with traffic mix and cell capacity evaluation.

C. *System capacity results for LTE downlink with SISO*

Fig. 10 and Fig. 11 show for both scheduling strategies how many users are served from the total number of users in the cell and the impact of the priority set for streaming service on the number and types of users scheduled. Half of the users in the cell are best effort users. The cell saturation is reached for 31 users served in TD RR and 20 in FD RR. When no priority is set (TD and FD RR), the number of served streaming users is equal to that of best effort users.

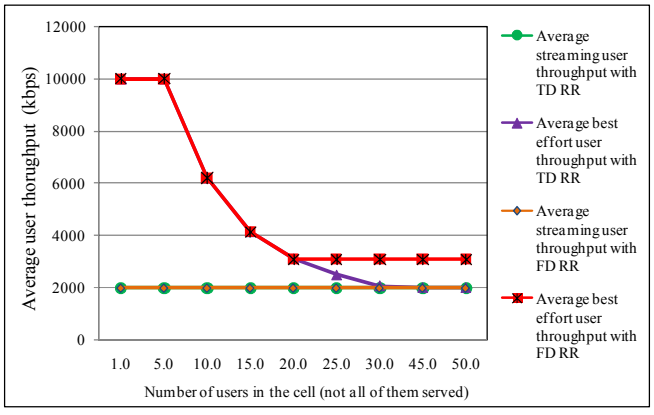


Figure 9. Average user throughput vs. the number of users in the cell in LTE downlink with SISO2 for TD RR and FD RR

For 50 users in the cell, in TD WRR there are 6 best effort users and 25 streaming users served, while in FD WRR there is no best effort user served and 20 streaming users served. The following sub-sections present simulation results that were not included in [6].

D. *Cell throughput results for LTE downlink with MIMO 2x2*

The results presented in sub-sections D, E and F were obtained through simulations of various scenarios considering MIMO 2x2 antenna configuration and 2 Mbps as the expected throughput for streaming users and 500 kbps as the minimum throughput for best effort users. Similar to SISO case, the cell throughput is evaluated for all SNR values from Table II and for several numbers of users in the cell. The dependence of the cell throughput on the SNR values with 50 users in the cell is depicted in Fig. 12. As in SISO scenario, in FD RR for streaming service the maximum cell throughput is limited to a value that in this case is equal to 40 Mbps.

The explanation lies in both PDCCHs limitation of 20 per TTI and the CBR service of 2 Mbps. But there is a major difference between this figure and Fig. 7 regarding the cell throughput in FD RR for best effort users.

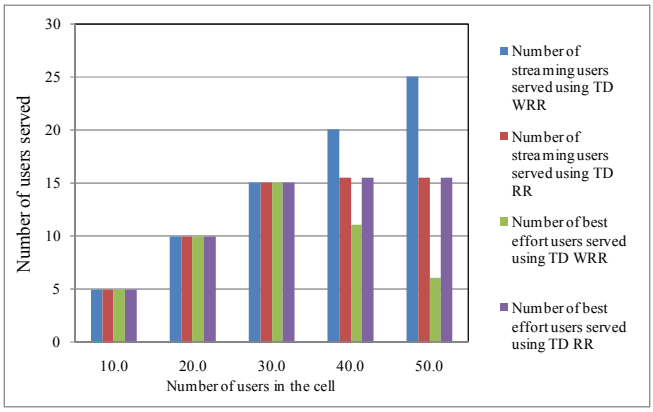


Figure 10. Number of users served vs. number of users in the cell in LTE downlink with SISO for TD RR and TD WRR

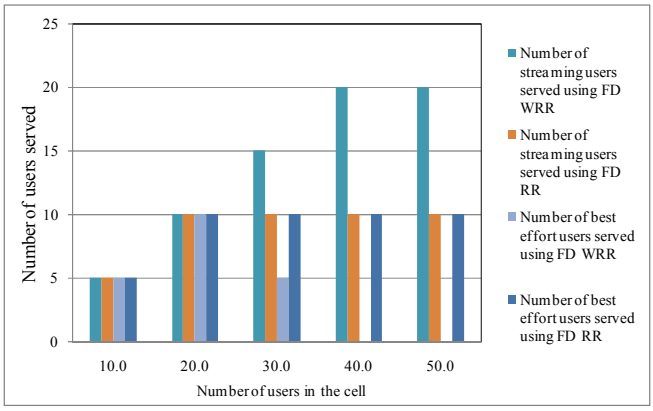


Figure 11. Number of users served vs. number of users in the with SISO for FD RR and FD WRR

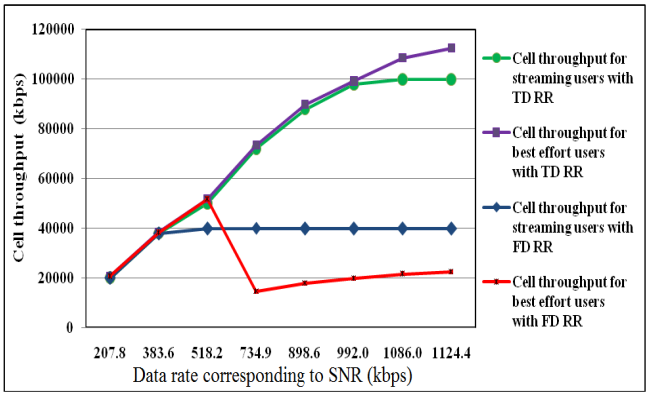


Figure 12. Cell throughput vs. SNR in LTE downlink with MIMO 2x2 for TD RR and FD RR

Because the best effort service requires a lower minimum throughput (500 kbps compared to 2 Mbps from previous scenario), there can be more than 20 users served in the cell for the last 5 SNR values. While in TD RR all 50 streaming and best effort users are served in the case where users experience the best radio conditions of those presented in Table II, in FD RR only 45 best effort users are assigned resources to get the required service. This emphasizes the poor performance of FD RR with 1 RB assigned and imposes the use of FD RR with more RBs assigned (e.g. 5 RBs) that has the same results as TD RR, but is more suited for power limited scenarios, low traffic or services with certain latency requirements.

The cell throughput evolution with the number of users, considering all users in the best radio conditions, is depicted in Fig. 13. Compared to Fig. 8, cell throughput for TD RR and FD RR, in the case of best effort traffic only, does not show the same evolution. This is due to the fact that in this case FD RR strategy allows more than 20 users in the cell to be served (45), thus limiting to 20 the effective number of RBs to be assigned to users every TTI (as 20 MHz bandwidth has 100 RBs and the maximum number of PDCCHs per TTI is 20).

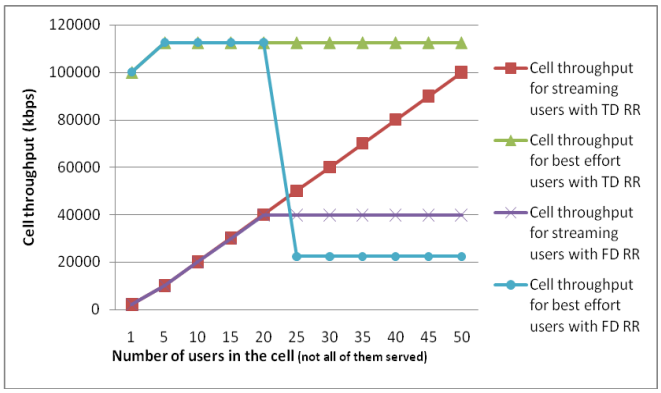


Figure 13. Cell throughput vs. the number of users in the cell in LTE downlink with MIMO 2x2 for TD RR and FD RR for users experiencing the best radio conditions

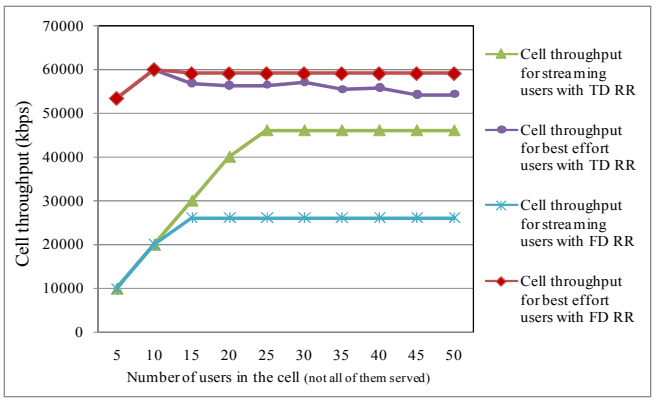


Figure 14. Cell throughput vs. the number of users in the cell in LTE downlink with MIMO 2x2 for TD RR and FD RR for users uniformly distributed in the cell

For streaming traffic only, the cell throughput with TD RR is higher than in FD RR due to a higher number of users served in the first case. In FD RR the maximum cell throughput is limited to 40 Mbps due to PDCCH, while in TD RR the cell throughput reaches 100 Mbps.

The cell throughput evolution with the number of users when the users are uniformly distributed in the cell, thus experiencing different radio conditions, is illustrated in Fig. 14. The number of PDCCHs in this case will be less than 20 per TTI because it will be a mix of PDCCH formats (40% Format 0, 30 % Format 1, 20 % Format 2 and 10% Format 3) [11], not only format 0, as considered so far. It results ~13 PDCCHs per TTI for downlink. Comparing Fig. 14 with Fig. 13, the maximum cell throughput value is the first difference to be noticed. As expected, in the scenario for Fig. 14, which is closer to a real one as different users experience different radio conditions, cell throughput barely exceeds 60 Mbps. And this is the case for best effort users that expect a lower throughput than the streaming users. In the latter case, the cell throughput reaches 46 Mbps, meaning 23 streaming users served. With FD RR, there are less than 20 users accepted that make streaming traffic or best effort traffic. More specifically, in this scenario, with FD RR only 13 streaming users are served vs. 20 in the previous scenario and 13 best effort users vs. 45 are allowed to make the traffic required.

E. Average user throughput results for LTE downlink with MIMO 2x2

The evolution of average user throughput with the number of users in the cell when users experience the best radio conditions is depicted in Fig. 15. For streaming service the user throughput is constant at 2 Mbps (as imposed by streaming service requirements), while for best effort users it varies, and cell saturation is reached for 45 users with FD RR. Comparing TD RR with FD RR in the case of best effort traffic only, besides the fact that with

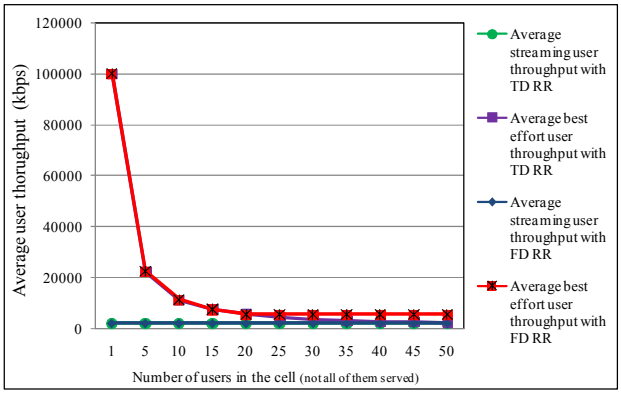


Figure 15. Average user throughput vs. the number of users in the cell in LTE downlink with MIMO 2x2 for TD RR and FD RR for users experiencing the best radio conditions

TD RR all 50 best effort users are served, the best effort user throughput for all 45 users is 2249 kbps with TD RR and 500 kbps with FD RR.

The average user throughput for both TD RR and FD RR with one type of users in the cell (streaming or best effort) when the users are uniformly distributed in the cell is depicted in Fig. 5.16. A comparison between Fig. 16 and Fig. 15 is necessary in order to outline the decrease in average user throughput when the users are uniformly distributed in the cell versus the case where all users were experiencing the best radio conditions. For 5 users in the cell it was obtained ~11 Mbps vs. ~23 Mbps. Similar to the previous case, in FD RR the average user throughput is higher than the one with TD RR (less users served, the cell resources divided between fewer users).

F. *System capacity results for LTE downlink with MIMO 2x2*

Fig. 17 and Fig. 18 show for both scheduling strategies how many users are served from the total number of users in the cell and the impact of the priority set for streaming service on the number and types of users scheduled. Half of the users in the cell are best effort users and the simulation is performed taking the last SNR value from Table II.

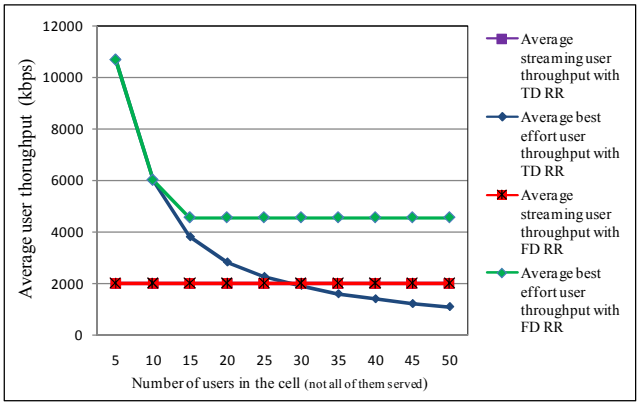


Figure 16. Average user throughput vs. the number of users in the cell in LTE downlink with MIMO 2x2 for TD RR and FD RR for users uniformly distributed in the cell

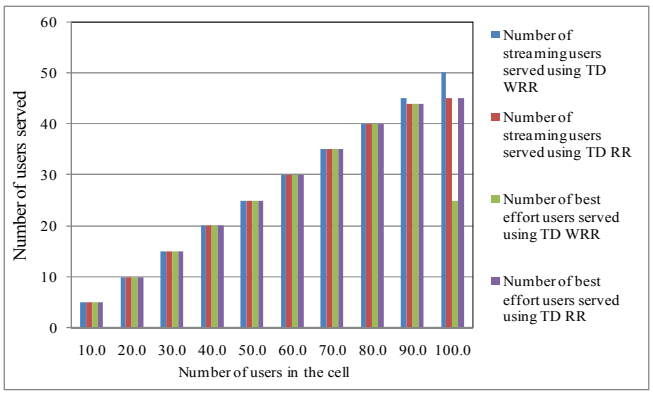


Figure 17. Number of users served vs. number of users in the cell in LTE downlink with MIMO 2x2 for TD RR and TD WRR

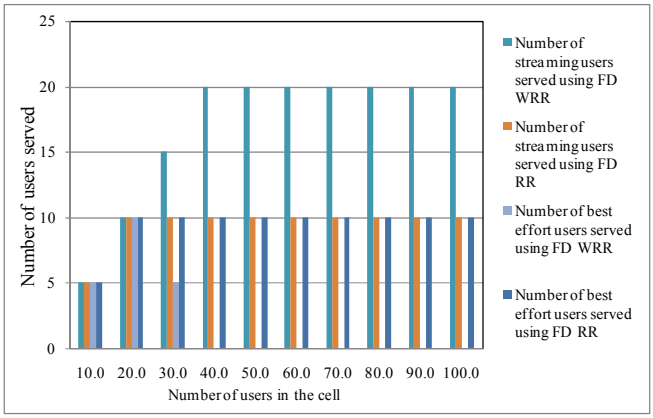


Figure 18. Number of users served vs. number of users in the cell in LTE downlink with MIMO 2x2 for FD RR and FD WRR

When no priority is set (TD RR and FD RR), the number of served streaming users is equal to that of best effort users. For 100 users in the cell and priority set (WRR), in TD WRR there are 25 best effort users and all 50 streaming users served, while in FD WRR there is no best effort user served and 20 streaming users served. These results emphasize the waste of resources generated by FD RR strategy with only 1 RB allocated to each user.

G. *Cell throughput results for LTE uplink with SIMO 1x2*

The following sub-sections present the simulation results for LTE uplink. For uplink performance evaluation, it was chosen a scenario with 1x2 SIMO, 20 MHz system bandwidth, with best effort and streaming users having category 3 terminals (~50 Mbps). For streaming service it is defined a constant throughput of 1 Mbps, while for best effort service a minimum throughput of 200 kbps.

It has to be reminded the uplink control overhead mentioned in Section III and specified in Table I that limits the actual transmission bandwidth. Also, the single-carrier property of uplink transmission cannot be neglected and in order to assure adjacent RBs in FD RR in the scenarios with up to 20 users in the cell, the users are assigned from the start with a several number of RBs (instead of 1 to each user before reassigning to the first one).

The cell throughput evolution with SNR values is shown in Fig. 19. 50 users were considered in the cell trying to reach the service. As for the first two SNR values, only 20 users best effort users can be served in FD RR, the cell throughput values are equal to those in TD RR (the cell resources are fully utilized). For the other SNR values, there can be more best effort users served in FD RR, but due to the minimum throughput of 200 Kbps abd the PDDCH limitation, the transmission bandwidth is limited to 20 RBs (considering that the FD RR with 1 RB allocated to each user). The cell throughput for streaming service in FD RR is limited to 20 Mbps, also due to PDCCH constraint.

Fig. 20 illustrates the cell throughput evolution with the number of users in the cell in the best radio conditions scenario. As in Fig. 13, the FD RR cell throughput for best effort traffic drops when there are more than 20 users in the cell due to the transmission bandwidth limitation to 20 RBs (given by the PDCCH constraint). As expected, the cell throughput with FD RR with streaming users is limited to 20 Mbps (20 users served), while in TD RR 42 streaming users make the required service. The TD RR throughput is higher than the FD RR one when there are more than 20 users in the cell.

H. Average user throughput results for LTE uplink with SIMO 1x2

Fig. 21 illustrates the average user throughput evolution with the number of the users in the cells, the simulation being made with the highest SNR value.

The streaming user throughput was expected to be 1 Mbps, while an interesting evolution is seen in FD RR with best effort traffic: for less than 20 users in the cell, the cell resources are fully utilized and the users get a high throughput, while for more than 20 users the RBs that can be allocated are limited to 20 and in the case of 50 best effort users trying to reach their service, they are all served, but with 203 kbps. With TD RR strategy, all 50 best effort users are served, the minimum service throughput acquired being 851 kbps.

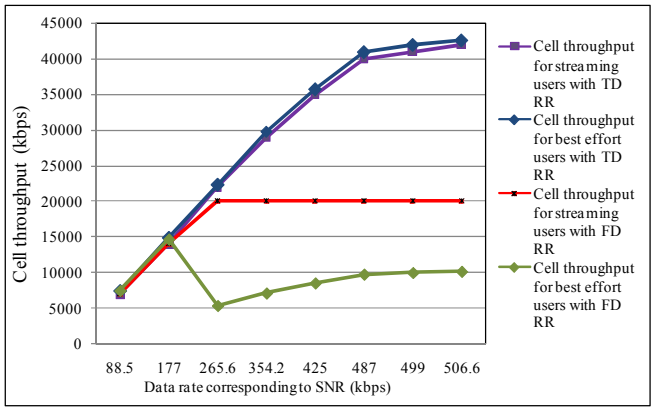


Figure 19. Cell throughput vs. SNR in LTE uplink with SIMO 1x2 for TD RR and FD RR

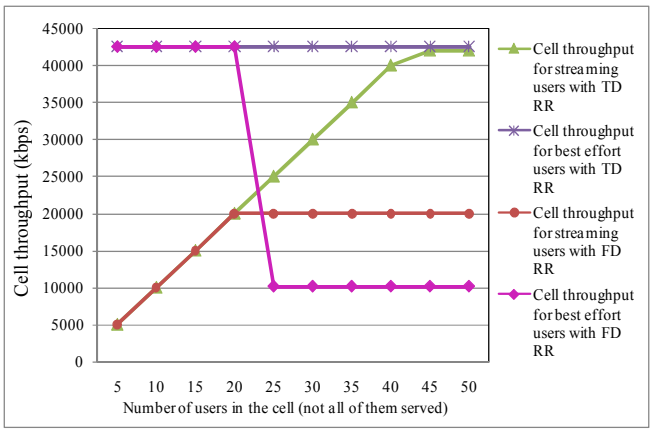


Figure 20. Cell throughput vs. the number of users in the cell in LTE uplink with SIMO 1x2 for TD RR and FD RR

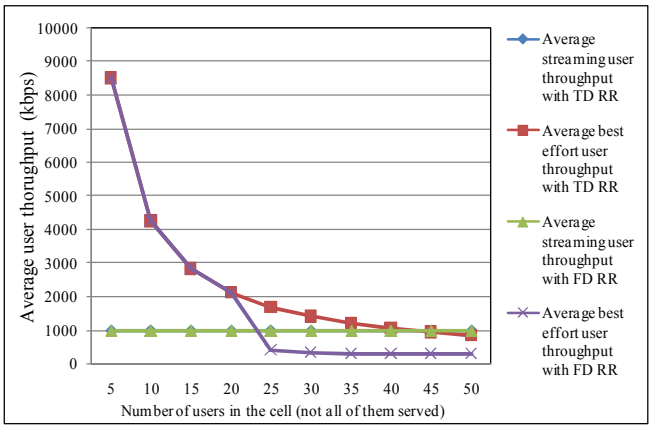


Figure 21. Average user throughput vs. the number of users in the cell in LTE uplink with SIMO 1x2 for TD RR and FD RR

I. System capacity results for LTE uplink with SIMO 1x2

All the previous results for LTE uplink have been obtained considering, in turn, streaming and best effort users. This Section presents the case with traffic mix and evaluates cell capacity with RR and WRR.

Fig. 22 and Fig. 23 show for RR and WRR scheduling algorithms, in TD and FD, how many users of a certain service category are served from the total number of users in the cell All users are assumed to be experiencing the highest SNR value from Table III. Half of the users in the cell are best effort users.

For TD RR and FD RR the number of served streaming users is equal to that of best effort users. For 80 users in the cell, in TD WRR all 40 streaming users are served, but only 7 best effort users are accepted, while in FD RR all best effort users are rejected. In the case of equal priorities between streaming nd best effort service (TDRR and FD RR) for 80 users in the cell, with TD RR there are 35 streaming and 35 best effort users served, and with FD RR 10 users of each category are rejected.

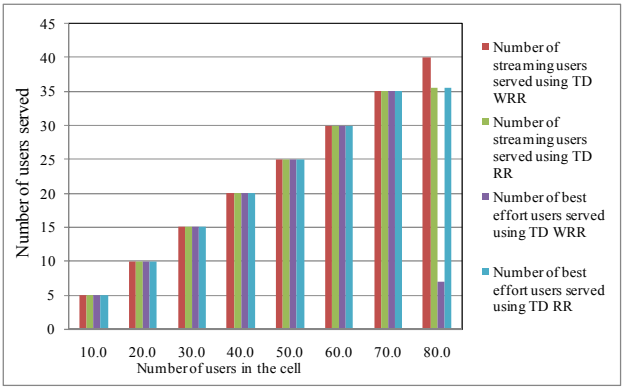


Figure 22. Number of users served vs. number of users in the cell in LTE uplink with SIMO 1x2 for TD RR and TD WRR

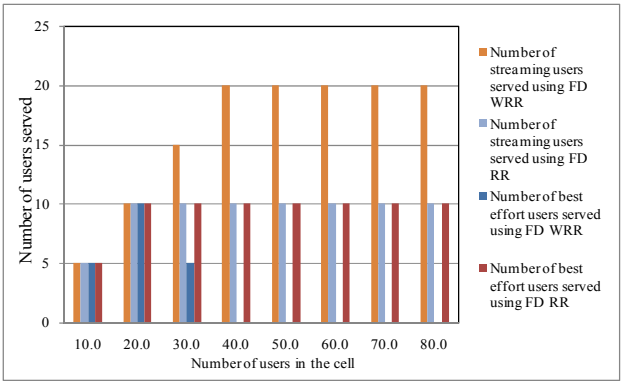


Figure 23. Number of users served vs. number of users in the cell in LTE uplink with SIMO 1x2 for FD RR and FD WRR

VI. CONCLUSIONS AND FUTURE WORK

This paper evaluates the performance of LTE downlink and uplink in what concerns cell throughput, average user throughput and cell capacity using two scheduling models in various scenarios of antenna configurations, radio conditions, number of users and service categories. The constraint of PDCCHs on the number of users scheduled each TTI, both for LTE downlink and uplink, has also been outlined and depicted in the simulation results, making FD RR with 1 RB assigned to each user less efficient when the number of users in the cell is higher than the PDCCHs. It was also discussed the limitation in the actual number of RBs in the transmission bandwidth brought by the uplink control channels. Taking into account that the mobile terminal is power limited and may not be able to support the assignment of the entire system bandwidth, the FD RR with more than 1 RB per user per TTI is more suited.

Despite its limitations, these scheduling models can offer an image of the LTE network performance and may be a useful tool to design an optimized LTE network, the most important aspect being the cell capacity evaluation with certain minimum or expected service throughput. Certain scenarios presented in this paper have been replicated in other simulation environments and the results obtained were similar to those presented in Section V.

Future work will focus on the analysis of LTE network performance using Opnet simulator as it offers some performance indicators evolution in time, latency results and more complex traffic mix scenarios.

ACKNOWLEDGMENT

Oana Iosif is POSDRU grant beneficiary offered through POSDRU/6/1.5/S/16 contract.

REFERENCES

[1] 3GPP TR 25.913v8.0.0 Release 8, “Requirements for evolved UTRA (E-UTRA) and evolved UTRAN (E-UTRAN)”.

[2] 3GPP TS 36.300v8.12.0 Release 8, “Evolved universal terrestrial radio access (E-UTRA) and evolved universal terrestrial radio access network (E-UTRAN); Overall description; Stage 2”.

[3] H.A.M. Ramli, K. Sandrasegaran, R. Basukala, and W. Leijia, “Modeling and simulation of packet scheduling in the downlink long term evolution system”, 15th Asia-Pacific Conference on Communications, APCC 2009, pp. 68-71

[4] H.A.M. Ramli, R. Basukala, K. Sandrasegaran, and R. Patachaianand, “Performance of well known packet scheduling algorithms in the downlink 3 GPP LTE systems”, IEEE 9th Malaysia International Conference on Communications, MICC 2009, pp. 815-820

[5] F. Capozzi, D. Laselva, F. Frederiksen, J. Wigard, I.Z. Kovacs, and P.E. Mogensen, “UTRAN LTE Downlink System Performance under Realistic Control Channel Constraints”, IEEE 70th Vehicular Technology Conference Fall (VTC 2009-Fall), pp. 1-5

[6] O. Iosif and I. Banica, “On the analysis of packet scheduling in downlink 3GPP LTE system”, The Fourth International Conference on Communication Theory, Reliability and Quality of Service, CTRQ 2011, April 17-22, Budapest, pp. 99-102, IARIA XPS Press, ISBN: 978-1-61208-126-7

[7] E. Yaacoub, H. Al-Asadi, and Z. Dawy, “Low complexity scheduling algorithms for LTE uplink”, IEEE Symposium on Computers and Communications, ISCC 2009, pp. 266-270

[8] S. Lee, I. Pefkianakis, A. Meyerson, S. Xu, and S. Lu, “Proportional Fair Frequency-Domain Packet Scheduling for 3GPP LTE Uplink”, INFOCOM 2009, IEEE, pp. 2611-2615

[9] H. Yang, F. Ren, C. Lin, and J. Zhang, “Frequency-Domain Packet Scheduling for 3GPP LTE Uplink”, INFOCOM 2010, Proceedings IEEE, pp. 1-9

[10] 3GPP TS 36.211v8.9.0 Release 8, “Evolved universal terrestrial radio access (E-UTRA); Physical channels and modulation”.

[11] H. Holma and A.Toskala, “LTE for UMTS: OFDMA and SC-FDMA based radio access”, 2009 John Wiley & Sons

[12] F. Khan, “LTE for 4G mobile broadband”, Cambridge University Press 2009.

[13] D. Astély, E. Dahlman, A. Furuskär, Y. Jading, M. Lindström, and S. Parkvall, “LTE: The evolution of mobile broadband”, Communications Magazine, IEEE In Communications Magazine, IEEE, Vol. 47, No. 4. (05 May 2009), pp. 44-51.

[14] R.Love, R. Kuchibhotla, A. Ghosh, R.Ratasuk, B. Classon, and Y. Blankenship, “Downlink control channel design for 3GPP LTE”, Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE, pp. 813-818.

[15] D. Laselva, F. Capozzi, F. Frederiksen, K. I. Pedersen, J. Wigard, and I.Z. Kovács, “On the impact of realistic control channel constraints on QoS provisioning in UTRAN LTE”,

Vehicular Technology Conference Fall, 2009 IEEE 70th, pp. 1-5.

[16] S. Sesia, I. Toufik, and M. Baker, "LTE – The UMTS Long Term Evolution: From Theory to Practice", 2009 John Wiley & Sons

[17] R. Kreher and K. Gaenger, "LTE Signaling, Troubleshooting and Optimization", 2011 John Wiley & Sons

[18] S. Hussain, "Dynamic radio resource management in 3GPP LTE", Blekinge Institute of Technology 2009.

[19] C. Han, K. C. Beh, M. Nicolaou, S. Armour, and A. Doufexi, "Power efficient dynamic resource scheduling algorithms for LTE", Vehicular Technology Conference Fall, 2010 IEEE 72nd, pp. 1-5.

Fairness for Growth in the Internet Value Chain

Alessandro Bogliolo[1,2] and Erika Pigliapoco[1]
[1]*STI-DiSBeF - University of Urbino, Urbino, Italy 61029*
[2]*NeuNet Cultural Association, Urbino, Italy 61029*
Email: {alessandro.bogliolo, erika.pigliapoco}@uniurb.it

***Abstract*—Empyrical data show an exponential growth of IP traffic and a corresponding growth of the overall capitalization of the Internet market. However, the revenues generated by the Internet are not fairly distributed among all the players involved in the value chain. In spite of the increasing returns for over-the-top service providers, application developers, device producers, network operators, and content right owners are not taking advantage of Internet evolution. Analysts forecast that in a few years this imbalance will cause the congestion of the network without any motivation for new investments on it, thus ultimately bringing the Internet to collapse. On the other hand, if properly distributed, the value generated by Internet traffic would be sufficient to sustain innovation and growth. This paper demonstrates with mathematical arguments that a fair distribution of the operating incomes across the value chain would maximize the development rate. Furthermore, it analyses the bottlenecks in the value chain induced by the access-based business models currently adopted by operators and often enforced by regulatory authorities. Net neutrality and market law are the pillars of an alternative service-based model which could be adopted to grant to the network the degrees of freedom necessary to overcome its own bottlenecks while reducing the need for policy enforcement.**

***Keywords*-Internet value chain; Growth; Fairness; Sustainability; Neutral Access Networks**

I. Introduction

The exponential growth of IP traffic is not occasional. Rather, it is the result of many concomitant causes: the ever increasing pervasiveness of the Internet, users' addiction to network connectivity, the progressive shift of usage patterns towards bandwidth intensive services, the significant improvements in the usability of interfaces, the ubiquitous availability of connected devices, the increasing share of consumer traffic, and the convergence of popular services (voice, TV, video on demand) over IP networks [5]. Global mobile data traffic is expected to increase 26 times in 5 years, reaching 6.3 exabytes per month in 2015 [6], while in 2014 the annual growth of fixed Internet traffic is expected to become greater than the overall volume in 2009 [7].

The beneficial effect of Moore's law, which keeps improving the performance and the cost effectiveness of network equipment, is not sufficient to sustain this exponential trend, so that continuous investments are required to boost network capacity. The question is: Does the network generate enough value to sustain its own development? According to aggregate financial data the answer seems to be positive, since the overall capitalization of the Internet follows the same exponential trend of IP traffic. A closer look at the Internet supply chain, however, points out a significant imbalance between segments which benefit from traffic growth (including user-interface producers and over-the-top service providers) and segments which suffer from the lack of incremental revenues (including content right owners and connectivity providers) [8]. Such an imbalance risks to impair network development.

Analysts observe that the *capital expenditures* (CapEx) required to fund incremental capacity both in fixed and in mobile networks are much higher than those obtained from the projections based on historical data. CapEx is the amount of money spent by a company to acquire or upgrade its assets in order to increase its capacity or efficiency for more than one accounting period. For a network operator the assets include network infrastructure, equipment, software, sites, and civil assets [9]. The ongoing costs incurred for running the business are called *operating expenditures* (OpEx). Although the revenues of network operators are still sufficient to pay for OpEx, in order for network development to keep pace with the estimated traffic growth, in the next 5 years mobile and fixed infrastructures will ask for a CapEx which is 50% and 30% higher, respectively, than currently planned for the same years [7]. Such additional investments cannot be made as long as operators do not take advantage from evolution. Hence, the imbalance between costs and revenues and the unfair capitalization of Internet value induced by current business models will end up impairing evolution and bringing the network to a congestion which will affect the whole value chain.

Although governmental measures (such as public funding, antitrust rules, and neutrality enforcement) have been often adopted to mitigate this phenomenon [4], they cannot be considered as ultimate solutions to guarantee a sustainable growth and the Internet prompts for new models [7], [2], [10], [11].

This paper starts from the observed and forecast trends of IP traffic and Internet capitalization to investigate how the value should be ideally distributed along the Internet supply chain in order to sustain the maximum rate of development. Extending the analysis recently conducted by the same authors [1] this paper proposes the adoption of a service-based network model (as opposed to the traditional access-based one) which could grant to the Internet the capability of

overcoming its own bottlenecks without giving up network neutrality and without requiring external enforcement.

The rest of the paper is organized as follows. Section II demonstrates, with simple mathematical arguments, that a fair distribution of the revenues along the entire value chain is the key to the development of the Internet. Moreover, it shows that, in the medium period, all the players involved can gain a higher benefit from a fair participation in Internet growth rather than from grabbing a higher share in the short term, so that collective welfare matches individual interests. Section III investigates whether and to what extent the results of Section II can be impaired by bit devaluation caused by the increase in network capacity. Section IV provides a detailed description of the Internet value chain, points out the bottlenecks induced by traditional business models, and introduces a service-based value chain as opposed to the current one, which is access-based. Section V proposes a service-based network model and shows how it could be exploited to achieve the conditions to maximize the development by following market law while also preserving network neutrality. Section VI analyses market signs which prompt for the adoption of a service-based model, while Section VII draws conclusions.

II. Fairness for growth

The *value* of a good or service can be defined as its worth determined by the market. The *value chain* (VC) describes the full range of activities which are required to bring the good/service from conception to delivery [12]. To our purposes, we call *value per bit*, denoted by V, the overall worth generated by the processing of 1 bit on the network across the entire VC. We call *operating profit per bit*, denoted by *OpProfit*, the difference between the value per bit and the operational costs *OpEx* incurred at all the N steps in the value chain to manage that bit. In symbols:

$$OpProfit = V - \sum_{k=1}^{N} OpEx_k \tag{1}$$

The opearting profit at stage n is defined accordingly as:

$$OpProfit_n = V_n - OpEx_n \tag{2}$$

where V_n is the revenue per bit at stage n.

Assuming that there is a positive overall operating profit, the value has to be distributed over the VC in such a way that the following condition is met at each stage

$$V_n \geq OpEx_n \qquad \forall n \in [1, N] \tag{3}$$

or otherwise the entire chain would not be sustainable. Then, operating profit can be used to sustain development according to the business models adopted by the players involved. For our purposes, we represent the business model adopted at the n-th stage by the *reinvested earning per bit*, denoted by RE_n, that is the percentage of the operating profit generated by a bit at stage n which will be re-invested. In symbols:

$$RE_n = \frac{CapEx_n}{OpProfit_n} \tag{4}$$

The development rate that can be achieved at a given stage (say, n) of the VC can be computed as the ratio between the actual $CapEx_n$ and the *marginal CapEx* required to increase of 1 bit the throughput of that stage ($MCapEx_n$). Since the overall capacity of the VC (in terms of number of bits it can process in a time unit) is equal to the minimum of the capacities of its N stages, the maximum development is achieved when all the stages evolve at the same rate. It can be easily demonstrated that this condition is met when the operating profit is distributed in such a way that each stage receives a share proportional to the investment per bit required at that stage ($MCapEx_n$) divided by the reinvestment model adopted (RE_n). In symbols:

$$V_n = OpEx_n + OpProfit \frac{\frac{MCapEx_n}{RE_n}}{\sum_{k=1}^{N} \frac{MCapEx_k}{RE_k}} \tag{5}$$

In this case, in fact, the development rate will be the same at all stages, avoiding bottlenecks which will cause diseconomies and impair evolution. The common development rate is given by:

$$DevRate = \frac{OpProfit}{\sum_{k=1}^{N} \frac{MCapEx_k}{RE_k}} \tag{6}$$

As long as the rate is maintained, the development follows an exponential trend which induces an exponential growth in time (t) of the overall capacity of the network (C) and of the profit generated at each stage, that can be expressed, respectively, as:

$$C(t) = DevRate^t \tag{7}$$

$$Profit_n(t) = (V_n - OpEx_n - CapEx_n) DevRate^t \tag{8}$$

Both the capacity and the profit at time t are expressed referring to a single bit processed at time 0. This means that, in order to obtain the actual capacity and the actual profit at the time t, Equations 7 and 8 should be multiplied by the values taken by the corresponding figures at time 0. For our purposes, in the following we keep using normalized quantities.

Now assume that one of the players in the VC has the power to capture more value (V_i) than expected according to Equation 5. The consequence will be a higher profit per bit at that stage, but a lower development rate for the entire VC. Referring to equation 8, this means that the player can decide to increase the multiplicative constant of his profit curve, at

the cost of decreasing the base of the exponential. Needless to say, this behavior will become counterproductive in a very short time, since the new short-sighted trend cannot compete with the optimal one, which has a stronger exponential.

This simple reasoning demonstrates that the splitting provided by Equation 5 is an equilibrium point that could be autonomously reached in a competitive market where all the stages in the VC are managed by rational agents. In other terms, it represents a win-win solution where the maximization of collective welfare is achieved by the decisions taken by individual agents in the attempt of maximizing their own profit.

III. Dealing with bit devaluation

The previous section has shown that the growth of the Internet market, if properly managed, is not an issue per se, in that it provides the economic motivation to induce all the players to fairly participate in the development required to satisfy the growing demand.

However, the mathematical model has been derived from three main parameters (namely, V, $MCapEx$, and $OpEx$) which have been treated as constants over time. Since, by definition, they are referred to each single bit, they are likely to depend on the amount of bits that can be processed by the network. In particular, both the operating costs per bit ($OpEx$) and the capital expenditures required for each additional bit ($MCapEx$) are expected to benefit from Moore's law and scale economies, which act as negative exponentials.

$$OpEx(t) = OpEx^{(t=0)}\alpha^{-t} \tag{9}$$

$$MCapEx(t) = MCapEx^{(t=0)}\beta^{-t} \tag{10}$$

A similar effect can be observed on the worth of each bit, because of the increasing amount of bits traveling across the network:

$$V(t) = V^{(t=0)}\gamma^{-t} \tag{11}$$

This phenomenon, hereafter called *bit devaluation*, is due to the reduction of the value of a unit of product (i.e., the bit) caused by the increase in the amount of supplied product units (i.e., the overall traffic). This section investigates whether, and to what extent, bit devaluation might impact the results of Section II.

For the sake of simplicity, and without loss of generality, let's assume that in Equations 9 and 10 $\alpha = \beta$. As for γ (which apperas in Equation 11) there are two main reasons, confirmed by empyrical observations, to assess that it has to be lower than α and β: first, because the devaluation of the bits is one of the effects of network development, and it is unlikely that the effect goes faster than its cause; second, because the whole capitalization of the Internet market is growing, while it would decrease if the worth of each bit (V) reduced faster than the costs incurred to generate it.

On the basis of the above arguments, the case in which $\gamma = \alpha = \beta$ can be regarded as the worst-case scenario to be used to evaluate the effects on the development rate expressed by Equation 6. Since the negative exponentials appear both at numerator and denominator of a fraction, their effects cancel out, so that $DevRate$ remains constant over time. On the other hand, if γ was lower than α and β, then $DevRate$ would grow over time.

To better highlight the possible effects of devaluation on profits, Equation 8 is rewritten as the product of three terms:

$$\begin{aligned} Profit_n(t) &= (OpProfit_n - CapEx_n)DevRate^t \\ &= OpProfit_n(1 - \frac{CapEx_n}{OpProfit_n})DevRate^t \\ &= OpProfit_n(1 - RE_n)DevRate^t \end{aligned} \tag{12}$$

where $DevRate$ has already been studied, while the second term does not contain time-dependent parameters. Hence, the only term which needs to be discussed is the first one, which represents the operational profit per bit, the time dependence of which can be expressed as

$$\begin{aligned} OpProfit_n(t) &= V_n^{(t=0)}\gamma^{-t} - OpEx_n^{(t=0)}\alpha^{-t} \\ &= \gamma^{-t}(V_n^{(t=0)} - OpEx_n^{(t=0)}\left(\frac{\alpha}{\gamma}\right)^{-t}) \end{aligned}$$

Replacing the expression of $OpProfit_n(t)$ in Equation 12, in the worst case of $\gamma = \alpha$ the profit at stage n can be expressed by the following function of time:

$$Profit_n(t) = \gamma^{-t}(V_n^{(t=0)} - OpEx_n^{(t=0)})(1 - RE_n)DevRate^t \tag{13}$$

Recognizing that two of the four terms of the product do not depend on time, we can define a constant

$$K_n = (V_n^{(t=0)} - OpEx_n^{(t=0)})(1 - RE_n)$$

and rewrite Equation 13 pointing out its dependence over time.

$$Profit_i(t) = K_i\left(\frac{DevRate}{\gamma}\right)^t \tag{14}$$

Equation 14 clearly shows that the profit keeps growing exponentially as long as γ is lower than the development rate, which is a reasonable assumption (compliant with empirical data) since γ represents the devaluation driven by development. This trend is confirmed by empyrical observations.

IV. The Internet value chain

Among the different ways to represent Internet value chain (VC), one of the most detailed and recent representations is provided by A.T. Kearney [8], which splits the Internet market into 5 segments, namely: content rights, online services, enabling technology services, connectivity, and user interface. In order to point out the differences between access-based and service-based business models, we adopt a 7-stage VC obtained by separating the Internet core from the access network (both of them included into the Connectivity segment in A.T. Kearney's report) and by distinguishing the services provided *over the top* (OTT) from those provided within operators' managed networks (the latter not explicitly mentioned in the above report).

Figure 1. The Internet value chain.

The resulting VC, shown in Figure 1, is composed of the following stages: contents and applications (stage 1), that could be either copy righted or generated by end-users; OTT online services (stage 2), made globally available on the Internet; support technologies (stage 3), which include content delivery overlay networks and hosting services; Internet core (stage 4), made of interchange points and core networks of incumbent operators; online services provided within managed networks (stage 5), which include IPTV services; access networks (stage 6), which include both backhauling and retail access up to the network termination points made available to end-users; user devices (stage 7), which include HW/SW user interfaces and customer premises equipment (CPE) used to connect to network termination points.

It is worth noticing that stage 4 includes both operators' backbones and interchange points, so that Figure 1 does not point out the re-distribution of value within the Internet core, which is governed by peering agreements and managed by international organizations.

According to historical data of market capitalization [7], VC segments have followed very different trends in the recent past: while stages 2, 3, and 7 have known a significant growth from 2004 to 2010 (4x, 2x, and 5x respectively), stages 1, 4, and 6 have not taken any advantage of the fast increase in Internet traffic and their capitalization has slightly decreased in the same period. As for segment 5, mainly represented by IPTV market, the compound annual growth rate is expected to be around 25% until 2014 [13].

The imbalance of Internet market capitalization is schematically represented in Figure 2 in order to provide a qualitative perception of the bottlenecks which risk to impair network development.

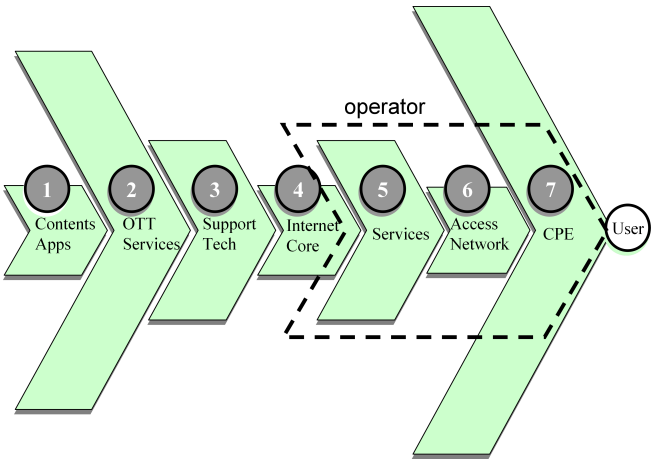


Figure 2. Schematic representation of the unbalanced capitalization of the Internet value chain.

A. Access-Based Value Chain

The current functioning of the network is dominated by two main features. The first one is *vertical integration*, which is the absorption into a single organization (namely, the so-called *operator*) of all the aspects required to go from the Internet core to end-users, often including even the provision of customer equipment (vertical integration is represented as a dashed macro-stage in Figure 2). The second one is the *all-or-nothing offer* of Internet access, which gives to end-users only the categorical choice between subscribing to full access to the network, or being completely cut off. Internet access is typically sold at a monthly flat fee depending only on the nominal (i.e., maximum) bandwidth at user's disposal.

From operators' stand point this business model was originally motivated by the perspective of: attracting costumers with a simple offer, avoiding the operating costs of complex accounting policies, taking advantage from average individual use well below the nominal bandwidth, and exploiting statistical sharing to over-book the bandwidth available.

From end-users' stand point, the model has induced the misleading perception that: Internet bandwidth is the only good customers pay for (while they also pay for access infrastructures and CPE), the nominal bandwidth is the actual one they are entitled to use all the time (while it represents only a peak value they are not allowed to pass), and the more they use the network the more convenient their contracts become (while the monthly rate was determined assuming they would not have used the Internet all the time).

From OTT service providers' stand point, the access-based business model has created a global market where to offer their services without caring about transport, allowing them to deliver most services for free and to get money from commercial sponsors.

It is a matter of fact that the Internet has become a two-sided market where the apparent gratuitousness of traffic

has created a short-circuit between the two sides (namely, service providers and end-users), cutting off from revenues network operators, which lay in the middle.

The ultimate effect of this phenomenon is the so-called *cloud computing*: users feel Internet services to be so close to them and reachable at no additional costs, that they keep on the cloud even their personal files that could fit at no cost in the storage devices embedded in their smart phones.

Although such a short circuit has significantly contributed to the diffusion of the Internet and to the development of advanced online services, the model suffers from many weaknesses which make it unsuitable to sustain the exponential development.

First, the advent of a huge variety of services with different bandwidth requirements has created a significant spread of usage patterns with a consequent inequality among users who pay the same fee in spite of heterogeneous needs (for instance, 1% of mobile data subscribers generate over 20% of mobile data traffic [6]). If such a monthly fee is higher than the perceived value of the Internet, individuals may be not motivated enough to subscribe.

Second, there are some stages in the VC of Figure 2 (such as stage 6) which significantly contribute to the costs incurred by operators without generating any direct value, since they are hidden to end-users. This misalignment between costs and revenues impairs innovation because operators are neither motivated to invest in access infrastructures nor interested in boosting the development of bandwidth-intensive services.

Third, as the average individual use gets close to the nominal bandwidth included in the monthly fee, over-booking causes the congestion of access networks with consequent loss of quality of service (QoS).

To contrast these effects, operators have tried to reach scope economies by adopting the so-called *triple-play* market strategy, which consists in providing additional services (namely, IPTV and VoIP) within the walled gardens of their own networks. Moreover, they have been induced to apply *traffic shaping* and *access tiering* techniques in order to delay the congestion of their networks and to mitigate its effects on QoS.

Governments, on the other hand, have come on stage in many ways in order to bridge digital divide, foster competition, and defend end-users' interests. In particular, public funds have been allocated in many countries to finance the development of *next generation networks* (NGNs) and the deployment of access infrastructures in market failure regions, regulations have been enacted to impose incumbent operators to make their infrastructures available to new entrants at controlled wholesale/unbundling conditions, and network neutrality has been enforced by preventing operators from adopting access tiering policies and from establishing commercial relationships with OTT service providers.

If state interventions can play a significant role in triggering development, they cannot guarantee sustainability (if not complemented by private investments and not supported by suitable business models) and they often produce side effects that may even end up thwarting their own original purposes. This is the case of neutrality enforcement and local loop unbundling, which discourage private investments in NGNs by reducing business opportunities, by avoiding bandwidth optimizations, and by making the break-even point unreachable in many scenarios. Moreover, state financial aids, even if targeted only to access networks (stage 6 in the VC), create significant distortions in many other markets (stages 4, 5, and 7 in the VC) because of vertical integration and triple-play market strategies currently adopted by incumbent operators.

B. Service-Based Value Chain

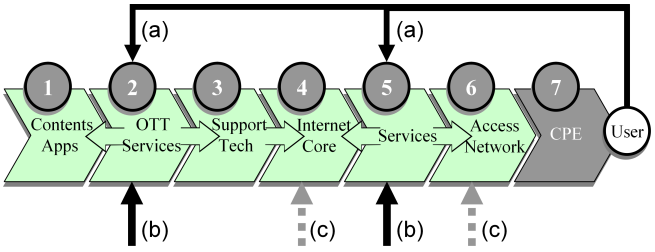


Figure 3. Service-based Internet value chain.

The VC proposed in this section is based on two main features: *vertical separation*, as opposed to vertical integration, and *service orientation*, as opposed to access orientation typical of the current Internet model discussed in the previous subsection.

Technically speaking, separation is an inherent property of the Internet induced by the layered structure of its protocol stack. Network neutrality, which has been one of the main driving forces behind Internet development and innovation, was naturally induced by the layered architecture before becoming a controversial principle. In this context, vertical separation is particularly intended as market segmentation, which enables each segment in the VC to be possibly managed by different actors who interact with all other segments by means of transparent and profitable commercial relationships. Separation also enables each market segment to make business with other industry sectors and public organizations, not represented in the VC, or to be targeted by state financial aids and welfare policies. Vertical separation has been identified by ITU as one of the four technology implications on market structure which prompt for new business models, the other three implications being service innovation, network innovation, and horizontal integration (i.e., network convergence) [14].

Service orientation, which is the second distinguishing feature of the proposed VC, means the opportunity for end-users to directly focus on the services they need, even if they

have not yet established commercial relationships with any operator. There are many motivations for focusing on services (delivered both OTT and within managed networks): services/applications are at the top of the TCP/IP stack, they are much more attractive than their enabling technologies (i.e., connection and transport), they provide great opportunities of diversification and innovation, and they have proved capable of taking advantage of traffic growth. Although market capitalization data clearly demonstrate that services are the main driving force of the Internet, current business models do not provide adequate instruments to distribute the revenues along the VC in order to support the development required at all its stages.

An ideal representation of a service-based VC is provided in Figure 3. End-users establish direct relationships with service providers (SPs), who operate both at stage 2 (OTT) and at stage 5 (within managed networks). These interactions, which may or may not involve payments, are represented by black arrows with label (a) in Figure 3, where thick arrows with label (b) represent revenues coming from sponsorships, advertisements, and any other form of business made with stakeholders who take advantage of the Internet without being directly involved in the VC. Both type-a and type-b incomes are collected at stages 2 and 5, even if all stages contribute to the VC. Transparent relations among the actors operating at different stages are then needed to enable a fair redistribution of revenues along the service-based VC. Inter-stage redistributions are represented by horizontal arrows in Figure 3. Finally, dashed arrows with label (c) represent financial aids possibly targeting backbones (stage 4) and access infrastructures (stage 6).

Stage 7 (i.e., CPE) is shadowed in Figure 3 and it is not involved in any inter-stage commercial transaction because it is a thriving market by itself, which is expected to be able to keep following and supporting Internet growth without the need for significant changes in its business model. In other terms, end-users' devices (such as smart phones, net books, PCs, set-top-boxes, ...) can be considered to be already at users' disposal, since customers are highly motivated to pay for them. Hence, they can be neglected in our analysis since they are neither a bottleneck to be overcome, nor a source of revenues suitable to be redistributed along the VC. Notice however that the lack of interactions between stage 7 and the rest of the VC does not mean that CPE cannot be provided by operators (as they are usually in current business models). Rather, it simply means that this kind of scope economies are not considered to be relevant for network development.

Internet bandwidth is nothing but a special kind of service provided at stage 5 by Internet service providers (ISPs) who manage gateways placed between access networks (stage 6) and Internet core (stage 4). Access infrastructures are assumed to be open to end-users, whose CPE associates for free in order to allow them to gain access to online services (including Internet bandwidth). SPs and ISPs pay a fee to the operators managing the access network in order to be allowed to expose their services to connected end-users. As long as SPs share their revenues with access network operators, the latter are motivated to open their networks to end-users, in that they add to the value of the network by making it more attractive for SPs. This allows operators to take advantage of the development of the two-sided market they enable, and provides the motivation required to invest in access infrastructures.

OTT SPs may keep exposing their services on the Internet without establishing direct relationships with network operators. In this case, they can be reached by end-users who subscribed with some ISP to gain access to the Internet, while they will not be reached by end-users who have connected only to the access infrastructure without buying Internet bandwidth. On the other hand, OTT SPs can decide to enter into a contract with an operator to make their services also reachable, within managed access networks, to end-users who associated for free with the access infrastructure. In the first case the traffic generated within the acces network is paid by end-users (as a share of the fee they pay to ISPs), while in the second case it is paid by SPs. Finally, depending on the nature of the services, OTT SPs may or may not share their revenues with content providers (stage 1) and enabling technology providers (stage 3).

Although many different business models can be conceived and adopted, commercial relationships should be mainly based on IP traffic in order to provide the so-called *price-signal* which acts as a positive feedback in triggering and sustaining development.

In summary, the service-based VC provides a suitable support for development and growth, in that it lowers access barriers for end-users, it reduces information asymmetry by avoiding end-users to be billed unawarely for the traffic generated by the services they use, and it allows operators to establish transparent commercial relationships with SPs without violating network neutrality. In fact, neutrality is preserved as long as the same conditions are applied at each stage to all the actors playing the same role in the VC.

V. A SERVICE-BASED MODEL

Moving from an access-based to a service-based model implies a paradigm shift in the Internet VC. While at some stages such change can emerge from the natural evolution of current business models, at some others it prompts for innovative architectural and commercial models. The most challenging issue in this context is the re-design of the relationships among end-users, operators, and SPs across access infrastructures. To this purpose, a suitable support can be provided by the so-called *neutral access network* (NAN) model [15].

NANs are a special category of open access networks [16] conceived to make the access infrastructures economically sustainable in market-failure regions by triggering

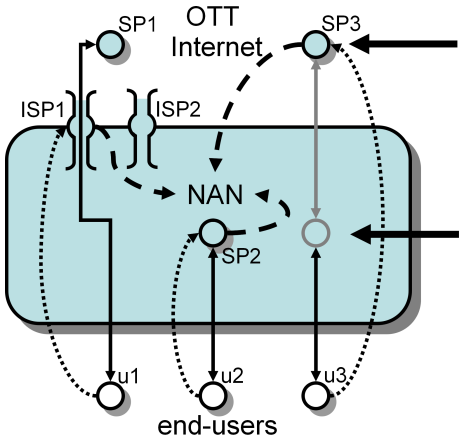


Figure 4. Interactions between end-users and SPs in a service-based neutral access network.

positive externalities and enhancing penetration [17]. A NAN exhibits the features of a full-fledged network by itself, containing a sizeable set of services made available to the users before they register with any ISP. End-users are allowed to associate with the NAN for free without prehemptive registration. Once the users have entered the NAN, they are exposed to all the services made available within the network, including Internet surfing through the gateways managed by ISPs. Registration and authentication are required only to gain access to the Internet or to those internal services which require user identification for accounting, personalization, privacy, or security needs. The entry of a new user into the NAN has a beneficial effect for all other users since it helps reaching the critical mass of users required to incentivize the provisioning of new services. Similarly, the entry of a new SP has a spillover benefit for all other providers since it induces new users to enter the shared marketplace and it contributes to cover the costs of the infrastructure. Service orientation is natural in a NAN. End users have commercial relationships only with SPs (including ISPs), who pay a share of their revenues to the NAN organization. The share is then possibly distributed among multiple stakeholders: real estate owners, investors, and local operators.

Figure 4 represents the possible relations that can be established in a NAN. Vertical solid arrows stay for IP traffic, dotted arrows stay for direct transactions between end-users and SPs, horizontal solid arrows stay for revenues coming from markets outside the VC (including sponsorhips and advertisment), while dashed arrows stay for commercial relationships between SPs and NAN operators. Three paradigmatic cases are depicted, referring to three end-users who are assumed to be connected for free to the NAN by means of their own CPE.

Case 1. End-user u1 wants to gain full access to the Internet. To this purpose, he/she registers with one of the virtual operators (namely, ISP1) offering Internet bandwidth in the NAN. The conditions at which Internet bandwidth is sold by ISP1 include the share he has to pay to the NAN operator for transporting u1's traffic across the NAN. Once on the Internet, u1 takes advantage of the service delivered by an OTT SP (namely, SP1) without taking care of transport. This case reproduces the same user experience of current access-based models, while retaining the benefits of service orientation. Commercial agreements between ISP1 and NAN operators can assume the form of a wholesale contract, but the key novelty is that u1 connected to the NAN before registering with ISP1 and was allowed to choose the ISP as a service.

Case 2. End-user u2 associates to the NAN without buying Internet bandwidth since it is only interested in a specific service (like tourist information, e-government, IPTV, ...) which is supplied by SP2 within the access network. The only relation he/she has to establish is with SP2, who is supposed to pay a fee to the NAN operator for web hosting and transport. Revenues for SP2 can come either from end-users (if they pay for the service), or from sponsors/subsidies (if the service is delivered for free), or from both (if a mixed model is adopted, such as the one of IPTVs providing both free channels and pay-per-view contents).

Case 3. End-user u3 behaves exactly as u2, even if the service he/she wants to use is provided by an OTT SP (namely, SP3). This is made possible by the agreement between SP3 and the NAN operator, signed to expose the online service of SP3 within the NAN. From a technical point of view, this could be done in many different ways, including mirroring, proximity caching, and white listing. The traffic generated across the NAN is then paid by SP3, while the service he/she provides makes the access infrastructure more attractive.

The trade-off between network neutrality, bandwidth optimization, and capitalization is reached thanks to the nature of the commercial relations established at all stages, which are not discriminatory, not exclusive, and inherently regulated by market law.

VI. Market signs

The urgent need for a paradigm shift in the Internet VC can be viewed in many recent events and market signs.

Amazon's *Kindle 2* has conquered the market of e-book readers by freeing end-users from the burden of connectivity. It integrates a hidden SIM card which allows end-users to be always connected (seemlessly) to the online store. The cost of download is included into the price of e-books thanks to an agreement between Amazon and AT&T, which in its turn has roaming agreements with mobile operators all around the world [18]. This is a neat example of a vertical application

built on top of a vertically-separated architecture to provide a service-oriented user experience.

Groupon (www.groupon.com) is a deal-of-the-day website which operates in hundreds of localized markets worldwide. The business model is fairly simple: it offers a deal per market per day. If users who sign up for the offer reach a given threshold, then the deal becomes available to all of them and the retailer shares his/her revenues with Groupon. For retailers, Groupon works as an *assurance contract* which guarantees a critical mass which makes the deal like a *quantity discount* [19]. In 2010, Groupon Inc. refused a 6 billion Dollar offer from Google, clearly demonstrating the value of localized on-line business. It is apparent that Groupon could provide its services within a NAN, making it available to local end-users even if they have not signed with any ISP.

In January 2011, Google Inc. accepted to allow publishers to quit *Google News* without affecting the results returned by its main search engine, and to disclose revenue-sharing arrangements for its *AdSense* partners. This agreement ended an antitrust investigation of the *Italian Competition Authority* (AGCM) triggered by the *Italian Federation of Newspaper Publishers* (FIEG) because most people were content with aggregated summaries found on Google News and bothered to click on the links that led to their newspaper websites, costing the publishers advertising and page views. This story shows that services (e.g., online aggregators and search engines) are much closer to end-users than contents (e.g., news), so that it is much easier for SPs than for content right owners to be paid by end-users and sponsors. The agreement found in Italy also demonstrates that it is worth for both categories to find a suitable revenue sharing mechanism which reduces the imbalance and makes the business sustainable.

Google Inc. has provided free Wi-Fi access in Mountain View (CA) for several years and it has contributed to the development of many other municipal networks. In February 2011 the City Council approved a 5-year extension of the *Google WiFi* deal, with an escape clause for Google. There are two signs that can be found in this piece of news: the first one is that OTT SPs are interested in widening their market by lowering access barriers, the second one is that they do not want to take the place of network operators (the escape clause was wanted by Google).

In December 2010 some of the major European mobile operators, including Orange, Telecom Italia, Telefonica and the Vodafone Group, have demanded that popular OTT services, such as those from Google, Facebook, Skype and Apple, contribute to pay for the traffic they generate on their networks. This request, motivated by the lack of return for operators from the exponential growth of IP traffic, has raised network neutrality issues due to the unsuitability of the business models adopted, which do not allow operators to establish commercial relationships with SPs without impairing the neutrality of the connection they provide.

In 2010 the European Commission launched a public consultation on "The open Internet and net neutrality in Europe" and received answers from 318 stakeholders [20], demonstrating the need for a thorough conciliation of the different interests involved in order to guarantee the development and the openness of the Internet. Similar consultations were launched in 2011 in many European countries. The pragmatic positions expressed by the European Parliament and by the national authorities based on the results of the public consultations clearly show the intent of policy makers to create the conditions for a fair competition in the Internet market with minimum interference from regulators [3], [4], [21].

In the second quarter of 2011 KPN, the incumbent operator in the Netherlands, decided to apply a surcharge on competing Internet-based services (including voice and instant messaging) to compensate the reduced earnings registered in the first quarter. In June 2011 the Dutch parliament approved the world's strongest net neutrality bill, banning operators from hindering or delaying OTT services and from applying surcharges. At the end of July 2011, KPN announced much higher data tariffs in response to the net neutrality act [22], demonstrating that regulatory intervention, if dispropotionate or precipitate, can turn out to be counter-productive. In October 2011, the digital agenda European commissioner Kroes attacked the unilateral decision of the Dutch Parliament, recommending more careful and coordinated interventions [23].

VII. Conclusions

In spite of the exponential growth of Internet traffic, the unequal distribution of revenues along the Internet VC, together with the imbalance between costs and revenues caused by the business models currently adopted by network operators, risk to impair evolution towards broadband next generation networks.

The Internet supply chain is like a pipeline the capacity of which is limited by the thinnest pipe, so that a fair distribution of revenues along the VC is essential to trigger and sustain network development. This has been shown in Sections II and III with simple mathematical arguments that demonstrate that fairness is the key for keeping pace with the exponential growth of Internet traffic.

The Internet VC has been analysed in Section IV in order to point out the limitations of current access-based models and to propose a paradigm shift towards a new service-based approach. Service orientation, complemented by suitable business models, allows all stages of the VC to take advantage of the attractiveness and diversity of online services and to benefit from the revenues they can generate in terms of sponsorships and advertisement.

Furthermore, it has been shown in Section V that neutral access networks provide a suitable support to the adoption of a service-based model, allowing end-users to connect for free to the access infrastructure and then focus only on the services they need, including Internet bandwidth. The systematic application of not exclusive agreements among the actors involved (service providers, content providers, and network operators) provides the basis for a fair redistribution of revenues along the VC, driven by market law rather than by policy enforcement.

Finally, market signs have been analysed in Section VI to give evidence of the urgency of the paradigm shift envisioned in this paper.

In conclusions, service orientation has been proposed in this paper as the key for granting to the Internet the degrees of freedom required to autonomously find the best balance among the segments in the VC, thus overcoming the bottlenecks and creating the preconditions for development.

ACKNOWLEDGEMENT

The research leading to these results has received funding from the EU IST Seventh Framework Programme ([FP7/2007-2013]) under grant agreement n. 25741, project ULOOP (User-centric Wireless Local Loop).

REFERENCES

[1] E. Pigliapoco and A. Bogliolo, "A Service-Based Model for the Internet Value Chain," in *Proceedings of the International Conference on Access Networks (ACCESS-11)*, 2011, pp. 13–18.

[2] H. W. Friederiszick, J. Kaluzny, S. Kohnz, M. Grajek, and L.-H. Roller, "Assessment of a Sustainable Internet Model for the Near Future," *ESMT White Paper*, 2011.

[3] M. Cave and P. Crocioni, "Net Neutrality in Europe," *communications & Convergence Review*, vol. 3, no. 1, pp. 57–70, 2011.

[4] J. S. Marcus, P. Nooren, J. Cave, and K. R. Carter, "Network Neutrality: Challenges and responses in the EU and in the U.S." *European Parliament - Policy Department A*, 2011.

[5] Akamai, "Q2 2011 - The State of the Internet," *Akamai report*, 2011.

[6] Cisco, "Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2010-2015," *Cisco White Paper*, 2011.

[7] A. T. Kearney, "A Viable Future Model for the Internet," *A.T. Kearney report*, 2010.

[8] ——, "Internet Value Chain Economics," *The Economics of the Internet, Vodafone Policy Paper Series*, 2010.

[9] S. Verbrugge *et al.*, "Methodology and input availability parameters for calculating OpEx and CapEx costs for realistic network scenarios," *OSA Journal of Optical Networking*, vol. 5, no. 6, pp. 509–520, 2006.

[10] E. Altman, P. Bernhard, S. Caron, G. Kesidis, J. Rojas-Mora, and S. Wong, "A model of network neutrality with usage-based prices," *Telecommunication Systems*, pp. 1–9, 2011.

[11] F. Maier-Rigaud, "Network Neutrality: A competition angle," *Competition Policy International*, vol.2, pp. 1–10, 2011.

[12] M. Porter, *Competitive Advantage: creating and sustaining superior Performance*. Free Press, 1985.

[13] Multimedia Research Group Inc., *IPTV Global Forecast 2010 to 2014 - Semiannual IPTV Global Forecast Report*. MRG, Inc., June 2010.

[14] International Telecommunication Union, *ICT regulation toolkit*. http://www.ictregulationtoolkit.org/, last visited in January 2012.

[15] A. Bogliolo, "Introducing Neutral Access Networks," in *Proceedings of the 5th IEEE Conference on Next Generation Internet Networks*, 2009, pp. 243–248.

[16] J. Barceló, A. Sfairopoulou, and B. Bellalta, "Wireless open metropolitan area networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 12, no. 3, pp. 34–44, 2008.

[17] E. Pigliapoco and A. Bogliolo, "Enhancing broadband penetration in a competitive market," in *Proc. of the International Conference on Evolving Internet*. IEEE Computer Society, 2010, pp. 159–163.

[18] C. Loebbecke, A. Soehnel, S. Weniger, and T. Weiss, "Innovating for the mobile end-user market: Amazon's kindle 2 strategy as emerging business model," in *Proc. of the International Conference on Mobile Business*. IEEE Computer Society, 2010, pp. 51–57.

[19] A. Mason, *System and Methods for Discount Retailing*. US Patent 2010/0287103 A1 (assigned to Groupon Inc.), 2010.

[20] EC Information Society and Media Directorate-General, "Report on the Public Consultation on the open Internet and net neutrality in Europe," *European Commission*, 2010.

[21] Ofcom, "Ofcom's approach to net neutrality," Statement published on November 24, 2011.

[22] Associated Press, "Dutch telecom hikes rates after net neutrality law," *AP report*, July 19, 2011.

[23] N. Kroes, "Investing in digital networks: a bridge to Europe's future," *ETNO Financial Times 2011 CEO SUMMIT - SPEECH/11/623*, October 3, 2011.

Movement synchronization for improving File-Sharing efficiency using bi-directional recursive data-replication in Vehicular P2P Systems

Constandinos X. Mavromoustakis
Department of Computer Science,
University of Nicosia
46 Makedonitissas Avenue, P.O.Box 24005
1700 Nicosia, Cyprus
mavromoustakis.c@unic.ac.cy

Muneer Masadeh Bani Yassein
Jordan University of Science and Technology,
P.O. Box 3030,
Irbid 22110, Jordan
masadeh@just.edu.jo

***Abstract*—Recent research in opportunistic networks address the diffusion policy and the practicality considerations in utilizing device-specific applications. Intermittent connectivity between mobile nodes and the random behavioral patterns of the users, make modeling and measuring the performance of opportunistic networks very challenging particularly in the case of file/object sharing among these devices. As the cache-and-forward replication policy plays a major role targeting the availability of requested resources, there should be a mechanism for controlling the minimization of the redundant replicas and the uncontrolled diffusion effects onto the communicating nodes. In this work the diffusion policy of requested replicated objects is examined with regards to the probabilistic synchronized movement of the devices, quantifying the parameters that affect the reliable transmission and the availability of requested resources by any node. The assigned scheme takes into consideration the movement synchronization of the moving devices and applies a recursive adaptive caching cooperation scheme in community-oriented relay regions. The scheme also uses the Message Ferry (MF) mobile Peer in a bi-directional mode, in order to enable higher degree of reliability in the availability of the requested resources. Conducted simulation experiments using real-time traffic traces show that the proposed scheme offers high throughput and reliability response for sharing resources on-the-move while it minimizes the redundancy of replications.**

Keywords- bi-directional recursive data-replication; partially synchronized mobility scheme; object sharing scheme; reliability and availability of resources; resource exchange efficiency; evaluation through simulation.

I. INTRODUCTION

On-the-move reliable resource exchange has been a fertile ground for enabling researches to explore further techniques for successful resource diffusion according to users' demands. In this direction a catalytic factor has been the Wireless technologies growth which represents another orthogonal area of growth, in both wide-area applications like 2.5G/3G and local area applications like 802.11b/g and Bluetooth. Many constraints exist in such networks like resource availability whereas the topological scheme followed in these infrastructures should be combined with the availability of the requested resources and the time-access for sharing resources with the synchronized motion within a specified time duration *t*. A Vehicular Ad-Hoc Network (VANET) is a technology that uses moving cars as devices/nodes in a dynamically changing network to establish a mobile network connectivity. In this paper a reliable file sharing scheme for vehicular Mobile Peer-to-Peer (MP2P) devices is proposed taking the advantages of moving devices within a specified roadmap with different pathways like in real time vehicular networks. This work exploits the movements of the devices and the passive device synchronization to increase end-to-end file sharing efficiency through vehicular users [1] and Mobile Infostations [1][3]. Through geographical roadmaps landscapes where mobile Infostations are set and initialized, the passive synchronization enables through the replication policy to create a replicated object in order to establish reliable file sharing. Role-based Mobile Infostations (MIs) are selected based on their velocity, residual energy, remaining capacity etc and are assigned according to the passive Message Ferry peer. This scheme proved its robustness in node's density since it does not require the knowledge and the global figure of the number of users. Additionally it does not require spatial distributions to efficiently spread information while enables reliability in supported mobility without the scheduled 'rendezvous', whereas it effectively passes the requested replicas to designated users.

The organization of the paper is as follows: Section II discusses the related work that has been done on similar schemes which use similar approaches for establishing and maintaining end-to-end file sharing efficiency. Section III then introduces the proposed model based on [1] where instead of using a uni-directional replication mechanism as in [1], this work utilizes a bi-directional recursive data-replication mechanism for opportunistically connected vehicular P2P Systems. The proposed mechanism optimizes significantly the work done in [1] by using the movement exploitation in a synchronized form, to increase end-to-end file sharing reliability and hosts a stochastic measure to estimate the end-to-end capacity within the path where the requested replicas were created. Section IV shows the

experimental simulation-based performance evaluation of the proposed scheme and the comparisons done under different convergent parameterized conditions. Particular focus was given to the impact of certain probabilistic movements made by Vehicular-Peer-to-Peer (VP2P) devices where multi-client applications, dynamically demand resources directly from certain nodal vehicles on-the-move. The stochastic model introduced in [1] is being used measuring the end-to-end capacity and the dynamic caching activity of the requested objects onto opportunistic neighboring devices. In addition the proposed model takes into consideration a number of parameters, in order to limit the potential inadequacies of resource sharing process due to intermittent connectivity, whereas through these parameters it enables further accuracy in the resource sharing process. A resource assignment cooperation engine is hosted under the proposed framework considering the proposed Go-back-N caching cooperation scheme taking place in a bi-directional mode using cluster-based approach with ranked requests.

II. RECENT SCHEMES AND WORK DONE

Mobile resource sharing policy needs to be supported by a reliable mechanism which will guarantee the resource exchange in an end-to-end available manner. A great amount of research effort has been invested to facilitate mobile applications in the last few years hosting approaches that can be categorized into three types: network layer–based approaches, transport layer–based approaches, and proxy-based approaches. A significant and common goal of these approaches is to maintain the connectivity for mobile users even when they perform vertical handoffs between different networks [2]. As mobility in opportunistic and autonomic communication is an essential parameter and along with the user's demands they pose the vision of what self-behaving flexibility should encompass in next-generation self-tuning behavior [1], the resource exchange apparatus should guarantee the consistency and reliability by using assisted mechanisms using resilience metrics [2] for enabling delay sensitive resource sharing. The capacity of the nodes which are traversed in the requested path, can be reduced significantly particularly if we are dealing with delay sensitive traffic or bursty traffic [1] whereas the underlying end-to-end supporting mechanism should be aware of the dynamic movements in a Peer-to-Peer manner. Obviously, if the transmission-range of a node increases, then the interference it causes will increase and probably the number of nodes which will have copy/copies of the packets that should be forwarded, will increase. Toumpis and Goldsmith [4] define and study capacity regions for wireless Ad-hoc networks with an arbitrary number of nodes and topology. These regions describe the set of achievable rate combinations between all source-destination pairs in the network under various transmission strategies for EC content sharing and power control. In this work we consider the capacity but in an end-to-end path-request manner, and take into consideration the variations caused by the dynamic movements of the devices/vehicles. Most existing architectures (including Grace [5], Widens [6], MobileMan [7]) rely on local information and local devices' views, without considering the global networking context or views which may be very useful for wireless networks in optimizing load balancing, routing, energy management, and even some self-behaving properties like self-organization.

Further to the research work done in [1] where, author associates the synchronized movements and the related connectivity aspects among vehicles, this work proposes and utilizes a scheme where the end-to-end file sharing efficiency, increases in vehicular MP2P devices. The scheme in [1] extends the advantages offered by the Hybrid Mobile Infostation System (HyMIS) architecture proposed by Mavromoustakis and Karatza, in [8], where the Primary Infostation (PI) is not static but can move according to the pathway(s) of the roadmaps. HyMIS adopts the basic concept of pure Infostation system in terms of capacity service node but it avoids flooding the network with unnecessary flow of information (redundant diffusion of unnecessary resources). This node plays a role of control storage node (backup capacity node) as Haas and Small mention in [9]. Taking the advantages of the proxy caching [10] and the cache-and-forward apparatus work done [2] this work proposes an exploitation of the mobility characteristics of each user by utilizing the MI peer to be dynamically selected according to characteristics such as the residual capacity of the device based on the push-based activities by other nodes. Additionally with the work done in [1] the innovating research aspect is that the proposed resource assignment cooperation scheme enables the caching mechanism to affect the degree of reliability using the variable window size into the requested replicas by N-hop peers. The proposed Go-back-N caching cooperation scheme takes place in a bi-directional mode in order to enable availability of requested resources (ranked requests) by certain peers in the community cluster. Heavy emphasis of this work has been put on push-based dissemination explored in [9] by Little and Agarwal, and in [12] by Lochert et al, and analytical dissemination through vehicle-to-vehicle propagation proposed by Wu, Fujimoto and Riley [13] as well as on some recent findings on practical systems as in [14] [18] by Lee et al, and Mahajan et al respectively, for pull-based diffusion activities. The proposed cache-and-forward replication scheme targets the availability of requested resources by using an index-based mechanism which will enable the selection of the *MI* in a formed cluster *L* (as in [1]). The following section explores the passive synchronized mobility model in the end-to-end path and presents an analytical model for the end-to-end capacity estimations.

III. CACHE AND FORWARD COOPERATION SCHEME USING SYNCHRONIZED MOBILITY AND CAPACITY CONSIDERATIONS MODEL

A. Communicating scheme in Vehicle-to-Vehicle communications

The interactions with roadside equipment for exchanging resources can be characterized accurate, whereas most vehicles have restrictions in their range of motion. As an example vehicles cannot follow other vehicles and they are subject to constraints for following a certain 'caravan' of vehicles in a highway. Therefore resource sharing can be performed using any web technology -pure Infostations

approach [1], [2]- available in the car. According to recent literature [15] [16] for a better delivery ratio and in order to reduce broadcast storms, a message has to be relayed by a minimum of intermediate nodes to the destination. In order to have this achieved, nodes are organized on a basis of a set of clusters, in which one node or more (Cluster Head) gathers data in his cluster and send them after to the next cluster. By using cluster-based solutions for disseminating the requested information into a "locally" limited number of nodes (that potentially will maintain their connectivity for time t (according to the path followed in the roadside)), these solutions provide less propagation delay and high delivery ratio with also bandwidth equity. In [14] the authors use a distributed clustering algorithm to create a virtual backbone that allows only some nodes to broadcast messages and thus, to reduce significantly broadcast storms. As recent studies have reported that intermittent network connection is inevitable for mobile users on a daily basis [15] [16] [17] have also shown that network capacity can be increased dramatically by exploiting node mobility as a type of multiuser diversity. As a result the opportunistic nature of the ad-hoc connections can be useful for "virtually" extending the coverage of wireless communications by using the notation of the cache-and-forward replication policy. There are two distinct approaches that allow mobile users to request and transfer data on the go: namely, the cache-based approach or the static Infostation-based approach. Cache-based approaches facilitate mobile file transfer by prefetching popularly requested files (e.g., commercial ads, movie trailers, and song previews) to a local storage. In this work the cluster's local storage is provided with a role that is being utilized and processed by any node within the virtual cluster. The Infostation-based approaches, on the other hand, support on-demand FTP requests by deploying dedicated servers as bridges between the Internet and mobile networks. However, the capability of these approaches for mobile file transfer is limited because they are basically centralized and have non-autonomic control over the resource sharing process, and as a result they fail to exploit and reflect the diversity and properties of network mobility.

This work uses the cache-oriented apparatus to facilitate mobile resource sharing/file transfer by prefetching popular requested files for a certain time duration onto any other nearby node within a specified number of hops. A significant aspect of the reliability and the availability of the requested resources in wireless systems, is the sudden partitioning of the connectivity, namely intermittent-connectivity experienced by nodes. This work differs from [1] in the sense that it combines the strengths of cache-based approach (i.e., prefetching the most likely requested files to a local storage, Figure 1) in an adaptive way by considering a window size model for estimating scheduled retransmissions of the requested file chunks. In addition the model considers the relay epoch and the mobility considerations for each node in the k-hop path for efficient end-to-end dissemination, and to further utilize the opportunistic communication in a reliable manner.

In order to avoid any sudden -unpredictable- network partitioning problems and prevent the exchange of any requested information a cache-based approach replication policy is used. Requested object replication [12] [27] and replicas redundancy [9] face the requests' failures whereas they create severe duplications. However these approaches aggravate the capacity of the end-to-end path whereas, as the path remains relatively small in terms of hops, these approaches can face the resource sharing problems adequately. On the other hand if the path is 'long' hosting many hops then the redundancy of duplications of the requested packets, aggravates the system's performance geometrically with the number of hops [24]. Considering all the extracted factors above this work enables the resource sharing process to be performed via the the Passive Opportunistic Synchronized Approach (POSA) [1] using a certain likelihood for this resource synchronization. Figure 2 shows the vehicular P2P resource sharing process using the push and pull procedure in a certain path. The scheme uses the Passive Opportunistic Synchronized Approach in order to share resources and enable duplications to nearby requesting nodes(nodes that are requesting certain resources/i.e.,file).

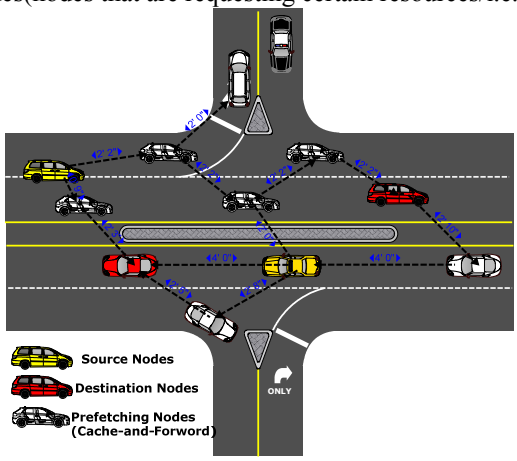


Figure 1. Cache-and-Forword (Prefetching configuration) of popularly requested file chunks for a certain time duration onto any other nearby node within a specified number of hops for Vehicular MP2P devices while moving in probabilistic paths.

Any vehicle/device can communicate directly with any other vehicle within the transmission range of each device. Therefore on a hop-by-hop basis each vehicle can push and pull information using the transmission channel of each node that it communicates. The system environment is assumed to be a dynamically changing MP2P network where mobile hosts access data items held as originals by other mobile hosts. Data items are periodically updated and ranked according to the ranking criteria of the intercluster and intracluster requests [8] [21] [28]. Each mobile host creates replicas of the data items which were highly ranked, and keeps the replicas in its memory space. When a mobile host issues an access request for a data item, the request is considered as successful in either case: (a) the request issue host itself holds the original/replica of the data item or (b) at least one mobile host (which is not directly necessarily connected) has the file (packets) of a replica of it. Figure 2 also shows the proposed vehicular MP2P *push* and *pull* procedure where the *i-th* vehicle is assigned as MI and can pull requested resources to *i-1, i-2, i-3, i-k*, whereas the vehicle which the MI follows can then push any of these

resources to the *i+1, i+2, i+MI_k* vehicle (dash lines denote the push procedure which takes place and solid lines denote the pull procedure). Both procedures take place until the next and preceding MI is reached while *i-th node* is sharing resources, respectively. These notations can also be seen in a more clear form in the Figure's 5 pseudocode, which shows a single step for the vehicle's MI transition.

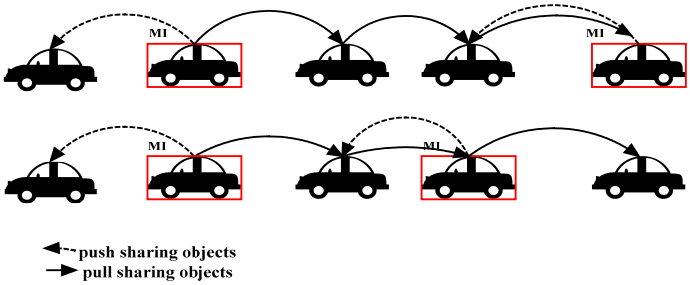


Figure 2. The push and pull configuration for Vehicular MP2P devices while moving in predetermined paths.

In order to reduce significantly the rendandancy of replicated packets, the HyMIS [3] [8] scheme is used where the primary Infostation is non-static (PI) but can move as the pathway allows according to certain likelihood. The PI is called Mobile Infostation (MI), and enables recoverability for any requested object in the end-to-end path while it maintains the sharing reliability. As vehicles are moving from one direction to the other the *i-th* vehicle (MI) can pull requested resources to *i-1*, *i-2*, *i-3, i-k*, where *k* is the number of peer vehicle in the end-to-end path requesting resource R_i. As Figure 4 shows, a cluster of replicated objects upon creation, considering the trajectory of the synchronized devices through their mobility and the likelihood in accessing the certain path using cluster interactions, the update rate of the requests, the cost of the *n-hop* replications.

Let k-hop path be the path that follows a certain $\vec{d}$ direction. Then $\vec{d}'$ notation consists of the converse direction of $\vec{d}$, and the path namely j-hop path. As the problem of saturation was faced in work done in [1], this work in order to avoid any saturation issues of the non-ending replications, certain criteria were set for all the requested high ranked file chunks as in [30]. In addition, it considers the opposite lane j-hop combination-comparisons of file chunks' requests with the requests of all the nodes in the j-path (of the converse pathway/Cluster C_j). Considering that the above scenario is used in real-time like in a vehicular raw-lane network, where requests of the j-hop may have a different direction, then it stands that for $Rank(i_N)$:

$$Min[Rank(i_N)]\forall N \notin C_i \quad (1)$$

where (1) is minimized for the node that the resource was downloaded at least once or when the distance d (Figure 5 pseudocode) is over a certain threshold D_{thress} from the k-hop peer- which means that the requested resource(s) set on this node have been redirected to any other path. Equation 1 sets the rank of the node containing the requested resource to minimum, for the nodes that are not members of the cluster where the resource was requested iff *d* is over a certain threshold D_{thress}. If the D_{thress} increases then the resource is isolated and it no longer belongs to the C_i. This enables the prevention of huge duplicated information delivery, whereas it considers the nodes which are located far from source node and to maintain only the j-hops duplications-after the performed comparisons- avoiding redundant transmissions.

Considering the k-hop scenario of Figure 2, the evaluated duration of the requested file chunks is evaluated as follows:

$$C_d = k \cdot E_i \quad (1.1)$$

where C is the caching duration that is allowed for node *i* and E_i is the relay epoch according to the number of hops permitted [3]. Therefore, it stands that the greater the number of hops, then the greater the time duration that is allowed to be achieved. The delay epoch duration is modelled according to the derivation of the *Definition 1.1*, taking into account the hop-count path, ping delays and the total delays from the end-to-end perspective.

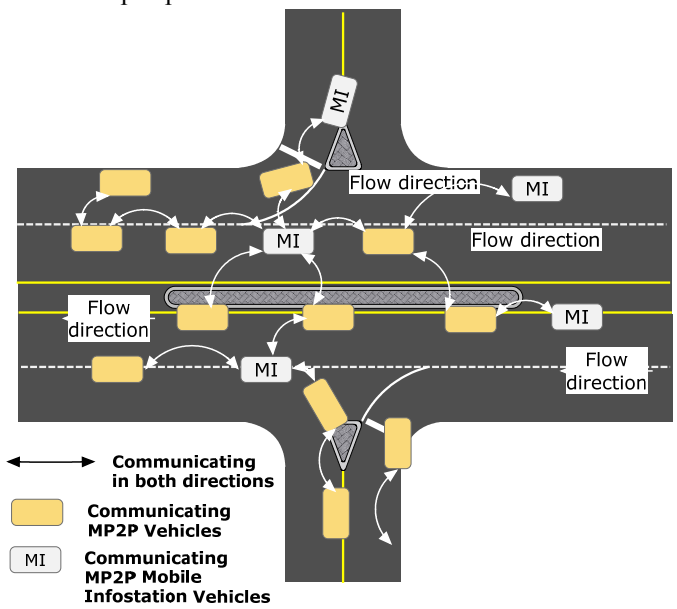


Figure 3. File chunk's blocks and segments are replicated using the HyMIS scheme in order to enable resource availability. High ranked resources that are requested are replicated in the cluster in order to be available according to ranking requests.

Definition 1.1 (Dense population region): We consider the concept of dense population region of a certain transmitter–relay node pair (u,w) traversing n paths/clusters is defined as the relay region which has any end-to-end connectivity in the relay path at a given time as follows:

$$\Delta R_{u\to w} = \{(x,y,u) \in \Re^2 : W_{u\to w\to(x,y,u)} > P_{u\to(x,y,u)} \forall u \in P_n\}. \quad (1.2)$$

Thus in an end-to-end path $\forall u \in P_n$ the minimized ping delays between the nodes in the end-to-end path the minimized evaluated delay is according to the:

$$d_p = Min\sum_{i=1}^{n} D_i \quad (1.3)$$

where Di is the delay from a node i to node j, and d_p is the end-to-end available path. Therefore the delay epoch $E_{i(t)}$ of each node is defined as a function of the number of created replicas on the j-hosts as follows:

$$E_{i(t)} = d_{r_{i\to j}} \cdot {}^{r_{i\to j}}\!/_{Total\_d_{r_{i\to j}}} \quad (1.4)$$

where D is the delay via the ping assigned durations, $r_{i\to j}$ is the number of replicas form node *i* to *j* in the *j-hop* path and $Total\_d_{r_{i\to j}}$ is the total duration that all the requested replicas can be downloaded from the j-hop path.

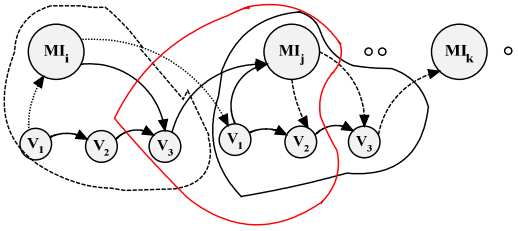


Figure 4. The Replication policy with respect to the different Clusters which are created on-the-move using the Mobile Infostation (MI) model for the Inter-cluster outsourcing, according to the ranking requests.

```
Set communication Path(A,B, N)
{

If (MI criteria meet==TRUE)
  Set MI in the Path(A,B, N);
else
   form Path(A,B, M)  ∀M_i ∈ N
 //Inter-cluster info
  For any request in the C_i split chunks into
n blocks and do {
  { Check_delay_epoch(peer A, peer B)
while
((communication==1)&&(D_delay_epoch()==valid))
{
   If delay criteria meet
       { if ((exists(A)==LOCAL_CLUSTER)&&
            Measure_delay_epoch(C_i)==valid)&&
            (Rank(i)==Max(Table(res))) {
            If (Total_delay_epoch<D_thresshold){
      while ((file_chunk==EXISTS)&&
  (TOTAL_delay()<Tlim(filechunk))){
    if (duration_interaction(node A, node B,
range(true, duration=true))==valid)
      // K_valid ∈ C_N
         {
           Check_CLUSTER();  //initialize  the
intracluster chunk sharing
push_requestedObj(Ob_id, Cap, Peers,
estimated_delay);

}
```

Figure 5. Pseudocode for a pull-based procedure by the vehicle's MI transition in order to enable object replication placement scheme between synchronized moving peers.

B. *Multi-hop probabilistic mobility model and user's capacity in the end-to-end path*

Resources availability problems can be also faced using a local summary of the global system-or clustered information for the subsystem- by using the property of aggregation in distributed systems concept introduced by Renesse et al in [14]. MP2P systems require to guarantee the availability any requested resources as well as to enforce appropriate access control policies. In our application scenario, we assume that a common look-up application is being used in order to enable nodes to interexchange locally the requested information objects. As a starting measure we estimate the synchronized cooperative movements of each vehicle by measuring the motion performed while measuring at the same time the reserved capacity by each vehicle. Since vehicles are moving in an organized and in a predictable way, the pull and push model aggravates the capacity of each device, as in a MP2P environment. Through the proposed resource exchange scheme for Vehicle-to-Vehicle communications as well as through the additional parameters that are being considered (like the evaluated end-to-end relay epoch/latency, the mobility pattern and the time frame for the allowed promiscuous caching introduced in [21] by Mavromoustakis), the proposed model enables efficient capacity manipulation in the end-to-end relay region and efficient data manipulation in the intercluster communication.

By adopting the modified scheme of Mavromoustakis and Karatza [8] and by assigning the role of MI to be adjusted into the vehicular devices, the PI and MI are being implemented by a certain frontal vehicle, where only unidirectional sharing and connectivity occurs.

When mobility is considered, the design of efficient rendezvous data dissemination protocols is complex for enabling efficient manipulation and availability of resources, whereas the existing solutions do not consider the random probabilistic movements of devices while disseminating data. In order to measure the direction movement we enable a probabilistic model for the direction of the movement of each device. Each device is associated with a random variable which represents the direction movement. For the motion, this work considers a probabilistic Random Walk in a predefined pathway represented as a Graph (G) where this G enables as a random variable the weights of these random movement. A device can perform random movements according to the topological graph G = (V,E) where it comprises of a pair of sets V (or V(G)) and E (or E(G)) called vertices (or nodes) and edges (or arcs), respectively, where the edges join different pairs of vertices. This work considers a connected graph with *n* nodes labeled *{1, 2, . . . , n}* in a cluster L^n with weight $w_{ij} \geq 0$ on the edge (i, j). If edge (i, j) does not exist, we set $w_{ij} = 0$. Each node moves from its current location to a new location by randomly

(probabilistically) choosing an arbitrary direction and speed from a given range. Such a move is performed either for a constant time for a constant distance traveled. Then new speed and direction are chosen. According to the probabilistic Mobility model, the mobility is described as a memoryless mobility pattern. This occurs because it retains no knowledge concerning its past locations and speed values. In this work a Probabilistic optimized approach of the Random Walk Mobility Model is used, as in [27] by Ibe. In this model 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 the movement. This leads to a walk that leaves the starting point much faster than the original random walk model. Given that the device/vehicle is currently at node *i*, the next node *j* is chosen from among the neighbors of *i* with probability:

$$p_{ij}^{L} = \frac{w_{ij}}{\sum_{k} w_{ik}} \quad (2)$$

where in (2) above the p_{ij} is proportional to the weight of the edge (i, j).

Node mobility impacts the effectiveness of opportunistic resource sharing process. Previous studies have shown that the overhead carried by epidemic- and/or flooding-based routing schemes can be reduced by considering node mobility in a probabilistic manner as Section B presents. For instance, The proposed scheme takes the mobility pattern into account—that is, a message is forwarded to a neighbor node if and only if that node has a mobility pattern similar to that of the destination node. This is performed according to the probabilities explored in (2), following the probabilistic model for the direction of the movement of each device. Thus, if a device follows another device with a certain probability (Figure 6) where the device followed, follows it turn another device, this obeys to the following equation:

$$P_{i\to N} = norm\left[\frac{1}{N-1}\sum P_{i,j}\right]^{0..1} \forall P_{i,j} > P_{Thress} \quad (2.1)$$

where should be over the range of values for $P_{i\to N} > P_{Thress}$. After consecutive experiments the values of $P_{Thress} > 0.31$ is found to be ideal in contrast to the density of the devices and the topology formation factor mentioned in [21]. This evaluation is performed in order to enable probabilistic resource sharing among users taking into consideration the probabilistic mobility and the effects of the total aggregation of this likelihood in a normalized manner as equation 2 presents.

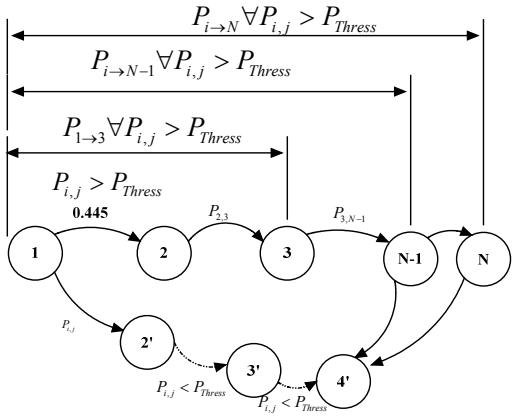


Figure 6. Probabilistic pathway followed by devices in the same path.

Therefore, if the threshold is satisfied then the replication takes place according to the replication policy explored by the HyMIS [3] and follows the limitations set by the adaptive replication scheme using the Go-back-N caching cooperation presented in the next Section.

C. Cooperation and storage model using passive message ferries and bi-directional resource's replications

In order to define which requested objects should be outsourced onto preceding m-peers a ranking model has been applied as follows: To find the rank of an object $a1\ a2\ .\ .\ .\ a_m$, one should find the number of objects preceding it. It can be found by the following function:

function rank($a_1, a_2, \ldots, a_m$)
rank← 1 ;
for $i \leftarrow 1$ **to** m **do**
for each $k<a_i$
$rank \leftarrow rank + N(a_1, a_2, \ldots, a_{i-1} \mid k)$

Then the new ranked sequence of shared objects will cache onto other nodes in the path the first *i*-requested objects in regards, to the given for each node, *k* parameter, where *k* is defined as a function of the remaining capacity onto each device as:

$$|k| = \inf\left(\frac{\sum_{N=i}^{N}(1-\rho_N)}{N}\right) \quad (3)$$

where ρ_N is the utilized capacity and N is the number of hops in the requested path. Nodes in the path are moving according to the 2-D plane mobility model $L \subset \Lambda, \Lambda \subset \Re^2$. A moving square (the $\{\Lambda_1, \Lambda_2, \Lambda_3, ...\}$ bounded area) is divided into multiple sub-squares, called cells as in [1], and time is divided into slots of equal duration. At each time slot a node is in and can be only in one cell. The initial position of a node is uniformly chosen from all cells. At the beginning of each time slot, the node jumps from its current cell to one of its adjacent cells with equal probability. Two mobile nodes can communicate with each other whenever they are within a distance of *d*, the transmission range of the mobile node. In order not to have an optimistic assumption a

low density population network is assumed with regards to the number of traversing nodes per Λ_i. We assume that no conspiracy policy exists where, nodes somehow conspire together not to meet each other forever and move at *d>D* and in parallel.

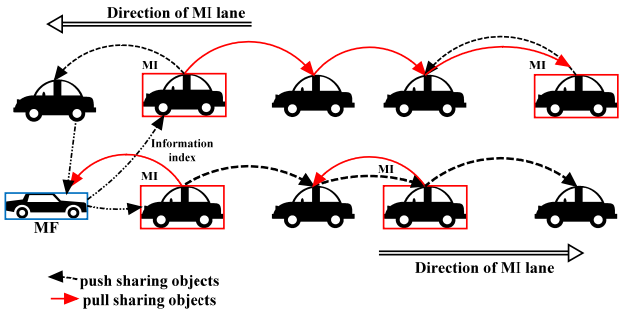


Figure 7. Passive message ferries where any other device can play the role of the messanger regarding the information index.

The *index* of each node is being transferred using the message ferries that are passively passing from any other pathway within the distance of communication range of each device. Figure 7 shows this approach where the message ferries are passing from an opposite pathway whereas at the same time they are in the transmission distance range with each device they communicate. As a result the MFs are forwarding information from one node to another by using an opposite lane and pathway. This configuration enables the bi-directional resource replication of high ranked requested resources.

Taking into account the delay characteristics, let N be the number of source peers in the network (*N* different end-to-end paths) and $C_i(t)$ be the service capacity of source peer *i* at time slot *t*. An end-to-end download can be then depicted as a function of time as derived from Chiu and Young Eun in [16] and the $w^L{}_{ij}$ of the end-to-end path in the cluster L as:

$$T = \min\left\{ s_{ij} > 0 \,|\, \sum_{t=1}^{s} C(t) \geq F \right\} \tag{4}$$

where *F* is the file capacity defined as $\{f_1, f_2, f_3, f_4, \ldots f_n\}$ equi-divided file chunks and *s* a given end-to-end bounded allowed delay for this file to be downloaded from any numbers of peers in the end-to-end path. The obtained eq. (4) derived from Wald's equation introduced by Ross in [17] can therefore be expressed as:

$$F = \mathrm{E}\left\{ \sum_{t=1}^{T} C^L(t) \right\} = \mathrm{E}\{C^L(t)\}\mathrm{E}\{T\} \tag{4.1}$$

where we can easily extract the slotted amount of file chunks that are shared in the end-to-end path. The $A(\vec{c})$ is the minimum average capacity offered by each link in the path as:

$$A(\vec{c}) = \frac{1}{N}\sum_{1}^{N} \inf(C_{ij}(t)) \tag{4.2}$$

where $A(\vec{c})$ is the requested and available arithmetic mean for the capacity in the path. The average capacity offered by the end-to-end path considering all the links in the path of the requested file F, can be denoted as $A(\vec{c}) \Big/ \mathrm{E}\{C_{ij}{}^L(t)\}$.the average download time is:

$$\mathrm{E}\{T_{F_{ij}}\} = \frac{F_c}{A(\vec{c})} = \frac{N \cdot F_c}{\sum_{1}^{N} C_{ij}(t)} \forall w_{ij} \in L \tag{4.3}$$

while it stands that for $C_{ij}(t) = \min(\inf(C_{ij}(t)) | \mathrm{E}\{T_{F_{ij}}\}$.

Let $t_\lambda = \max(\Theta_{MI,j})$ be the contact rate estimation and $\Theta_{MI,j}$ is the estimated contact time between MI and a moving node *j*, then it stands that a vehicle remains as a MI in the path if the following is satisfied:

$t_{\lambda_{ij}} \geq \frac{A(\vec{c})}{BW_{ij}}$ where $t_{\lambda_{ij}}$ is the contact rate in the path between i,j and BW_{ij} is the associated bandwidth in the path between i,j. The estimation of $t_{\lambda_{ij}}$ is essential since it can determine the time that a mobile node can remain as a MI.

1) Adaptive replication scheme using the recursive Go-back-N caching cooperation

One important aspect in P2P vehicular communication is the limitation of the redundant replications that are performed while requests are taking place in the peer-to-peer connectivity. When resource sharing process occurs in a region, the packets that are sent are considered to have a bounded time delay τ to reach any specified destination. This work proposes the utilization of a methodology that enables the replication of the requested file chunks using a specified window size. The proposed method uses the adaptive precision Go-back-N caching cooperation where the number of files N that were requested can be re-selected to be securely replicated onto node according to the contact criteria using the random walk framework and the $P_{i,j} > P_{Thress}$ (subject to equation 2.1). Assuming a file which consists of N chunks as follows:

$$N_{chunks} = \{1, 2, 3, \ldots n\} \tag{5}$$

Where N_{chunks} are the chunks that are selected according to the contact and request rate and depicted as missing file chunks according to the movements and the state as: (k, $\Delta\xi_k$), where k is the location of the walker, where a move to

a neighbor indicates a success (increase the $\xi_k \forall P_{move}$) and a move to a $d_t<d_{t+1}$, where d_t is the distance between two nodes in current contact, indicates a failure (decrease the $\xi_k \forall w_{ij}$); and $\Delta\xi_k$ is the result index, which is defined by $\Delta\xi_{ij}(t) = \frac{dij}{D_{ij}}$

where d_{ij} denotes the number of hops in the path for node i to j and D_{ij} depicts the total number of hops in the end-to-end path. Therefore the increase and decrease of the likelihood for the Correlated Random Walk is respectively $\xi_+ = \frac{P_{move(i\to j)}(t-1)+\Delta\xi_{ij}}{1+\Delta\xi_{ij}}$ and $\xi_- = \frac{P_{move(i\to j)}(t)}{1+\Delta\xi_{ij}}$.

$P_{move(i\to j)}$ is the probability of moving in the *i* to *j* path, and (1- $P_{move(i\to j)}$) of abandoning the path and the cluster C_i. This enables the estimation of whether a mobile node will follow a certain pathway or not. The mobility is modeled according to the Section B (Equation 2) where the replication policy follows the Equation 2.1.

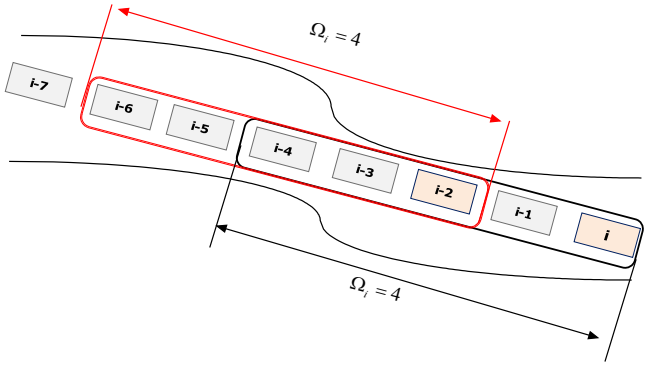


Figure 8. The Adaptive Go-back-N caching cooperation scheme with respect to the number of file chunks that the device-i requests, and considering the missing chunks that should be reached in order to be available to the requests. The Adaptive Go-back-N caching enables the parameter to be adaptively tuned in order to enable the recoverability in the resource exchanging cluster, where –in any other case- the missing chunks will affect the end-to-end completion of the file.

Taking further into consideration that the request rate is incremental and satisfies the:

$$\sup(R_i(\tau,\rho_i)) > \sup(R_i(\tau-1,\rho_j))\forall i \neq j, \tau > \tau - 1 \quad (5.1)$$

where R_i is the request rate for the certain point/location ρ_i for the time τ. The proposed scheme adaptively calculates the N cached chunks using the:

$$\Omega_i = R_i \cdot L_i + R_i \cdot L_t \text{ for time } t \quad (5.2)$$

where t is set in $[\tau, \tau + t]$ and R_i is the request rate for the certain point/location, L_i is the delay length between interarrival time of the requested file chunks, and L_t is the delay length of the repeated requests of the missing file chunks. The parameter Ω_i represents the number of file chunks that the *i* requests and should be outsourced to nearby neighbors in the cluster path. Figure 8 shows the related concept for $\Omega_i = 4$.

D. Considering contact interactions for collaborative streaming

In this section we propose a number of social interaction parameters which take place in collaboration with the file chunk outsourcing of the previous section. The metrics are community-oriented and are considering the number of created clusters $C_N(t)$ in a specified Relay region of a certain transmitter–and a number of receivers (1, N] under the relay node pair (u,w |MI_i) -as a modified definition of [18]- as follows the:

$$C_N(t) = \frac{2|h_N(t)|}{|I_{C(N)}(t)|\cdot(|I_{C(N)}(t)|-1)}, \text{ iff } P_{u\to w\to(x,y)} > W_N(t) \quad (6)$$

where *W* is the Community streaming factor and is defined as the number of existing communities in the intercluster communicational links at a given time instant. The $h_N(t)$ is the number of hops in the existing clusters and the $I_{C(N)}(t)$ is the number of interconnected nodes N in the cluster $C_N(t)$. *W* can be defined according to the download frequency of the file chunks in the intercommunity as follows:

$$W_N(t) = \frac{DldRate \cdot \#\, sharingChunks}{Total\#\, dlds(t) \cdot \#\, inactiveChunks} \quad (6.1)$$

where in (6.1) the download rate is considered in contrast with the number of chunks being shared in a specified instant time *t*.

IV. Performance Evaluation, Experimental Results and Discussion

A. Dedicated Short Range Communications (DSRC)

To emulate the scenario described earlier, a possible realistic environment must be achieved. Dedicated Short Range Communications (DSRC) was used for the evaluation of the proposed scenario which is two-way short- to medium-range wireless communication channels specifically designed for automotive use and utilizes a corresponding set of protocols and standards [19]. Considered to be short to medium range communication technology it operates in the 5.9 GHz range. The Standards Committee E17.51 endorses a variation of the IEEE 802.11a MAC for the DSRC link. DSRC supports vehicle speed up to 120 mph, nominal transmission rage of 300m (up to 1000 m), and default data rate of 6 Mbps (up to 27 Mbps). This will enable operations related to the improvement of traffic flow, highway safety,

and other Intelligent Transport System (ITS) applications in a variety of application environments called DSRC/WAVE (Wireless Access in a Vehicular Environment). In the evaluation of the proposed scheme we evaluated the Peer-to-Peer/Ad hoc mode (vehicle–vehicle) scenario and took into account the signal strength parameters and 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 for the wireless devices, it is necessary to apply efficient routing algorithms to create, maintain and repair paths, with least possible overhead production. The proposed scenario uses the trajectory based routing (i.e., SIFT) [26]. The number of nodes varies depending on the mobility degree and the distance variations of each user within a connectivity scope. The user's transition probability arises from a specified location where certain information is pending to be received by this user. In this way the likelihood varies based on the demanded by users, requested resources.

B. *Simulation results of the proposed scenario and discussion*

In this section, we present the results extracted after conducting the discrete time performance evaluation through simulation of the proposed scenario. The simulation used a two-dimensional network, consisting of 250 nodes dynamically changing the topology on a non-periodic basis (asynchronously as real time mobile users do). It stands that after random time each node moves at a random walk to one of the possible destinations (north, east, west, south) in an organized vehicular way. Each link (frequency channel) has max speed reaching of 4*Mb per sec,* according to the regional EU standards of the DSRC. The propagation path loss is the two-ray model without fading. The network traffic is modeled according to the traffic sources modeled modules of the ns-2 simulator [29]. Initially there is a transient and initialization stage that is responsible to extract the resources (i.e., certain files) requests among users. All mobile nodes collaborate via a shared application that uses a distributed look-up service, for the shared resources. Radio coverage is small, and is assigned according to the DSRC specifications, whereas, nodes cannot contact each other directly when sharing resources. Each source node transmits one 512-bytes (~4Kbits) packet. Packets during initialization period are generated at every time step, destined for a random destination uniformly selected. Nodes have at any time incoming file-sharing requests by other peers whereas, during the time of the requests, the details of the requested resources are being transferred to any potential destination node. In addition, the modeled simulation environment considers the probabilistic mobility [1] taking also into account the traffic lights' patterns and the randomized vehicular events that may occur. The results show that the proposed bi-directional scheme significantly outperforms previous approaches in all test cases, while its traffic overhead remains moderate and the nodes have generated adequately enough replications for the peer-devices, avoiding in this way the redundancy.

Figure 9 shows the network dimensions with the data and capacity exchanged through the created clusters. Figure 9 shows that even when the files that are being exchanged are greater than the network dimensions, the system can handle more than one resource per user. This result outperforms the previous findings where the dimension of the network bottlenecks the number of resources that are interexchanged in the moving cluster. The proposed scheme effectively handles the end-to-end transmissions and enables the complete download whereas, for this evaluation, two measures were taken into consideration: the data exchanged within the cluster *i* and the data exchanged with other clusters both versus the capacity limitations (with limited capacity and unlimited capacity onto users devices).

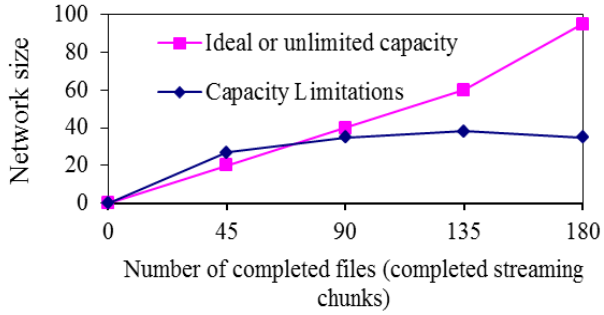


Figure 9. Indication of Network dimensions with the data and capacity exchanged through the clusters formed.

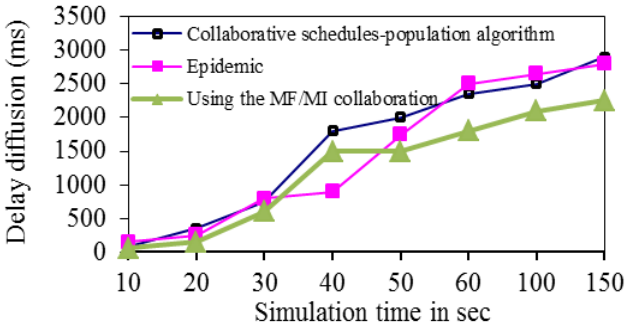


Figure 10. The delay of the diffusion outsourcing process with the simulation time compared with Epidemic and collaborative schedules schemes.

Figure 10 shows the delay of the diffusion outsourcing process with the simulation time compared with two different in implementation schemes: the epidemic and collaborative schedules schemes. It is easily spotted that Figure 10 shows the better performance of the proposed scheme for this specific scenario in vehicular P2P systems. It presents also the effectiveness with the significant robustness in the delay diffusion process-which is further minimized. Figure 11, presents the number of successfully received transmissions over of total of 25 transmissions in the path/clustered end-to-end transmission, with the mean number of sessions created in the system. Figure 12 shows the number of transmitted packets with the number of lost packets for three different traffic classifications: heavy, moderate and light traffic. The proposed scheme shows that it can successfully treat the traffic with limitations and transmission deadlines accurately by minimizing the losses in regards to the number of transmitted packets. The number of participating nodes with

the number of high ranked resources over the timeline's limitation of the transmission is shown in Figure 13, whereas Figure 14 shows the performance of the bi-directional scheme in contrast to other existing schemes like the Most Recently Claimed and the passive/generic caching. The latency for the proposed scheme outperforms of the other compared schemes, in regards to the requested file chunks and the number of created replicas onto N-hop nodes. In Figure 15 the VBR requests per cluster with the successfully shared capacity is presented, whereas Figure 16 presents the average available capacity per user with the successfully shared MBs, depicting the efficiency of the proposed scheme in handling 250 users concurrently. It should be noted that the exchanged resources cannot be unranked and this is due to the availability of requested resources which should be kept consistent, unless a low ranked resource will be highly outsourced and becomes highly demanded. Figure 17 shows the number of users sharing resources with the capacity that is shared in for different users' capacity measurements. Figure 17 depicts the response of the system in contrast to the shared requested capacity and the storage capability of each user.

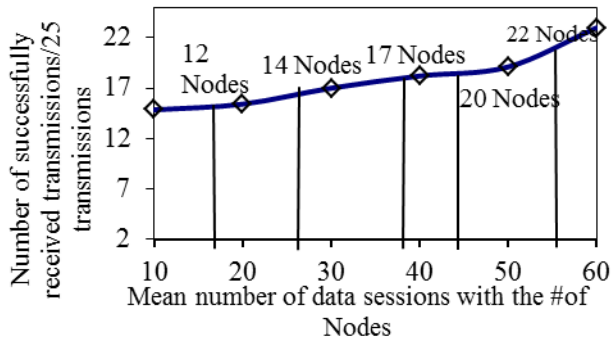


Figure 11. The number of successfully received transmissions over of total /25 transmissions with the mean number of sessions created in the system.

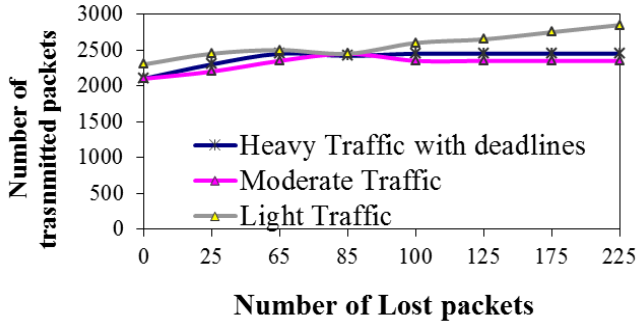


Figure 12. Number of transmitted packets with the number of lost packets for three different traffic classifications: heavy, moderate and light traffic.

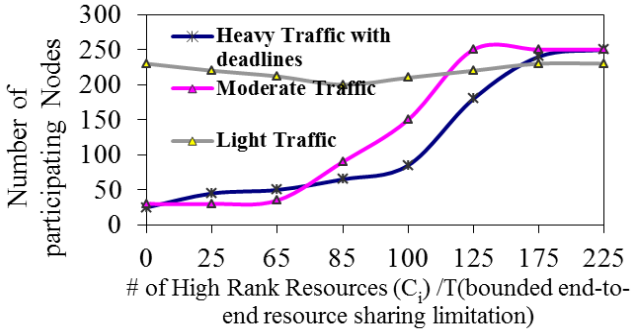


Figure 13. Number of participating nodes with the number of high ranked resources over the timelines' limitation of the transmission.

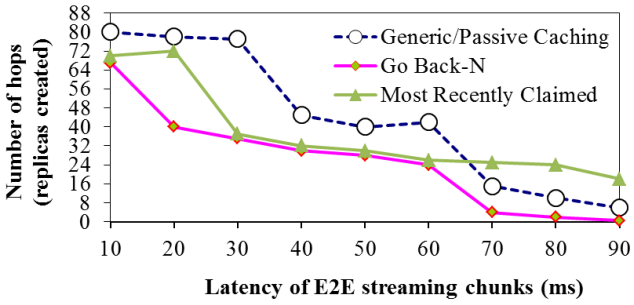


Figure 14. The latency of the streaming requested file chunks and the number of created replicas onto N-hop nodes.

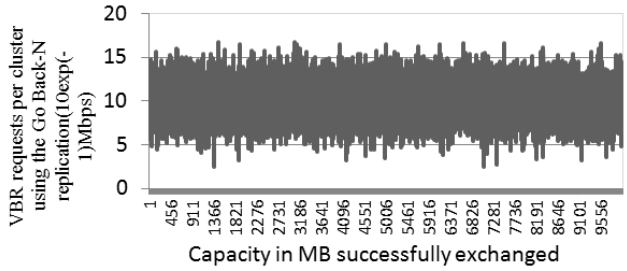


Figure 15. VBR requests per cluster with the successfully shared capacity.

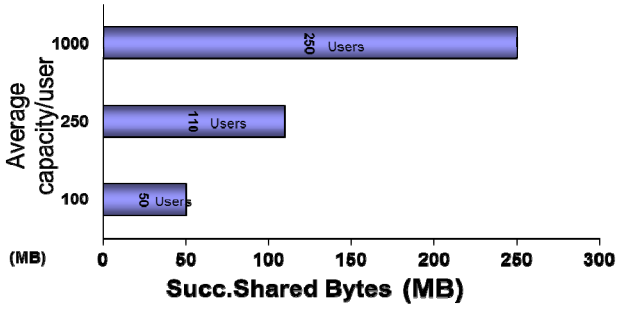


Figure 16. Average available capacity per user with the successfully shared MBs.

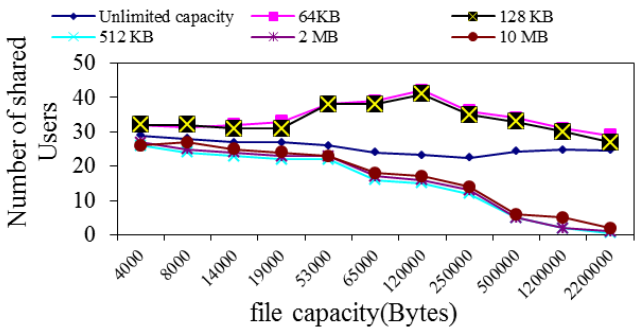


Figure 17. Number of users sharing resources with the capacity that is shared in for different users' capacity measurements.

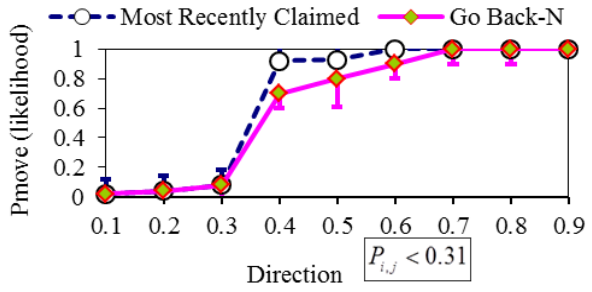


Figure 18. Likelihood of the movements with the direction of each device according to the modeled estimations.

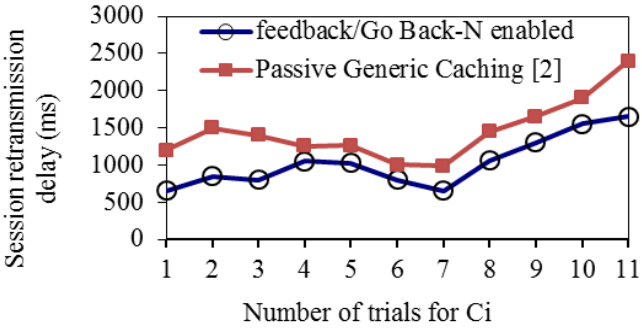


Figure 19. Number of trials with the session retransmission delay in msec.

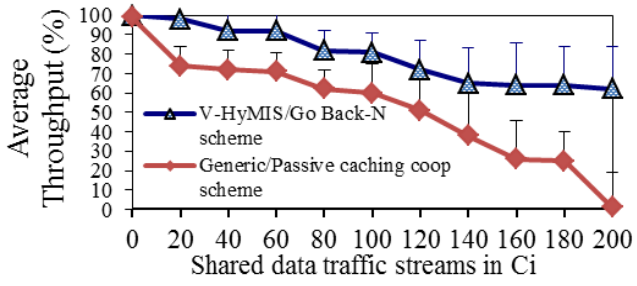


Figure 20. Performance of the Cluster with Bidirectional outsourced traffic requests compared with passive caching scheme.

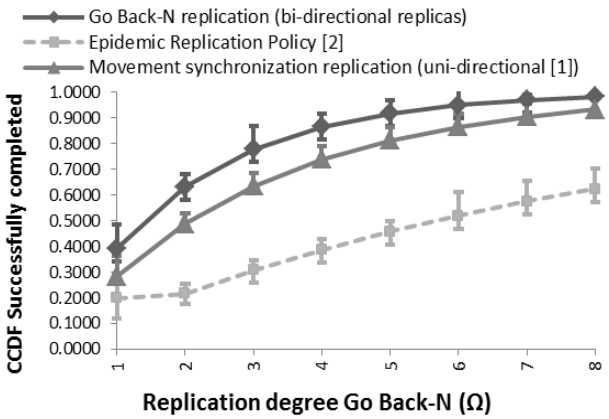


Figure 21. Complementary Cumulative Distribution Function (CCDF or simply the tail distribution) with the replication degree of Go back-N for successfully transmitted packets and completed downloads.

The likelihood of the movements with the direction of each device according to the modeled estimations is shown in Figure 18. In Figure 18 the likelihood corresponding to 0.31 is considered after consecutive simulations as the ideal to set it as a threshold for the movement of each node. Any value for the likelihood below this value indicates that the device will soon change a direction and will not follow a synchronized motion with other devices. Figure 19 presents the number of trials with the session retransmission delay in *msec*, where the trial's measure corresponds to the intra-cluster retransmissions with the associated delay measures. In Figure 20 the performance of the Cluster with Bidirectional outsourced traffic requests compared with passive caching scheme is shown. Finally, Figure 21 shows the Complementary Cumulative Distribution Function (CCDF or simply the tail distribution) with the replication degree of Go back-N for the successfully transmitted packets and completed downloads. The distribution of successfully shared resources can be adequate even if the replication of resources using the Go back-N is kept low. During the simulation process we have extracted different confidence intervals for the three compared schemes where, these are shown in Figure 21. The proposed replication policy exposed significant improvement for the reliability degree, in contrast to the Epidemic replication proposed in [2] and the work done in [1]. It is undoubtetly true that Figure 21 shows that by using the Go Back-N in a bi-directional way, the CCDF for resource sharing can be significantly improved along with the reliability degree. In addition, it also presents the reliability increment, by using the CCDF tail distribution of the successfully completed exchanged resources, when compared with similar schemes for Vehicular-based resource exchange.

V. CONCLUSION AND FURTHER RESEARCH

This work extends and re-considers, under a different approach in the resource sharing policy, the work done in [1] in a recursive bi-directional replication manner in order to promote resource availability among moving devices. The resource assignment policy takes into consideration the

number of synchronized peers in the resource sharing cluster, and assigns resources according to the modeled Go back-N scheme, for replicating the requested resources onto N nodes. The scheme takes into consideration the probabilistic synchronized motion of the nodes that are requesting resources. According to the likelihood of the motion expressed by the node, the requested resources can be outsourced and replicated onto other nodes in the range of communication. As the process of exchanging resources is characterized by bounded end-to-end delay, the scheme considers also the time frame that these ranked requested resources should be potentially completed; otherwise they are outsourced to nearby nodes in order to be available for future requests by other nodes. Finally the methodology encompasses the assignment of the resources and the cache-and-forward scheme by using and assigning the role of a Mobile Infostation (MI) peer to a certain vehicle whereas, this is done in a bi-directional way with the introduced Message Ferry (MF) mobile Peer. Passive message ferries are utilized as a resource index for the end-to-end path in order to efficiently enable delay sensitive streaming. Extensive simulation experiments that were conducted have shown that the proposed scheme improves the existing scheme, whereas in comparison with the Epidemic and the passive replication schemes it outperforms them in major performance estimations. Moreover, results have shown that the scheme offers high throughput and significant end-to-end reliable exchange of resources whereas it offers high SDR for completed files.

Current and future research directions include the modeling of the mobility pattern of the peers by using different stochastic approaches like the fractional Brownian motion taking into account the global requests and different network partitioning parameters.

ACKNOWLEDGEMENTS

We would like to thank the University of Nicosia Computer Installation Center, for helping us with the terminals' simulation software installation in order to perform exhaustive evaluations of the proposed scheme and extract results from different machines. These machines' runs significantly helped us to derive the confidence interval among extracted results.

REFERENCES

[1] C.X. Mavromoustakis, "Exploiting movement synchronization to increase end-to-end file sharing efficiency for delay sensitive streams in vehicular P2P devices", The Seventh International Conference on Wireless and Mobile Communications ICWMC 2011, June 19-24, 2011, Luxembourg.

[2] C.X. Mavromoustakis and H. D. Karatza, "A Gossip-based optimistic replication for efficient delay-sensitive streaming using an interactive middleware support system", IEEE Systems Journal, IEEE USA, Vol. 4, no. 2, June 2010, pp. 253-264.

[3] C.X. Mavromoustakis and H. D. Karatza, "On the large scale performance evaluation of an end-to-end V-HYMIS reliable streaming scheme under delay sensitive traffic with finite capacity and intermittent connectivity", Proceedings of the 27th Annual UK Performance Engineering Workshop, 7-8 July 2011, Bradford, UK, Springer Proceedings, pp. 226-240.

[4] S. Toumpis and A. Goldsmith, "Capacity regions for wireless ad hoc networks", IEEE Transactions on Wireless Communications, Vol. 2, No. 4, July 2003, pp. 736-748.

[5] D. Grobe Sachs, C. J. Hughes, S. V. Adve, D. L. Jones, R. H. Kravets, and K. Nahrstedt, "GRACE: A Hierarchical Adaptation Framework for Saving Energy", Computer Science, University of Illinois Technical Report UIUCDCS-R-2004-2409, Februrary 2004.

[6] D. Kliazovich and F. Granelli, "A Cross-layer Scheme for TCP Performance Improvement in Wireless LANs", Globecom 2004, IEEE Communications Society, pp. 841-844.

[7] M. Conti, G. Maselli, G. Turi, and S. Giordano "Cross layering in mobile Ad Hoc Network Design", IEEE Computer Society, February 2004, pp. 48-51.

[8] C.X. Mavromoustakis and H. D. Karatza, "Segmented File Sharing with Recursive Epidemic Placement Policy for Reliability in Mobile Peer-to-Peer Devices". Proceedings of the 13th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Georgia Tech, Atlanta, Georgia, September 26-29, 2005, pp 371-380.

[9] Z. Haas and T. Small, "Evaluating the Capacity of Resource-Constrained DTNs," International Wireless Communications and Mobile Computing Conference, pp. 545-550, July 2006.

[10] J. Liu and J. Xu, "Proxy caching for media streaming over the Internet," IEEE Commun. Mag., vol. 42, no. 8, pp. 88–94, Aug. 2004.

[11] T. C. Little and A. G. Agarwal, "An Information Propagation Scheme for VANETs," Proc. 8th Intl. IEEE Conf. on Intelligent Transportation Systems (ITSC), Vienna Austria, Sept. 2005, pp. 361-368.

[12] C. Lochert, B. Scheuermann, M. Caliskan and M. Mauve, "The feasibility of information dissemination in vehicular ad-hoc networks", appeared in Proceedings of the 4th Annual Conference on Wireless On-demand Network Systems and Services (Jan. 2007), pp. 92-99.

[13] H. Wu, R. Fujimoto, and G. Riley, Analytical models for information propagation in vehicle-to-vehicle networks. IEEE

Vehicular Technology Conference, 2004. VTC2004-Fall. 2004, Sept. 2004, Vol. 6, pp. 4548-4552.

[14] L. Bononi and M. Di Felice, "A Cross Layered MAC and Clustering Scheme for Efficient Broadcast in VANETs", IEEE MASS'07, October 2007, Pisa, Italy.

[15] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden. A measurement study of vehicular internet access using in situ Wi-Fi networks. In ACM International Conference on Mobile Computing and Networking, pages 50–61, 2006.

[16] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. Wireless Communication and Mobile Computing Journal, 2(5):483–502, 2002.

[17] A. Chaintreau, P. Hui, J. Crowcroft, C. D. Richard Gass, and J. Scott. Impact of human mobility on the design of opportunistic forwarding algorithms. In IEEE International Conference on Computer Communications, pages 1–13, 2006.

[18] K. Lee, et al. First experience with cartorrent in a real vehicular ad hoc network testbed. In Proc. of MOVE (2007).

[19] R. Mahajan, J. Zahorjan, and B. Zill, "Understanding wifi-based connectivity from moving vehicle" . In Proc. of IMC (2007).

[20] R. Van Renesse, K.P. Birman, W. Vogels, Astrolabe: a robust and scalable technology for distributed system monitoring, management, and data mining, ACM Trans. Comput. Sys. 21 (2).

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

[22] Y.M. Chiu and D.Y. Eun, "Minimizing File Download Time in Stochastic Peer-to-Peer Networks," IEEE/ACM Transactions on Networking, Vol. 16, No. 2, pages 253-266, April 2008.

[23] S. M. Ross, *Stochastic Processes*, 2nd ed. New York: John Wiley & Son, 1996.

[24] M. Fiore and J. Harri. "The networking shape of vehicular mobility". In Proceedings ACM MobiHoc, pp. 261–272, 2008.

[25] Dedicated short-range communications (DSRC) for wireless communication channels found on http://en.wikipedia.org/wiki/Dedicated_short-range_communications, last accessed and retrieved document on 25 Nov. 2011.

[26] H. Labiod, N. Ababneh, and M. García de la Fuente, "An Efficient Scalable Trajectory Based Forwarding Scheme for VANETs," aina, 2010 24th IEEE International Conference on Advanced Information Networking and Applications, 2010, pp.600-606.

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

[28] P. Santi and D. Blough, "The critical transmitting range for connectivity in sparse wireless ad hoc networks. IEEE Transactions on Mobile Computing 2(1), 2003, 25–39.

[29] NS-2 Simulator, at http://www.isi.edu/nsnam/ns/, last accessed on 20/12/2011.

[30] C.X. Mavromoustakis and H. D. Karatza, "Embedded socio-oriented model for end-to-end reliable stream schedules by using collaborative outsourcing in MP2P systems", The Computer Journal, Vol. 54, no. 4, 19 pages, 2011.

A Bitmap-Centric Environmental Model for Mobile Navigation Inside Buildings

Martin Werner and Moritz Kessel
Mobile and Distributed Systems Group
Ludwig-Maximilians-University Munich, Germany
Email: martin.werner@ifi.lmu.de, moritz.kessel@ifi.lmu.de

***Abstract*—Most indoor navigation applications need a realistic description of ways that is easy to understand, remember, and follow. In this paper, we propose a bitmap-centric model that enables the on-demand computation of such ways on smartphones. We explain the creation of the model with the help of common bitmap floorplans by identifying rooms, doors, and their topological relationship, and we describe the generation of navigation graphs from our model. Those navigations graphs are efficiently calculated, as they are formulated as sequences of bitmap operations. Smartphones are well-optmized for such operations, as they are needed for the graphical user interface, too. The final results are realistic and easy to follow, which is achieved with the help of an algorithm for the identification of relevant landmarks. We also show the results of a performance analysis of the algorithms implemented on a standard smartphone, demonstrating the feasibility of our approach.**

***Keywords*-Indoor Navigation; Environmental Models; Image Processing; Indoor Positioning;**

I. Introduction

In the last years, many new multimedia services have been designed. Furthermore, the ongoing trend towards mobile computing and the immense development in the field of cellular phones lead to more and more context-information that can actually be used in numerous applications from the field of location based services. While it is comparably easy to provide location based services for the outdoor area, it is much more difficult to do the same for the indoor area. In the field of indoor positioning and navigation, there are many problems, which have been solved for the outside. The first problem is the availability of digital map data. While for most outdoor location based services the map functionality offered by major Internet companies (e.g., Google Maps) is enough, there is not yet a comparable service for the indoor area. There are several reasons behind: The complexity of indoor maps would be much higher than outdoors (different floors, differences in the treatment of free space). Moreover the content of indoor maps is often protected by intellectual property rights of architects. To enable more users to generate map data for indoor navigation purposes and to enable crowdsourcing approaches to the lack of environmental information, some algorithms were proposed in [1], which enable simple bitmap floorplans to be used for navigation. In this publication, these concepts are extended to a sufficiently detailed environmental model for indoor navigation applications.

Indoor navigation is a very promising technology. People want to have technological support for indoor orientation, which is comparable to the outdoor situation, especially in very complex buildings (industrial buildings), buildings not known to the majority of guests (foreign airports, exhibitions), or buildings not known exactly enough for safety services (firefighter, police, etc.). The most difficult task for indoor navigation is of course the positioning of the users. But in the last decade many promising technologies are under development, which will solve this problem completely in the near future. These include classical signal-strength methods based on Wireless LAN [2]–[8] or more specialised approaches based on UWB [9]–[11], as well as advanced statistical treatments of this measurement data [12]–[15]. Beside these propagation-based systems there have been proposals, which use GPS pseudolites and similar dedicated infrastructure [16]–[18], the variation of the magnetic field [19]–[21], or the variation of the acoustic background spectrum [22].

Furthermore, cellular phones are becoming more and more powerful in terms of calculational power as well as in terms of sensing capabilities. A complete integration of all data that a modern cellular phone can sense will lead to indoor positioning with acceptable accuracy in the future.

Often, indoor navigation has been implemented in a relatively small area due to the focus on positioning technology. In such small settings it is not really important to have efficient algorithms for several tasks. However, large-scale indoor navigation is an upcoming topic [23], [24].

With this paper, we want to introduce a bitmap-centric environmental model for use within indoor navigation applications. Using bitmaps, we follow a minimal modelling effort approach, as bitmaps can be generated from CAD-data or be scanned from a blueprint or even generated by hand by non-specialists using an image manipulation program. The drawback of such an approach is that bitmaps basically do not provide a method to store semantic models, such as the interconnection of rooms. However, this is easy to overcome, if we define a symbolism. In this paper, we provide a bitmap symbolism very near to the typical symbolism of floorplans (e.g., any floorplan can easily be completed and checked by non-specialists using a drawing program).

This symbolism allows us to find rooms and doors,

generate an interconnection graph, calculate shortest routes based on this navigation graph and, by removing the door symbols and calculating a shortest path on a per-pixel basis, we get a candidate for a shortest path inside the floorplan. Unfortunately, such a path will keep near walls. To remedy this effect, we provide two algorithms, which have already been presented in [1]. The first one allows for the transformation of a shortest way into an augmented form, which is easy to understand, remember, and follow.

The second algorithm allows to enhance the visualisability of a given way. Navigation in general is often based on a waypoint graph [25], [26]. A waypoint graph consists of a set of waypoints and an edge inbetween those waypoints, which are directly reachable from each other (usually on a straight line of walking). As the impact of the graph size on the performance of shortest path algorithms is high, people have studied algorithms generating small graphs. However the reduction of the number of waypoints in a graph leads to fewer ways inside the graph. Hence, the shortest way inside the graph is not a way that a human being would choose. One of the smallest and most efficient graphs is for example the corner graph. The corner graph is a graph containing all corners of the building (map) as vertices and an edge between two corners if and only if the direct line inbetween these two points is inside the building and free (walkable) space. The resulting ways tend to scrape along walls as can be seen in Figure 1(a). Another well-behaved form of a navigation graph consists of a subset of a grid of waypoints in navigation space defined by some properties (such as a minimal distance to a wall and being inside the building) and edges between adjacent waypoints if there is free space inbetween. A shortest way inside such a graph can contain some flaws as can be seen in Figure 1(b). In this example the relatively small grid size leads to the effect that the chosen way is too near to the upper wall. This type of problem is exactly what we want to remedy with the second algorithm in this paper.

When it comes to indoor navigation it is essential to fix a model of the surroundings. It is important to define exactly, in terms of available data, what is meant when talking for instance about a room or a door. In the following chapter, we give possible definitions for this in terms of images. Of course there are other views (especially concerning GIS databases), which can be more flexible, but our choice is made on the background that good GIS data does seldom exist for the indoor area, while a basic floorplan (possibly scanned from a building blueprint) is almost always available. Furthermore, cellular phones have more difficulties with handling vector GIS-data and the associated queries than with handling and manipulating bitmaps. Moreover, the standardisation of bitmap file formats allows for flexible and sustainable treatment of indoor navigation data.

In the following Section, we describe an environmental model in terms of bitmap floorplans. In Section III we explain, how to extract the needed room and navigation information out of said floorplans, in Section IV we explain the problem of finding relevant landmarks and describe an algorithm solving this problem. In Section V we describe an algorithm to enhance the visualizability of ways, which also clarifies turning points useful for textual description of the way. We then explain an implementation of this algorithm for mobile phones and give experimental results on the performance. Section VII finalizes this paper with a outlook.

II. The Indoor Environmental Model and its Association with Bitmaps

In the following paragraphs, we want to describe exactly our environmental model and how it is setup with different bitmaps. Starting with a building (or site) in some reference system, we define the bitmap projection by first projecting the building information into an orthogonal coordinate system and map the bounding box of the building to a bitmap by a choice of pixel size in terms of the orthogonal coordinate system. This pixel size can be used to mediate between a small bitmap using large pixel sizes and fine-grained geometry representation using small pixels.

For this bitmap floorplan we start with some definitions, which do not exactly resemble the definitions of common agreement.

Definition 1: An **area** is a subset of available pixels.

Note that an area need not be connected or otherwise have properties, which the term area describes in other contexts.

Definition 2: The **4-neighbour-floodfill** operation at (x, y) is defined to fill the pixel at (x, y), if it is of the same colour as the initial pixel and continues with the neighbouring pixels (above, left, right and below) of the same color.

Definition 3: The **floodfill-closure** of a point (x, y) is the area inside the bitmap that would be filled by a 4-neighbour-floodfill operation at (x, y) with a colour that is not used elsewhere in the bitmap.

Definition 4: An area is called **floodfill-connected**, if and only if it is the floodfill-closure of one (and hence any) of its points.

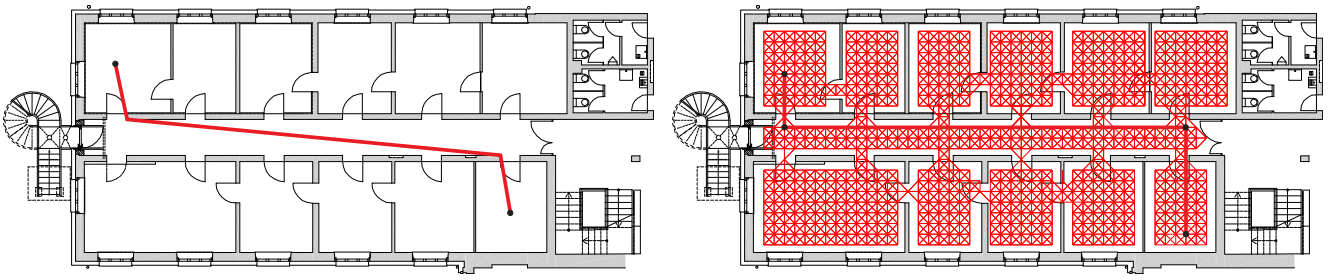
Definition 5: A **room** is a floodfill-connected walkable (non-black) area.

Definition 6: The **outside-space** is the floodfill-closure of the pixel coordinate (0,0).

Definition 7: A room R_1 is **inside** another room R_2 if and only if R_1 is in the convex closure of R_2.

Of course, this definition does not recover the usual term of a room being inside another room. But as it is unlikely that a room that is contained in another room in our sense is not reachable easily this discrepancy to the real world is not severe.

With these definitions in place, we formulate some properties that a floorplan should have for the application of our algorithms below:



(a) A shortest way inside a corner graph

(b) A shortest way inside an accessibility graph

Figure 1. Examples of waypoint graphs and problems

- *Closed-Building-Property*: The building is closed by black lines and completely surrounded by white space. This essentially means that the outside space resembles the usual definition of what is outside a building.
- *Doors-Are-Rooms-Property*: A door inside the building is drawn such that it itself is a room in the sense of Definition 5. An example of a legal door and an illegal door is given in Figure 2.
- *Walkable-Space-Is-Known*: There is a means of telling whether a straight line between two points lies in walkable space (which is defined to be the space where a human being can walk).

For the rest of the paper, we assume that we are given a bitmap - simply called floorplan in the sequel - where these properties hold. Especially, we assume, that the walkable space is provided as a marker color (say white) in the map. Then the property "Walkable-Space-Is-Known" is given by a simple line painting algorithm, which checks that all pixels in the line are of said marker color.

A. Map Symbol Recognition in Mobile Navigation Systems

The environmental model described so far is really minimalistic. A typical environmental model is a representation of data such that a given set of operations is possible. For the GIS-community working with large server infrastructures and desktop computing, this is interpreted in the sense that the set of operations should be as big as possible, but with minimal cost for the individual operation. The size of the model and the complexity of generating such a model are widely ignored. This can be best subsumed by saying that traditional development of location models for ubiquitous computing is based on the question "How should it be?".

A classical discussion of features a location model should provide is given in [27]. Essentially, the authors discuss some examples and queries and conclude that a location model should provide

- *Object Positions:* Positions of objects have to be modelled (either in form of geometric coordinates or in form of symbolic coordinates).
- *Distance Function:* Positions should be interrelated by a well-defined distance function.

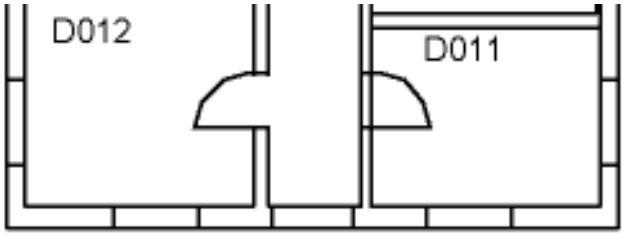


Figure 2. Examples of different doors. The left door is not legal with respect to our properties while the right door is legal

- *Topological Relations:* Spatial containment and spatial connection should be modelled.

Furthermore, they propose to regard these requirements in conjunction with the requirement of minimal modeling effort. The paper then explains some types of basic location models (e.g., Set-based models, Hierarchical models, Graph-based models, etc.) and how they can be structured to fulfill the requirements above.

Our basic "Indoor Environmental Model" does not fulfill the requirements above directly, but we will show how easy it is to come up with complete support for these queries by using some combination of algorithms, and that it is possible to embed all additional information into the floorplan images EXIF-tags.

1) Object Positions and Orientation: Object positions are easily modeled with our approach. A global coordinate consists of a unique identifier of the file (possibly an URL) together with a pixel coordinate (x, y) inside the image. It is of course possible to fix another scale (e.g., one unit on the axis is one meter in the real world) at the expense of saving the scale somewhere or defining it globally for the complete system.

2) Distance functions: As a distance function, we can use the Euclidian distance between pixel coordinates. This basic distance function can be used to calculate the distance between two individual places (e.g., two pixel coordinates (x_1, y_1) and (x_2, y_2)) or even as the distance between two rooms (compare Definition 5).

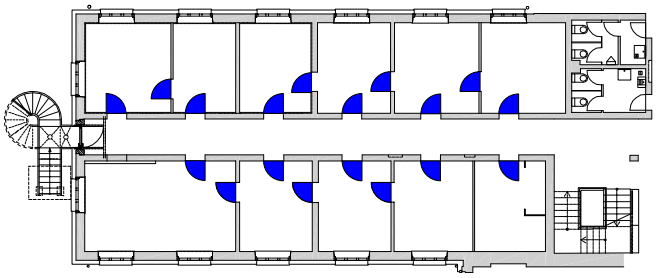


Figure 3. The training image (excerpt)

3) Topological Relations: The relation of a room being contained in another room is immediately available by using the convex closure (see Definition 7). This can be easily extended to any area, including the union of some rooms, a room and its neighbours et cetera. The topological relation of being connected to something is not modelled explicitly, but in the next Section, it will be explained how a navigation graph with one vertex per room from the given floorplan is constructed.

III. Automated Construction of a Navigation Graph from a Floorplan

A navigation graph is a graph consisting of vertices and edges. A vertex has a coordinate and an edge connects two vertices, if a direct walking connection is available inbetween. Such a graph structure can be used to easily and efficiently calculate ways between two points. To keep the graph small enough, we decided to use the rooms as symbolic coordinates. So the navigation graph consists of one vertex per room. If we now discuss the creation of edges, we need two functions based on two given vertices: First of all, we want to detect whether a given vertex constitutes a room and secondly, we need to decide whether two rooms are adjacent to each other. Then the edges in the navigation graphs are exactly between those vertices, whose corresponding rooms are neighboring and where exactly one of them is a door. For an automated construction of a navigation graph, we only need a stable and fast method for detecting doors (e.g., rooms, which have the shape of a door) and a fast method for detecting all neighboring rooms.

For enhanced performance, we will only give the graph in its natural implicit form and generate it on the fly while a graph search algorithm (e.g., Dijkstra or A^*, see [28]) is examining vertices and their neighbours. In this way, we do not have to explode the floorplan into a list of rooms, which could be quite large using our definitions. Think of a lattice drawn somewhere on the floorplan. This will consist of many rooms, which are all irrelevant for the connection graph.

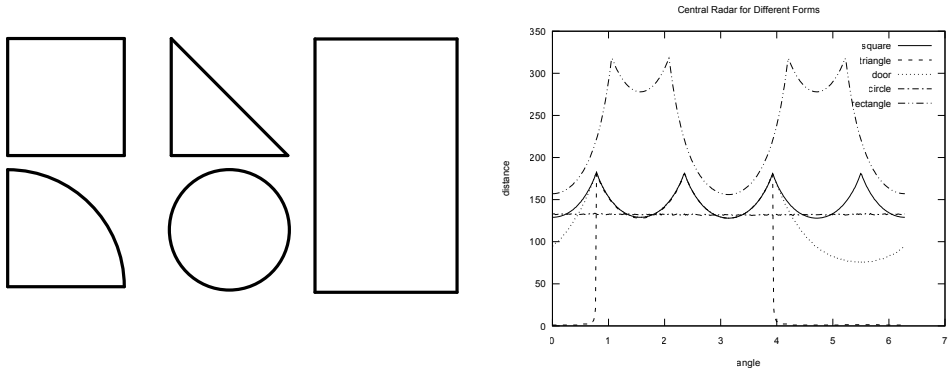
A. Detecting Door Symbols

For the detection of door symbols, we wanted to construct a condensed model with a simple classification system, which is then used to classify a room. Therefore, we set up a list of features, which can be easily extracted from a room (e.g., a floodfill-connected region). These feature vectors are then used together with a manually classified image, where all doors have been marked in blue color for the construction of a training set (see Figure 3). The strength and generalization abilities of the models created from this training data have been tested against rotation transformations. They are not scale-invariant by intention, as we will save the model into the image file itself. Hence, there is no need to recapture scaled versions of an object.

The features are basic features used to distinct between different convex areas, which are easy to compute even on a mobile device. The features are all numeric values describing some aspect of the form. Two very basic features used are given below.

- *Area:* The area is one of the simplest features, but still it is the strongest one. The area of a room is defined as the number of pixels that would be filled by a regular 4-neighbour-floodfill operation. As we want to embed the classification model into the graphics file, it is no problem that this measure is not scale-invariant.
- *Bounding Box:* The width and height of the bounding box of a room used. Here the bounding box is not a minimal enclosing box (which would be more complicated to compute) but is an axis-parallel box containing all pixels that belong to the room. We use the width and height of this box as features.

These simple features did not suffice to classify successfully between doors and other rooms inside a floorplan. The main reason is that the bounding box is not very distinctive for cases, where a door is rotated. It is difficult to differentiate between a square and a door by using a bounding box. Moreover, the bounding box is not rotation invariant. Hence, we constructed a pretty simple, yet efficient algorithm, which calculates a sequence of values based on the floodfill-area, which can be used to derive pretty distinctive features at least for convex forms. The general technique is a variant of a line sweeping mechanism, which we will call *Central Radar*. A ray casting is done in several equally stepped directions (say 1°) recording the radius of the room measured from the middlepoint of the minimal enclosing box. This measure can be zero for nonconvex rooms. Figure 4 shows the central radar of some convex forms and gives a hint, why the central radar is well able to classify between doors and other rooms. For the classification task of detecting doors, we found out that it suffices to use the following simple features derived from a central radar: The *maximum radius*, the *minimum radius* and as a measure for the deviation of the shape from a square their quotient called *squarity*. If the minimal radius is zero and hence the value of squarity is not defined, we declared squarity to be a missing value for the machine learning task.



(a) Some Forms (b) The central radar of these forms

Figure 4. Central radar as a method of obtaining distinctive features for automic recognition of door symbols

Algorithm	Datset 1.)	Dataset 2.)	Dataset 3.)
RIPPER [29]	99.85%	97.47%	99.91 %
Naive Bayes [30]	99.37%	99.15%	99.67 %
C4.5 [31]	99.85%	97.89%	99.90 %

Table I
SUCCESS-RATE OF DIFFERENT MACHINE-LEARNING ALGORITHMS DATASETS 1.) - 3.) DIFFERENTIATING BETWEEN DOORS AND ROOMS AS DETERMINED BY TENFOLD, STRATIFIED CROSS-VALIDATION

Using this feature set, we trained several simple classification algorithms. So we are not using very complex classification algorithms such as neuronal networks as long as simple algorithms provide comparable performance. In essence, all major, simple classification algorithms performed more or less equally well, see Table I. We used three datasets where all rooms where extracted from a floorplan pixel-by-pixel and classified according to a training image, where the doors have been coloured blue:

- *Dataset 1.):* A wing of the building as depicted in Figure 4 (238 rooms including 36 doors).
- *Dataset 2.):* The same wing of the building rotated by 20° (2447 rooms including 36 doors).
  Note that by rotating and the associated blur a lot of small-area "rooms" with non-black pixels appeared, which could have been ruled out by rejecting area below a threshold. But we wanted to see the influence of such typical image noise, which is fortunately void. Removing all rooms below an area of 30 pixels would leave only 198 rooms.
- *Dataset 3.):* A complete floor of the same building. (1755 rooms including 263 doors)
  Note that the building is not axis-parallel, but has some rotated wings as you can see in Figure 5.

As there is no significant performance difference in the classification algorithms, we can choose the right classification type by taking into account, which secondary properties the model does have. For example, it is easy to use RIPPER [29] to induce a set of rules and store this set of rules in an ASCII-string. It is user-readable, pretty easy to understand, and has good performance. Moreover, the rulesets generated with RIPPER were very small. For the complete map, RIPPPER used only three rules containing seven inequalities.Using these rules, only ten doors were overlooked. It could be easy to complete this model by storing ten coordinate pairs as a correction to the model giving another 40 bytes overhead for the classification model. As we want to bind the model to the image file itself, it is also possible to use unpruned and highly overfitted models. Though the models loose their generalization capabilities, the success rate gets almost perfect. A RIPPER model containing ten rules was able to reach a performance of 99.85% with only three misclassifications and uses only 230 bytes in a human-readable text format.

B. Detecting Neighboring Rooms

The only thing left, before a complete navigationg graph can recursively be constructed, is given by the need for a method to find the neighbours of a room. We decided to implement this also as a variant of the central radar algorithm such that the features for detecting doors can be calculated in one step together with the neighbours of a given room. Since it is sufficient to calculate only the neighbors of doors in order to derive a complete navigation graph, problems with non-convex rooms can be avoided. For calculation of neighbours, we use a marker color and find all neighbours floodfilling them to the colour. Then we can enumerate all neighbour rooms by finding the for example first topleft pixel having the marker color, storing this as a room-defining pixel, and floodfilling with white to remove the marker colour from the connected region.

For the colouring of the neighbours, we first fill the room with a marker color (say yellow). Then we use a rotating line around the center point of this room (e.g., the middlepoint of the bounding box), and find the first pixel along each line, which is not black or the marker color of the base

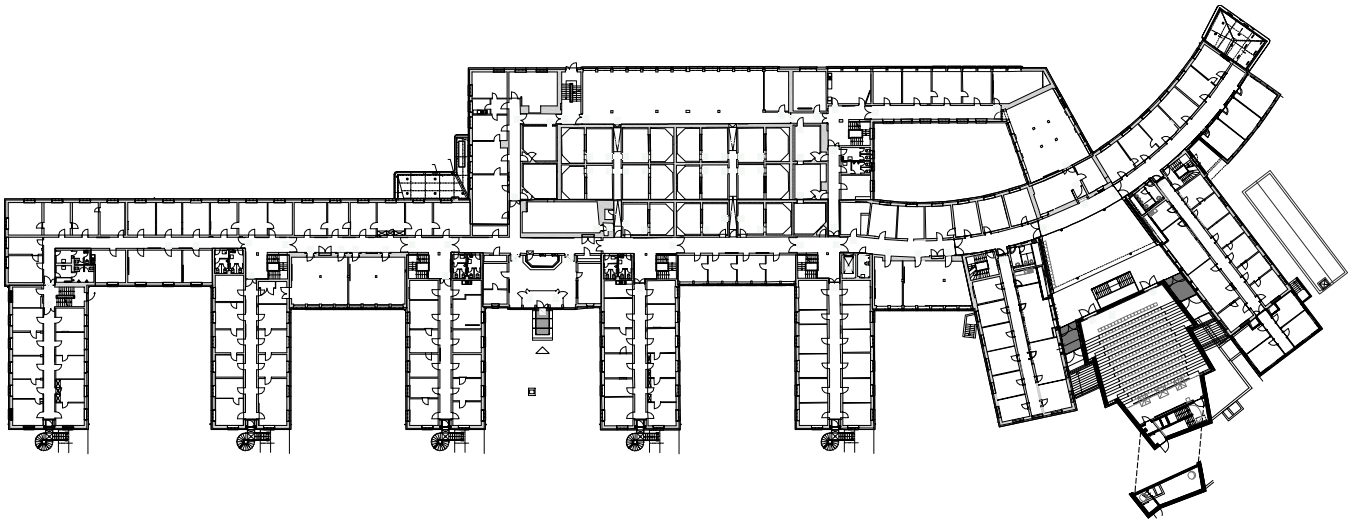


Figure 5. The complete building floorplan

room (yellow). We then floodfill at this pixel coordinate with the marker color for the neighbours (say magenta). Having done this, the base room is yellow and the neighbours are coloured magenta. Note that this algorithm does only work for convex rooms and that the neighbourship relation can still go through walls. But as we stated earlier, our navigation graph will only contain an edge between two neighbours if one of these neighbours is a door, which we can quickly decide using the algorithm of Section III-A.

C. Generating a Navigation Graph

Combining both algorithms from the previous two Sections, it is easy to calculate a navigation graph for the given floorplan as described in the following pseudo code:

```
void calculateGraph(int x, int y){
  if isReady(x,y) return
  for each (nx,ny) in Neighbours(x,y){
    if (isDoor(nx,ny) || isDoor(x,y)){
      AddVertex(nx,ny)
      AddEdge((x,y),(nx,ny))
      calculateGraph(nx,ny)
    }
  }
  isReady(x,y) = true
}
```

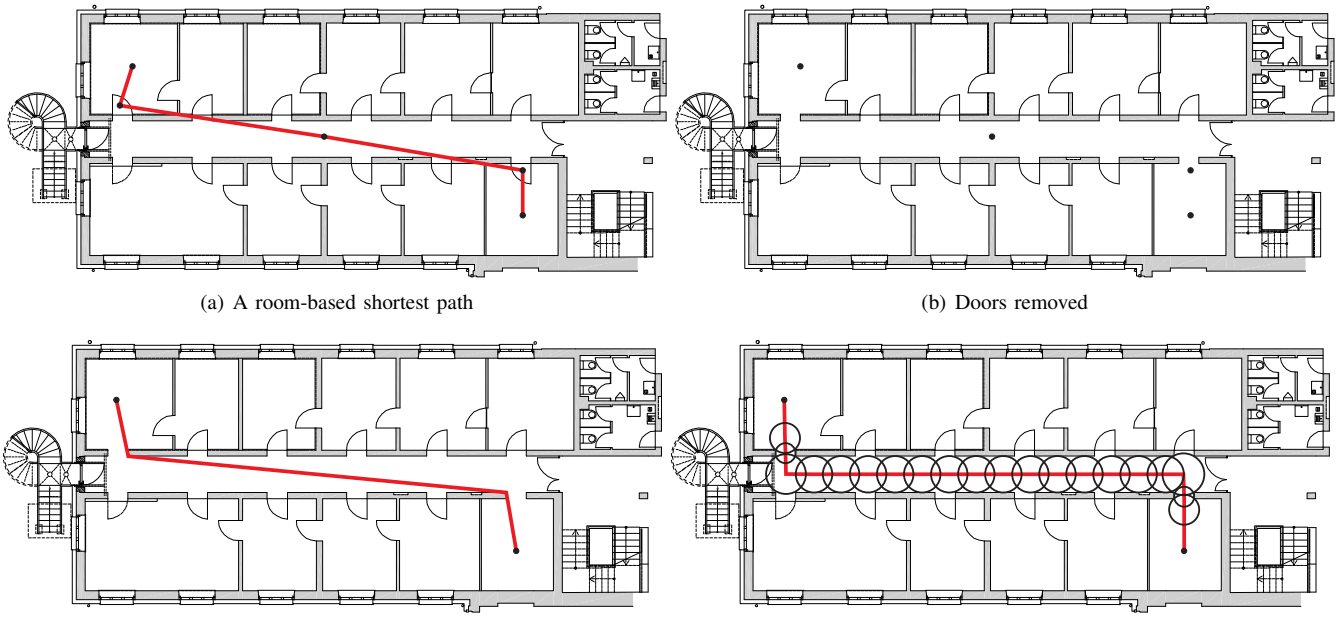
Starting with a specific pixel coordinate (x, y) given by the users position possibly estimated with a positioning system, we first check whether the room specified by (x, y) has already been done. The information of finished rooms can either be coded into the image using another marker color or be stored in a global table. If the room has not been done, we enumerate all neighboring rooms using the algorithm of Section III-B. We then add a vertex for each neighbour and an edge connecting this neighbour to the room specified by the current pixel coordinate (x, y) if one of both rooms is a door. For this task, we use the algorithm given in III-A. We can integrate this process with a graph search algorithm such as Dijkstra or A^*. Then the loop over all neighbours is invoked each time a specific room vertex is popped from the central priority queue. For A^* it is then very likely that not the complete map will be explored, but only a small portion of the given graph for navigation. The shortest path can then be readily given on a per-room-basis but there is no reason to believe that the edges lie inside the rooms.

D. Removing doors

Once we have constructed the navigation graph, we can calculate a shortest route on a per room basis. Once we have calculated such a shortest path inside the graph, we have a sequence of rooms, which leads from the given start position to the end position on a short route. The problem is now, that the shortest path might not lie inside the walkable space, as the graph consists of some points inside the rooms and doors. Even choosing the middlepoints as depicted in Figure 6 as room reference points does not solve this problem. In this situation, we do the following as a preparation for the visualisation algorithm presented in Section V below:

For each door in the shortest path, we calculate a minimum enclosing box, enlarge it a bit (such that the bounding lines of a door get inside the enclosing box) and fill it with white color. Thus, we retrieve one single walkable floodfill-connected region, which connects the startpoint and the endpoint.

Using this representation, we can calculate again a shortest route between the starting point and the endpoint on the basis of a implicit navigation graph consisting of white



(a) A room-based shortest path

(b) Doors removed

(c) The implicit graph structure

(d) The result of the PPW algorithm

Figure 6. Preparing a room-based shortest path for visualisation

pixels as vertices and up to eight edges connecting each pixel with all white neighbour pixels. A weighting for this graph is given by a constant distance of 1 for horizontal and vertical edges and a distance of $\sqrt{2}$ for diagonal edges. Figure 6(c) illustrates this step.

IV. Landmark Search

Pedestrian navigation results need a different presentation form as compared to vehicle navigation results. While for a vehicle navigation system based on GPS the typical errors of the positioning system do not have much influence on the identification of a turn and the orientation of the vehicle is known by the orientation of the street, this is not true for the indoor area. We usually have positioning systems with low accuracy and different possible turns within this accuracy. Typically the positioning errors do not allow the distinction of two doors, which are directly next to each other. In this situation, we want to augment a navigation solution with semantical information such that it is easy to remember and follow. For this case the concept of a landmark is often used. Landmarks are objects in the surroundings having a local uniqueness and being eye-catching. All classical signs are landmarks in this sense. But even shops and plants can serve as landmarks. With landmark information the problems of coarse positioning can be reduced. If landmarks are drawn into a map in form of a pictogram (e.g., the logos of the shops) the relation between the proposed way and those shops can be easily remembered and used for difficult way decisions. This role of a landmark is best explained by the following textual instruction, which can be generated if good landmark information is available: ”Before the post office turn left. Then you will see a red sculpture in 300m distance.” As you can see from this sentence, landmark information can be used to make explicit the position of a turn in relation to semantical information (as opposed to geometric information). Landmark information can also be used to enhance the confidence in having a good orientation as you can see from the second sentence.

The difficult task is now to reduce the set of landmarks (which is usually very big) to the set of landmarks that are visible from the way to facilitate more complex election algorithms for the actual integration of landmark information into visual and textual representations of the way.

The following algorithm is a good symbiosis of a search technology with a geometric enhancement technology. The results of this algorithms is an image containing a set of visible landmarks identified by a colour convention. It is possible to very quickly extract this information as a list of visible landmarks or to directly integrate the graphical result into the visualisation pipeline. One could for example highlight the visible space, draw a pictogram over visible landmark positions and so on.

A. Landmark Search By Image Processing

With the following algorithm we solve the problem of finding relevant landmarks out of a list of landmarks visible from within a way. Common algorithms to solve this problem are more or less searching for landmarks by checking whether a given landmark is visible. As there is no good geometric ordering of landmarks (i.e., reducing the search space by a distance limit will miss good landmarks in long

rooms) it is not easy to do such a search efficiently. This type of search problem also shows up in other problems of ubiquitous and context-aware computing.

A landmark in the sense of the following algorithm consists of a pixel coordinate and a connected information (identification in a database, etc.). The input of the algorithm consists of

- A set of landmarks
- A floorplan
- A way (given as a list of points forming a line-strip)

The configurable parameters influencing this algorithms are

- The maximal viewing distance
- The length threshold used during tessellation of the way

1) Step 1: Prepare Landmark Map: The very first step is to overlay our floorplan with drawn landmark locations. Therefore we use a colour palette mapping a colour (that is not used in the floorplan) to the identification data. This mapping is symbolised in the following pseudo-codes by the function `landmark_to_colour(landmark l)`.

```
void draw_landmarks(){
  copy (floorplan, landmark_map)
  for each landmark l in landmark_set{
    putpixel(landmark_map, position,
         landmark_to_colour(l))
  }
}
```

This step is of course general and the result can be cached for subsequent applications of this algorithm.

2) Step 2: Tessellation of the Way: As it is computationally very expensive to calculate the set of pixels, which are visible from a line-strip, we approximate this set of pixels by the set of pixels visible from the points of a tessellation of the line-strip. To obtain this tessellation, we keep inserting middlepoints between two subsequent points until the distance between all points is shorter than the *tessellation length*.

```
global tessellation_length
void tessellate(linestrip l){
  for each segment s of l{
    if (s.length() > tessellation_length){
      s.split()
      return tessellate(l)
    }
  }
}
```

3) Step 3: Calculate the mask bitmap: In this step we calculate a mask, which resembles the set of pixels visible from the way. This is done by preparing the mask bitmap to be of the same size as the floorplan and filled with black. We then use a radial floodfill operation starting at each point of the tessellated way and copying every examined pixel into the mask bitmap.

```
void calculate_mask(){
  for p in tessellated_way{
    radial_flood_fill(p);
  }
}
```

The radial floodfill algorithm fills out the area surrounding a point as long as there is a direct line between each point and the starting point. Note that the resulting area need not be convex (see Figure 7(b) for an example).

4) Step 4: Multiply the landmark map with the mask: In this step, we stamp the visible area out of the landmark map. For each black pixel in the mask bitmap, we black out the same pixel in the landmark map.

```
void mask_out(){
  for each (x,y) in landmark_map{
  if (mask_map(x,y) != black)
    result(x,y) = landmark_map(x,y)
  }
}
```

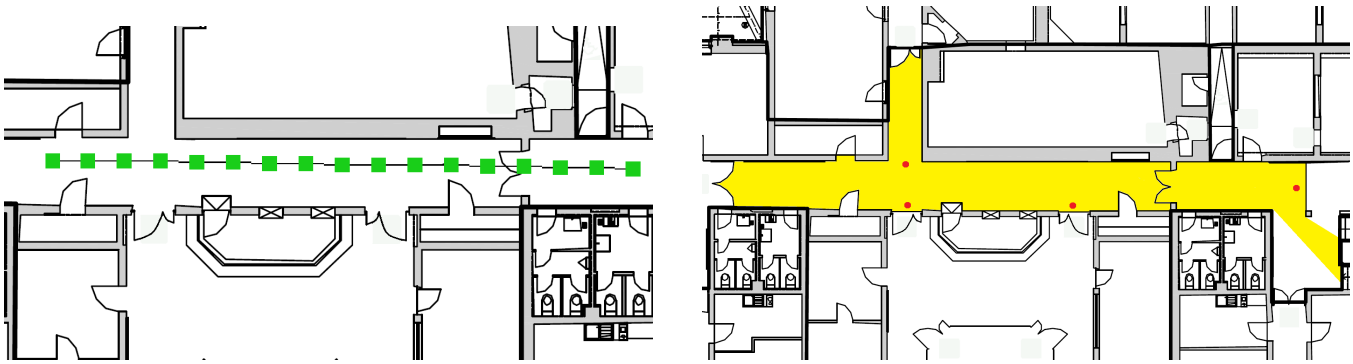
This results in a bitmap containing exactly the visible landmarks together with their geometric location. It is now up to the rest of the visualisation pipeline how to work further. One could quickly scan the image and get a list of visible landmarks or one could just overlay all landmark pixels with a pictogram assigned to the landmark.

Though this algorithm seems to be very complex, it has some beneficial properties. First of all, it is a formulation of the landmark search problem in terms of basic image processing. Secondly, its result is near to a complete visualisation of the landmarks and thirdly, it is highly parallelisable and designed to be run on dedicated graphics hardware. Examples for an application of this algorithm are given in Figure 7.

V. Post-Processing Ways (PPW-Algorithm)

Waypoint graphs in general impose visualisation problems. If the graph is too coarse, then the shortest ways will appear unnatural, e.g. moving to middlepoints of rooms. Even if this is not the case, the best natural-looking ways are usually not the shortest, but have some relation to the surrounding geometry making them easily visible. If the graph is too fine, then shortest ways tend to scrape along walls and cross large halls in diagonals. Hence, for an indoor navigation system it is difficult to find ways, which are short and at the same time have a specific quality with respect to visualisation. As the efficiency of search algorithms is tightly coupled with the number of vertices and edges inside the graph, it is common that people try to have small waypoint graphs.

With the following algorithm, we want to post-process such ways to obtain a relatively nice visualisation with acceptable computational overhead.



(a) The tessellation of a way

(b) The mask and landmarks (large dots) found

Figure 7. The results of the Landmark Search Algorithm

A. The PPW-Algorithm for Better Visualisation

Therefore, we propose the following algorithm, which essentially is a series of image processing operations. The input of the algorithm consists of

- A way (given as a list of points forming a linestrip)
- A collision map

The configurable parameters influencing this algorithms are

- The maximal length a point may move during the algorithm
- The length threshold used during tessellation of the way
- The valuation choosing the best movement in a set of possible movements

1) Step 1: Tessellation of the Way: For this algorithm we need a tessellation of the way just as for the previous algorithm. The reader is referred to Section IV-A2 for details.

2) Step 2: Move the tessellation points: For each point in the tessellation of the way determine a set of points, where we could move this point. Therefore we grow a circle until this circle collides and if this circle collides, we move the middlepoints away from the collision points until we get stuck. Hence we find out the position where - in a limited neighbourhood of each tessellation point - the biggest circle fits into free space and move this tessellation point to this position (see Figure 6(d)).

Number of Points	Running Time
5	0.759s
9	0.985s
17	1.633s
33	2.955s
65	5.623s

Table II
PERFORMANCE OF THE PPW ALGORITHM RUNNING ON HTC DESIRE

The following pseudo code illustrates this step. We found out in experiments, that a maximal movement distance of four times the *tessellation length* makes sense for relatively fine tessellations.

```
global tessellation_length
for each point p {
  res = grow_a_circle(p,
          tessellation_length * 4)
}
```

The method `grow_a_circle` is just scanning possible positions and calculating the maximal circle in free space centered around these positions. Each position is then assigned a valuation composed out of the radius of the circle and the distance of the movement. In the examples throughout this paper, we just used a valuation prefering the biggest circle in the allowed space and inbetween all those circles with maximal radius the one nearest to the original point.

```
void grow_a_circle(point p, double d){
  last_result = p;
  for  q in box (p-(d,d), p+(d,d)) {
    r = maximal_radius(q);
    if (is_better_than_last_result(q,r))
      last_result = p;
  }
}
```

The results of this algorithm are given in Figure 6(d). As you can see, this algorithm leads to a fairly good way. What is not obvious but has been tested with a multitude of other ways, is the fact that the algorithm has the beneficial side effect of having a relatively clear turn (e.g., a turn of merely exactly 90 degree in the Figure above) exactly where a turn should be indicated by a text generation engine. Furthermore, due to using a rotation-invariant definition of good way, the orientation of the rooms inside the bitmap is not relevant.

VI. An Implementation of PPW for Mobile Phones

The algorithms presented in this paper are relatively complex. If we apply these algorithms to long ways, the uniform tessellation algorithm leads to many points in the tessellation and for each of those points another complex operation is needed. To enhance clarity, we decided to explain the algorithms in the most simple form given above. Of course the growing circle algorithm can gain a real performance boost from not growing the radius one pixel at a time. Starting with an exponential growth of the radius and correcting the first collision by a nested interval algorithm in comparison to the last non-colliding circle will gain much speed. Moreover, for plans where the magnitude of rooms is constructed from parallel lines and the number of randomly placed obstructions is small, the circle can be replaced by a square without any harm. Using integral images in this case allows to answer the question, whether a square collides with geometry, in constant time. If we can afford the memory and the map does not change too often, we can even compute the maximal radius for each pixel and store it as a color component value inside another bitmap. In this case (ignoring the time of constructing this map), we can omit the process of growing a circle and concentrate on the movement of the center point.

We implemented the PPW algorithms for modern smartphones running Android OS 2.2 and above. The Post-Processing Ways algorithm (Section V) is running fast enough. The system is rendering a map into a screen buffer (a Java bitmap of the exact pixel size of the screen), which is then passed to our implementation of this algorithm via Java natives. In Java natives, we are performing the image processing as described on a 16-bit-per-pixel bitmap (using essentially 5 bits per color). For this task, we implemented a fast and stable bitmap manipulation library. The tessellation is carried out in Java. Experimental performance results for the Post-Processing Ways algorithm are given in Figure V-A2.

For moderate numbers of tessellation points, the running time of the algorithm is quite acceptable. The algorithm has to be run only once for each navigation result. As the effective screen resolution of a full-screen application (not drawing over the status bar) on the device is 480x725, the number of tessellation points to consider will not exceed 20 points. As mobile devices are able to run this type of algorithms natively, we are able to provide full navigation functionality with navigation graphs, which have relatively bad visualisation properties such as corner graphs.

VII. Outlook

With this paper we have presented an extension of our previous work [1]. We have shown that a bitmap can provide enough environmental information for high-quality location-based services by applying some simple image precessing tasks. Furthermore, a technique for the generation pretty general symbol recognition is provided and the possibility of embedding this information into the bitmaps EXIF tags in ASCII format has been demonstated. We applied this technique to differentiate between doors and rooms using highly overfitted models of how a door is drawn on the plan and were able to automatically generate a suitable navigation graph hierarchy. The first level consists of a navigation graph containing one vertex per room (and hence also a vertex per door, as we expect doors to be rooms). The second level consists of a free-space pixel-based graph, which connects adjacent pixels of the same colour. Having all this in place, we have a perfect starting point for the visualisation and augmentation algorithms presented before in [1].

This work allows us to extend the philosophy that the visualization properties of navigation graphs are not important. Furthermore, we showed that simple bitmaps conform to some conventions are feasible alternatives to the design and use of complex environmental models. For a navigation application on a smartphone, some simple rules and some good algorithms are enough to provide a complete and high-quality indoor navigation application. Finally, our integration of non-trivial functionality into the EXIF-tags of the floorplans makes it easy to implement our environmental model inside a mobile browser. Those browsers not supporting our new technology will still be able to show the map as a basic navigation aid.

References

[1] M. Werner, “Efficiently using bitmap floorplans for indoor navigation on mobile phones,” in *Proceedings of the Seventh International Conference on Wireless and Mobile Communications (ICWMC 2011)*, 2011, pp. 225–230.

[2] Y. Chen and H. Kobayashi, “Signal strength based indoor geolocation,” in *Proceedings of IEEE International Conference on Communications (ICC 2002)*, 2002, pp. 436–439.

[3] F. Evennou and F. Marx, “Advanced integration of wifi and inertial navigation systems for indoor mobile positioning,” *EURASIP Journal of Applied Signal Processing*, 2006.

[4] P. Bahl and V. N. Padmanabhan, “Radar: an in-building rf-based user location and tracking system,” in *Proceedings of the 19th IEEE Conference on Computer Communications (INFOCOM 2000)*, 2000, pp. 775–784.

[5] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, “Compass: A probabilistic indoor positioning system based on 802.11 and digital compasses,” in *Proceedings of the First International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH 2006)*, 2006, pp. 34–40.

[6] M. L. Zehner, K. Bannicke, and R. Bill, “Positionierungsansätze mittels WLAN-Ausbreitungsmodellen,” 2005.

[7] M. Kessel and M. Werner, "Smartpos: Accurate and precise indoor positioning on mobile phones," in *Proceedings of the First International Conference on Mobile Services, Resources, and Users (MOBILITY 2011)*, 2011, pp. 158–163.

[8] W. Xiao, W. Ni, and Y. Toh, "Integrated wi-fi fingerprinting and inertial sensing for indoor positioning," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN 2011)*, 2011.

[9] C. Falsi, D. Dardari, L. Mucchi, and M. Z. Win, "Time of arrival estimation for uwb localizers in realistic environments," *EURASIP Journal of Applied Signal Processing*, 2006.

[10] B. Waldmann, *Design of a Pulsed Frequency Modulated Ultra-Wideband System for High Precision Local Positioning*. Logos Verlag Berlin, 2011.

[11] P. Steggles and S. Gschwind, "The ubisense smart space platform," in *Adjunct Proceedings of the Third International Conference on Pervasive Computing*, 2005, pp. 73–76.

[12] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174 – 188, 2002.

[13] P. Davidson, J. Collin, and J. Takala, "Application of particle filters for indoor positioning using floor plans," in *Ubiquitous Positioning, Indoor Navigation, and Location Based Service*, 2010.

[14] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle filters for positioning, navigation and tracking," in *IEEE Transactions on Signal Processing*, vol. 50, no. 2, 2002, pp. 425 – 437.

[15] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House Publishers, 2004.

[16] H. S. Cobb, "Gps pseudolites: theory, design, and applications," *Ph. D. Thesis, Stanford University*, 1997.

[17] C. Kee, D. Yun, H. Jun, B. Parkinson, S. Pullen, and T. Lagenstein, "Centimeter-accuracy indoor navigation using GPS-like pseudolites," in *GPSWorld*, 2001.

[18] C. Rizos, G. Roberts, J. Barnes, and G. N., "Locata: A new high accuracy indoor positioning system," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN 2010)*, 2010.

[19] Ascension Technologies, "Motionstar magnetic tracker," 2001, http://www.ascension-tech.com/.

[20] W. Storms, J. Shockley, and J. Raquet, "Magnetic field navigation in an indoor environment," in *Ubiquitous Positioning, Indoor Navigation, and Location Based Service*, 2010.

[21] F. Raab, E. B. Blood, T. O. Steiner, and H. R. Jones, "Magnetic position and orientation tracking system," in *IEEE Transactions on Aerospace and Electronic Systems*, 1979, pp. 709–718.

[22] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik, "Indoor localization without infrastructure using the acoustic background spectrum," in *9th International Conference on Mobile Systems, Applications, and Services (MobiSys 2011)*, 2011, pp. 155–168.

[23] P. Ruppel, F. Gschwandtner, C. K. Schindhelm, and C. Linnhoff-Popien, "Indoor navigation on distributed stationary display systems," in *Proceedings of the 33rd International Conference Computer Software and Applications Conference (COMPSAC 2009)*, vol. 1, 2009, pp. 37–44.

[24] W. Chung, G. Kim, M. Kim, and C. Lee, "Integrated navigation system for indoor service robots in large-scale environments," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2004)*, 2004, pp. 5099–5104.

[25] P. Tozour, "Search space representations," *AI Game Programming Wisdom 2*, vol. 2, pp. 85–102, 2004.

[26] K. Yu, "Finding a natural-looking path by using generalized visibility graphs," in *PRICAI 2006: Trends in Artificial Intelligence*, 2006, pp. 170–179.

[27] C. Becker and F. Dürr, "On location models for ubiquitous computing," *Personal and Ubiquitous Computing*, vol. 9, no. 1, pp. 20–31, 2005.

[28] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*. Springer-Verlag Berlin Heidelberg, 2008.

[29] W. W. Cohen, "Fast effective rule induction," in *Proceedings of the Twelfth International Conference on Machine Learning*, 1995, pp. 115–123.

[30] G. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, 1995, pp. 338–345.

[31] R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, 1993.

Wireless Networks with Retrials and Heterogeneous Servers : Comparing Random Server and Fastest Free Server Disciplines

Nawel Gharbi
Computer Science Department
University of Sciences and Technology, USTHB
Algiers, Algeria
Email: ngharbi@wissal.dz

Leila Charabi
Computer Science Department
University of Sciences and Technology, USTHB
Algiers, Algeria
Email: leila.charabi@gmail.com

***Abstract*—This paper proposes an algorithmic approach based on Generalized Stochastic Petri Nets, for modeling and analyzing finite-source wireless networks with retrial phenomenon and two servers classes. The particularity of this approach is the direct computing of the infinitesimal generator of the proposed Generalized Stochastic Petri Net without generating neither the reachability graph nor the underlying Markov chain. Furthermore, we assume in this model that servers of one class are faster than those of the second one. In *Random Server* policy, customers requests are assigned randomly to free servers of both classes. The disadvantage of this policy is the increased response time when fast servers are free and requests are assigned (randomly) to slow ones. Hence, this paper aims at presenting another service policy, where priority is given to faster free servers. This policy is called the *Fastest Free Server* policy. Moreover, we compare through numerical examples, Random Service policy to Fastest Free Server one, by developing formulas of the main stationary performance indices of the network. We compare also these two policies to *Averaged Random* case, where the same global number of servers is assumed, but all homogeneous with the average service rate. We show that Fastest Free Server discipline gives better results than both Averaged Random case and Random Server discipline.**

***Keywords*-Wireless networks; Retrial phenomenon; Heterogeneous servers; Performance indices; Service disciplines.**

I. INTRODUCTION

Models with retrial phenomenon are characterized by the feature that a customer finding all servers busy or unavailable, is obliged to leave the service area, but he repeats his request after some random period of time. As we have seen in [1], these models play an important role in cellular mobile networks [5], [13], [14] and wireless sensor networks [15]. Significant references reveal the non-negligible impact of repeated calls, which arise due to a blocking in a system with limited capacity resources or are due to impatience of customers. For a systematic account of the fundamental methods and results on this topic, we refer the readers to [3], [4], [9].

Most studies on retrial models with finite source, assume that the service station consists of homogeneous (identical) servers. However, retrial models with heterogeneous servers arise in various practical areas as telecommunications and cellular mobile networks. In fact, heterogeneous models are far more difficult for mathematical analysis than models with homogeneous servers, and explicit results are available only in few special cases and almost all studies are investigated only by means of queueing theory. In fact, we have found in the literature, only the few papers of Efrosinin and Sztrik [7], [8], [11], [12], where heterogeneous servers case was considered using retrial queueing model, and the paper [10] where we have proposed the modeling and the analysis of multiclass retrial systems by means of colored generalized stochastic Petri nets.

From a modeling point of view, and compared to retrial queueing models, Generalized Stochastic Petri Nets (GSPNs) [2], [6] are a high-level graphical formalism, which allows an easier description of the behavior of complex retrial networks, and it has shown to be a very effective mathematical model. Moreover, from the GSPN model, a Continuous Time Markov Chain (CTMC) can be automatically derived for the performance analysis. However, generating the Markov chain from the GSPN and solving it, still require large storage space and long execution time, since the state space increases as a function of the customers source size and servers number. So, for real retrial networks, the corresponding models have a huge state space.

Hence, using the GSPN model as a support, we have proposed in [1] an algorithmic approach for analyzing performance of finite-source retrial networks with two servers classes, servers of one class are supposed to be faster than those of the second one. In fact, the proposed approach allows to compute directly the infinitesimal generator without generating the reachability graph nor the underlying Markov chain. In addition, we developed the formulas of the main stationary performance indices, as a function of the number of servers of each class, the size of the customers source, the stationary probabilities and independently of the reachability set markings. Nevertheless, the unique service policy we have employed in [1] was the *Random Service* policy, where the server to which a request is assigned is chosen randomly among all idle servers, in both classes. The inconvenience

of this discipline is that it can assign a customer's request to a slow server, while there is at least one fast server free.

In the present paper, we extend our idea by considering another service policy, that can improve performance by giving priority to fastest servers class, i.e., new requests are assigned to a server in the slowest class, only if all servers in the fastest class are busy, this policy is called *Fastest Free Server* policy. Moreover, using some numerical examples, we make a comparison between these two policies, with the *Averaged Random case*, where we suppose the same number of servers (in both classes), but all homogeneous with the average service rate.

This paper is organized as follows. In Section II, we describe the basic model of finite-source retrial networks with heterogeneous servers. In Section III, the basic notions of GSPNs are reviewed. Section IV presents the GSPN models describing retrial networks with heterogeneous servers for each policy namely, Random Server and Fastest Free Server. Next, the proposed stochastic analysis approach is detailed in Section V. The computational formulas for evaluating exact performance indices are derived in Section VI. Next, based on numerical examples, we validate the proposed approach, we discuss the effect of some network parameters on the main performance indices, as the mean response time and the blocking probability, and we compare the two service disciplines, in Section VII, finally, we provide a conclusion.

II. The basic model

We consider retrial networks with finite source (population) of customers of size L and a service station that consists of heterogeneous servers. Each customer can be in one of the following states: free, under service or in orbit at any time. The input stream of primary calls is the so called quasi-random input. The probability that any particular customer generates a primary request for service in any interval $(t, t+dt)$ is $\lambda dt + o(dt)$ as $dt \to 0$ if the customer is free at time t, and zero if the customer is being served or in orbit at time t.

The servers are partitioned in two classes: Class C_1 and Class C_2,where the servers of a given class have the same parameters. Each class C_j $(1 \leq j \leq 2)$ contains S_j identical and parallel servers. There are two possible states for a server: idle or busy (on service). If there is an idle server at the moment a customer request arrives, the service starts immediately. The customer becomes "*under service*" and the server becomes "*busy*". Service times are independent identically-distributed random variables, whose distribution is exponential with parameter μ_1 if a server of class C_1 is selected and μ_2 for servers of class C_2.

Each customer request must be served by one and only one server. Hence, we consider two service disciplines:

- the *Random Server discipline*, which means that, the server to which a request is assigned is chosen randomly among all idle servers, whatever their class.
- and the *Fastest Free Server discipline*, in which the request is affected randomly to an idle server of C_1 class (supposed to be the fastest), if at least one server is free, otherwise, it is assigned to a C_2 class server.

After service completion, the customer becomes free, so it can generate new primary calls, and the server becomes idle again. Otherwise, if all servers of the two classes are busy at the arrival of a request, the customer joins the orbit and starts generating a flow of repeated calls exponentially distributed with rate ν, until he finds one free server. We assume that all customers are persistent in the sense that they keep making retrials until they receive their requested service and that the total servers number $S_1 + S_2$ is smaller than the size of customers source L. Otherwise, the problem is not interesting (no customer in orbit at all).

As usual, we assume that the arrival, service and inter-retrial times are mutually independent of each other.

III. An overview of Generalized Stochastic Petri nets

A GSPN [2], [6] is a directed graph that consists of two kinds of nodes, called places and transitions that are partitioned into two different classes: timed and immediate transitions. Timed transitions describe the execution of time consuming activities and fire with an exponentially distributed delay. Immediate transitions, which fire in zero time once they are enabled, model logic activities, like synchronization, and they have priority over timed transitions.

The system state is described by means of markings. A marking is a mapping from P to $\mathbb{N}$, which gives the number of tokens in each place after each transition firing. A transition is said to be enabled in a given marking, if and only if each of its normal input places contains at least as many tokens as the multiplicity of the connecting arc, and each of its inhibitor input places contains fewer tokens than the multiplicity of the corresponding inhibitor arc.

The set of all markings reachable from initial marking M_0 is called the *reachability set*. The *reachability graph* is the associated graph obtained by representing each marking by a vertex and placing a directed edge from vertex M_i to vertex M_j, if marking M_j can be obtained by the firing of some transition enabled in marking M_i.

Markings enabling no immediate transitions are called *tangible markings*. In this case, one of the enabled timed transitions can fire next. Markings in which at least one immediate transition is enabled, are called *vanishing markings* and are passed through in zero time. In this case, only the enabled immediate transitions are allowed to fire. Since the process spends zero time in the vanishing markings, they do not contribute to the dynamic behavior of the system, so, they are eliminated from the reachability graph by merging them with their successor tangible markings. This elimination of vanishing markings results in a *tangible reachability graph*, which is isomorphic to a continuous time

Markov chain (CTMC) [2]. Hence, the states of the CTMC are the markings in the tangible reachability graph, and the state transition rates are the exponential firing rates of timed transitions in the GSPN.

The solution of this CTMC at steady-state (if the system is ergodic) is the stationary probability vector π, which is the solution of the linear system of equations:

$$\begin{cases} \pi.Q = 0 \\ \sum_i \pi_i = 1 \end{cases}$$

where π denotes the steady-state probability that the process is in state M_i and Q is the infinitesimal generator corresponding to the CTMC. Having the probability vector π, we can easily compute several stationary performance indices of the system, like the mean number of tokens in a place, the mean throughput of a transition, and the probability that an event occurs.

The process of generating stationary performance indices from the GSPN model is summarized in Figure 1.

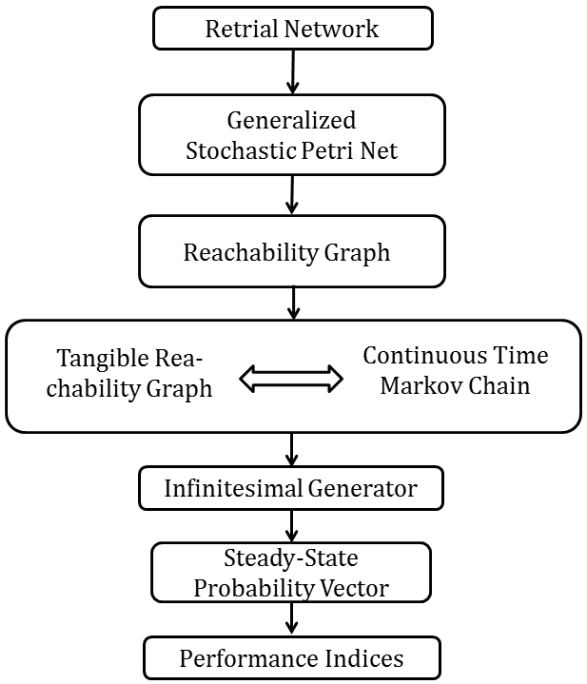


Figure 1. Steps of performance evaluation of retrial networks using GSPN formalism.

However, when modeling real retrial networks with an important customers source size and servers number, generating the GSPN, its reachability graph and then, the tangible reachability graph and the underlying CTMC, require a huge storage space and a very long execution time, since the state space increases as a function of the customers source size and servers number.

IV. GSPN MODEL OF RETRIAL NETWORKS WITH HETEROGENEOUS SERVERS

In the following, we present the GSPN model describing finite-source retrial systems with two servers classes, using Fastest Free Server policy. The detailed model corresponding to Random Server one is given in [1], only the scheme of the corresponding GSPN is given here (Figure 2).

We assume that servers of class C_1 are faster than those of class C_2. The flexibility of GSPN allows us to easily obtain Fastest Free Server policy model, which is depicted in Figure 3.

In this model, place *Cus_Free* represents the free customers, *Orbit* contains the customers waiting for the service, *Ser_Idle1* and *Ser_Idle2* indicate respectively the number of free servers of class C_1 and class C_2, while *Cus_Serv1* and *Cus_Serv2* model the busy servers of both classes.

The arrival of a primary call causes the firing of the transition *Arrival*, which firing rate is marking dependent and equals $\lambda.M(Cus\_Free)$ (*infinite service semantics*), which is represented by the symbol $\#$ placed next to transition, because all free customers are able to generate calls, independently of each other. The place *Choice* is then marked, at this moment, if at least one server in class C_1 is free (The place *Ser_Idle_1* is marked), it will serve the customer's request (firing of transition *Begin_Serv_1*). Otherwise, if place *Ser_Idle_1* is empty and place *Ser_Idle_2* contains at least one token (i.e., at least one C_2 server is idle), the transition *Begin_Serv_2* is enabled, and the request is assigned to a server of class C_2.

In case no server is available at the arrival moment of the primary call (neither in C_1 nor in C_2), the immediate transition *Go_Orbit* is enabled, and a token is put into place *Orbit*, which means that the customer asking for service joins the orbit, it starts generating a flow of repeated calls distributed exponentially with rate ν, as shown in transition *Retrial*.

The firing of the transition *Retrial* corresponds to the generation of a repeated call from a customer in orbit. This transition has infinite servers semantics, since all customers in orbit can trigger repeated calls independently.

By the end of a customer service under a server of class C_1 (C_2 respectively), the timed transition *Serv_End1* (*Serv_End2* respectively) fires. As several servers may be busy at the same time, the semantics of these two transitions is ∞-*servers* to allow modeling parallel services. After completion of service, the customer returns to free state (one token is added to place Cus_Free) and the server becomes

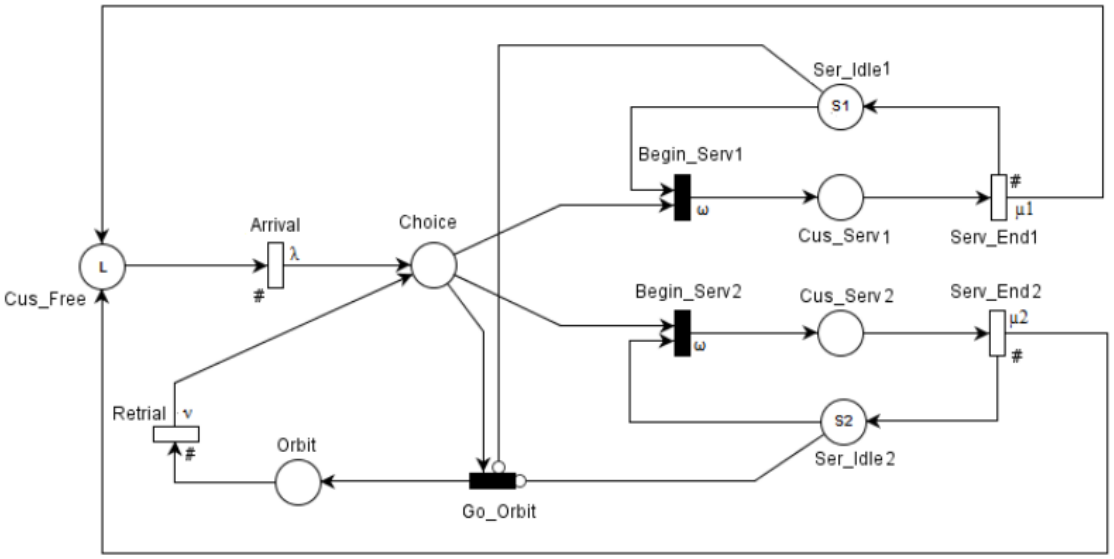


Figure 2. GSPN Model of finite-source retrial networks with two servers classes and Random server discipline.

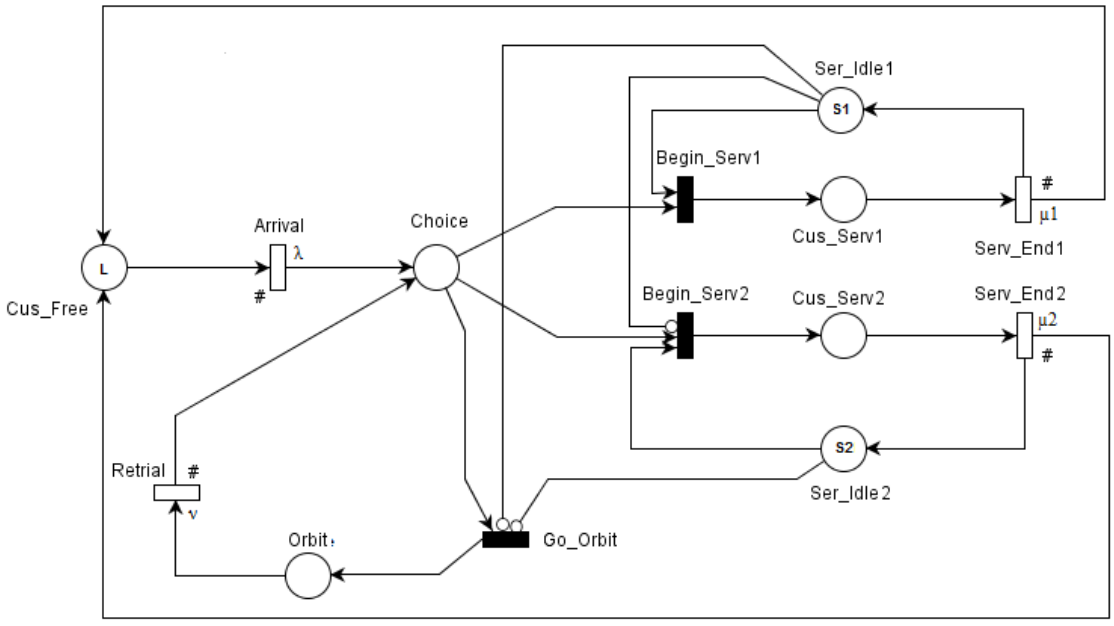


Figure 3. GSPN Model for finite-source retrial networks with two servers classes and Fastest Free Server discipline.

available (one token is put in place *Ser_Idle1* or *Ser_Idle2*, according to the server class).

V. Stochastic analysis

As it is shown in the end of Section III, the disadvantage of calculating performance indices of a retrial network using GSPN formalism was the increase of the state space as a function of customers source size and number of servers when generating the underlying CTMC. In order to overcome this problem, this paper aims to avoid these steps by designing *an algorithm* that computes directly the infinitesimal generator as a function of network parameters, without generating neither the reachability graphs nor the underlying CTMC, as to be shown in Figure 4.

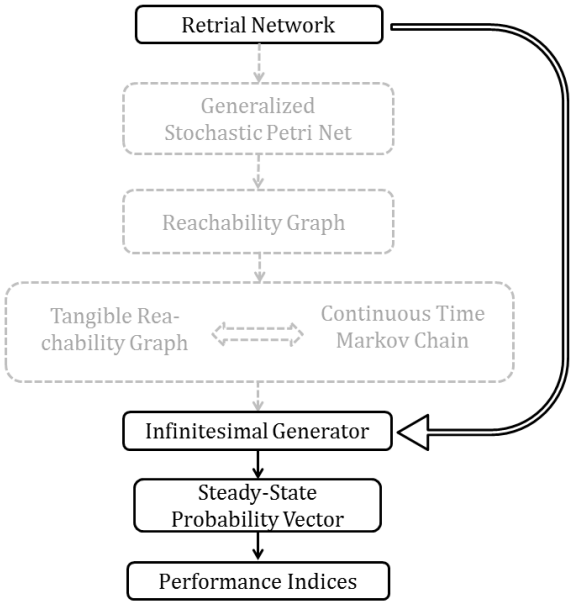


Figure 4. Our approach steps of retrial networks performance evaluation.

This section describes in detail, how to derive this algorithm [1]. We show that the discussion is the same for both service disciplines, whereas, CTMC we obtain for Random Service policy is different from Fastest Free Server policy's one. Consequently, the algorithms that generate the infinitesimal generator are different.

Initially, the orbit is empty, all customers are free and all servers are available. Thus, the initial marking can be expressed in this form:

$$\begin{aligned} M_0 &= \{M(Cus\_Free), M(Choice), M(Orbit), \\ &\quad M(Ser\_Idle1), M(Cus\_Serv1), M(Ser\_Idle2), \\ &\quad M(Cus\_Serv2)\} \\ &= \{L, 0, 0, S_1, 0, S_2, 0\} \end{aligned}$$

Whatever the values of L, S_1 and S_2, the conservation of the number of customers and servers of the two classes, gives the following equations:

$$\begin{cases} M(Ser\_Idle1) + M(Cus\_Serv1) = S_1 \\ M(Ser\_Idle2) + M(Cus\_Serv2) = S_2 \\ M(Cus\_Free) + M(Cus\_Serv1) \\ +M(Cus\_Serv2) + M(Orbit) = L \end{cases} \quad (1)$$

Observing these three equations, we note that the system state at steady-state can be described by means of three variables (i, j, k), which we call a *micro-state*, where:

- i represents the number of customers being served by servers of class C_1 (in place Cus_Serv1);
- j represents the number of customers being served by servers of class C_2 (in place Cus_Serv2);
- and k is the number of customers in orbit (in place $Orbit$).

Hence, having the micro-state (i, j, k), the markings of all places can be obtained, since

$$\begin{cases} M(Ser\_Idle1) = S_1 - i \\ M(Ser\_Idle2) = S_2 - j \\ M(Cus\_Free) = L - (i + j + k) \end{cases} \quad (2)$$

On the other hand, applying (1), we can deduce:

$$\begin{cases} 0 \leq i \leq S_1 \\ 0 \leq j \leq S_2 \\ 0 \leq k \leq L - (S_1 + S_2) \end{cases} \quad (3)$$

In fact, we introduce the concept of micro-state as a compact state description derived by the analysis of P-invariants of the model, so that it is always possible to define a one-to-one correspondence between the micro-states and the ordinary states of the classical approach.

The corresponding CTMC contains n micro-states corresponding to the accessible tangible markings, where n equals

$$n = (S_1 + 1) \cdot (S_2 + 1) \cdot (L + 1 - S) \text{ and } S = S_1 + S_2 \quad (4)$$

Figure 5 describes the CTMC corresponding to the Random Service policy model, while The CTMC resultant from the Fastest Free Server GSPN is given in Figure 6.

Thus, the corresponding infinitesimal generator Q is a $n \times n$ matrix, defined by:

$$\begin{cases} Q[(i,j,k),(x,y,z)] = \theta[(i,j,k),(x,y,z)] \\ Q[(i,j,k),(i,j,k)] = -\sum_{(l,m,n)\neq(i,j,k)} \theta[(i,j,k),(l,m,n)] \end{cases} \quad (5)$$

where $\theta[(i,j,k),(x,y,z)]$ is the transition rate from state (i,j,k) to state (x,y,z).

By analyzing the micro-states and the transition rates of each CTMC, we obtain the following rates, for the Random Service discipline :

- $[0 \leq i \leq S_1 - 1, 0 \leq j \leq S_2 - 1] : (i,j,k) \xrightarrow{\frac{1}{2}(L-i-j-k)\lambda} (i+1,j,k)$ and $(i,j,k) \xrightarrow{\frac{1}{2}(L-i-j-k)\lambda} (i,j+1,k)$
- $[0 \leq i \leq S_1 - 1] : (i,S_2,k) \xrightarrow{(L-i-S_2-k)\lambda} (i+1,S_2,k)$,
- $[0 \leq j \leq S_2 - 1] : (S_1,j,k) \xrightarrow{(L-S_1-j-k)\lambda} (S_1,j+1,k)$,
- $[0 \leq k < L - (S_1+S_2)] : (S_1,S_2,k) \xrightarrow{(L-S-k)\lambda} (S_1,S_2,k+1)$,
- $[i > 0] : (i,j,k) \xrightarrow{i\mu_1} (i-1,j,k)$,
- $[j > 0] : (i,j,k) \xrightarrow{j\mu_2} (i,j-1,k)$,
- $[0 \leq i \leq S_1 - 1, 0 \leq j \leq S_2 - 1, k > 0] : (i,j,k) \xrightarrow{\frac{1}{2}k\nu} (i+1,j,k-1)$ and $(i,j,k) \xrightarrow{\frac{1}{2}k\nu} (i,j+1,k-1)$,
- $[0 \leq i \leq S_1 - 1, k > 0] : (i,S_2,k) \xrightarrow{k\nu} (i+1,S_2,k-1)$,
- $[0 \leq j \leq S_2 - 1, k > 0] : (S_1,j,k) \xrightarrow{k\nu} (S_1,j+1,k-1)$,

As a consequence, the infinitesimal generator can be automatically calculated by means of Algorithm 1 given below.

In the same manner, rates $\theta[(i,j,k)(x,y,z)]$ of the Fastest Free Server discipline are given by :

- $[0 \leq i < S_1] : (i,j,k) \xrightarrow{(L-i-j-k)\lambda} (i+1,j,k)$,
- $[0 \leq j < S_2] : (S_1,j,k) \xrightarrow{(L-S_1-j-k)\lambda} (S_1,j+1,k)$,
- $[0 \leq k < L - S] : (S_1,S_2,k) \xrightarrow{(L-S-k)\lambda} (S_1,S_2,k+1)$,
- $[0 < i \leq S_1] : (i,j,k) \xrightarrow{i\mu_1} (i-1,j,k)$,
- $[0 < j \leq S_2] : (i,j,k) \xrightarrow{j\mu_2} (i,j-1,k)$,

Algorithm 1 Infinitesimal Generator Construction - Random Server Policy

```
for k ← 0, L − S do
    for i ← 0, S1 − 1 do
        for j ← 0, S2 − 1 do
            Q[(i,j,k),(i+1,j,k)] ← 1/2(L−i−j−k)λ
            Q[(i,j,k),(i,j+1,k)] ← 1/2(L−i−j−k)λ
            Q[(S1,j,k),(S1,j+1,k)] ← (L−S1−j−k)λ
        end for
        Q[(i,S2,k),(i+1,S2,k)] ← (L−i−S2−k)λ
    end for
end for
for k ← 0, L − S − 1 do
    Q[(S1,S2,k),(S1,S2,k+1)] ← (L − S − k)λ
end for
for k ← 0, L − S do
    for i ← 1, S1 do
        for j ← 0, S2 do
            Q[(i,j,k),(i−1,j,k)] ← iμ1
        end for
    end for
    for j ← 1, S2 do
        for i ← 0, S1 do
            Q[(i,j,k),(i,j−1,k)] ← jμ2
        end for
    end for
end for
for k ← 1, L − S do
    for i ← 0, S1 − 1 do
        for j ← 0, S2 − 1 do
            Q[(i,j,k),(i+1,j,k−1)] ← 1/2.kν
            Q[(i,j,k),(i,j+1,k−1)] ← 1/2.kν
        end for
        Q[(i,S2,k),(i+1,S2,k−1)] ← kν
    end for
    for j ← 0, S2 − 1 do
        Q[(S1,j,k),(S1,j+1,k)] ← kν
    end for
end for
```

- $[0 \leq i < S_1, 0 < k \leq L-S] : (i,j,k) \xrightarrow{k\nu} (i+1,j,k-1)$,
- $[0 \leq j < S_2, 0 < k \leq L - S)] : (S_1,j,k) \xrightarrow{k\nu} (S_1,j+1,k-1)$,

And the algorithm that generates the infinitesimal generator is given in Algorithm 2 (Fastest Free Server policy).

VI. Performance measures

The aim of this section is to derive the formulas of the most important stationary performance indices. As the

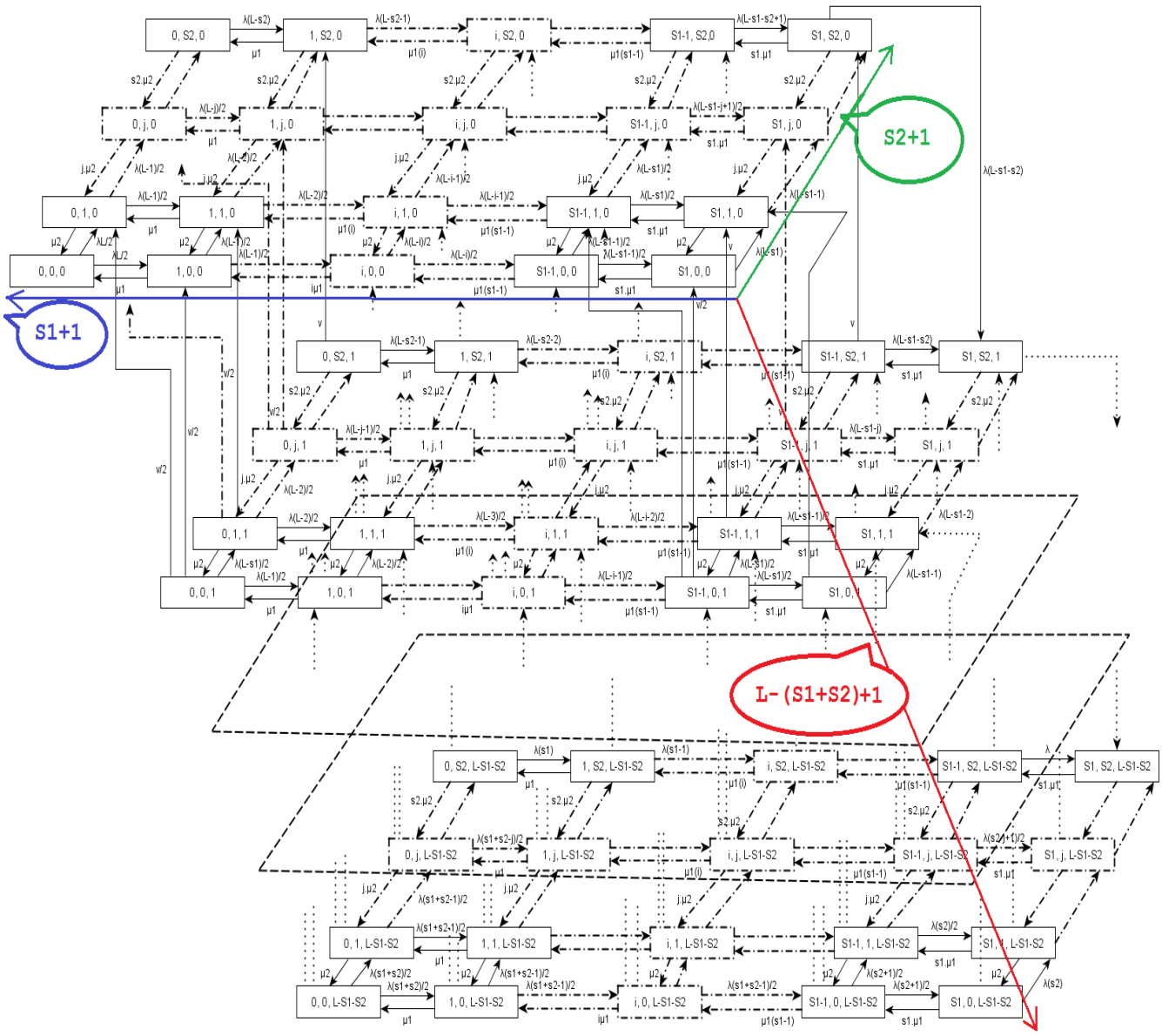


Figure 5. The CTMC describing finite-source retrial networks with two servers classes and Random Server discipline.

proposed models are bounded and the initial marking is a home state, the underlying process is ergodic. Hence, the steady-state solution exists and is unique. The infinitesimal generators Q corresponding to the proposed GSPN models can be obtained automatically by applying the above algorithms. Then, the steady-state probability vector π can be computed by solving the linear system of equations:

$$\begin{cases} \pi.Q = 0 \\ \sum_i \pi_i = 1 \end{cases} \quad (6)$$

where π_i denotes the steady-state probability that the process is in state M_i.

Having the probability distribution π, we can derive several exact stationary performance measures of finite-source retrial networks with two classes of servers, applying the formulas given below, which are based essentially on Equation (2) and the definition of the three variables i, j, and k given in Section V. In following, $M_i(p)$ indicates the number of tokens in place p in marking M_i, A is the set of reachable tangible markings, and $A(t)$ is the set of tangible markings reachable by transition t.

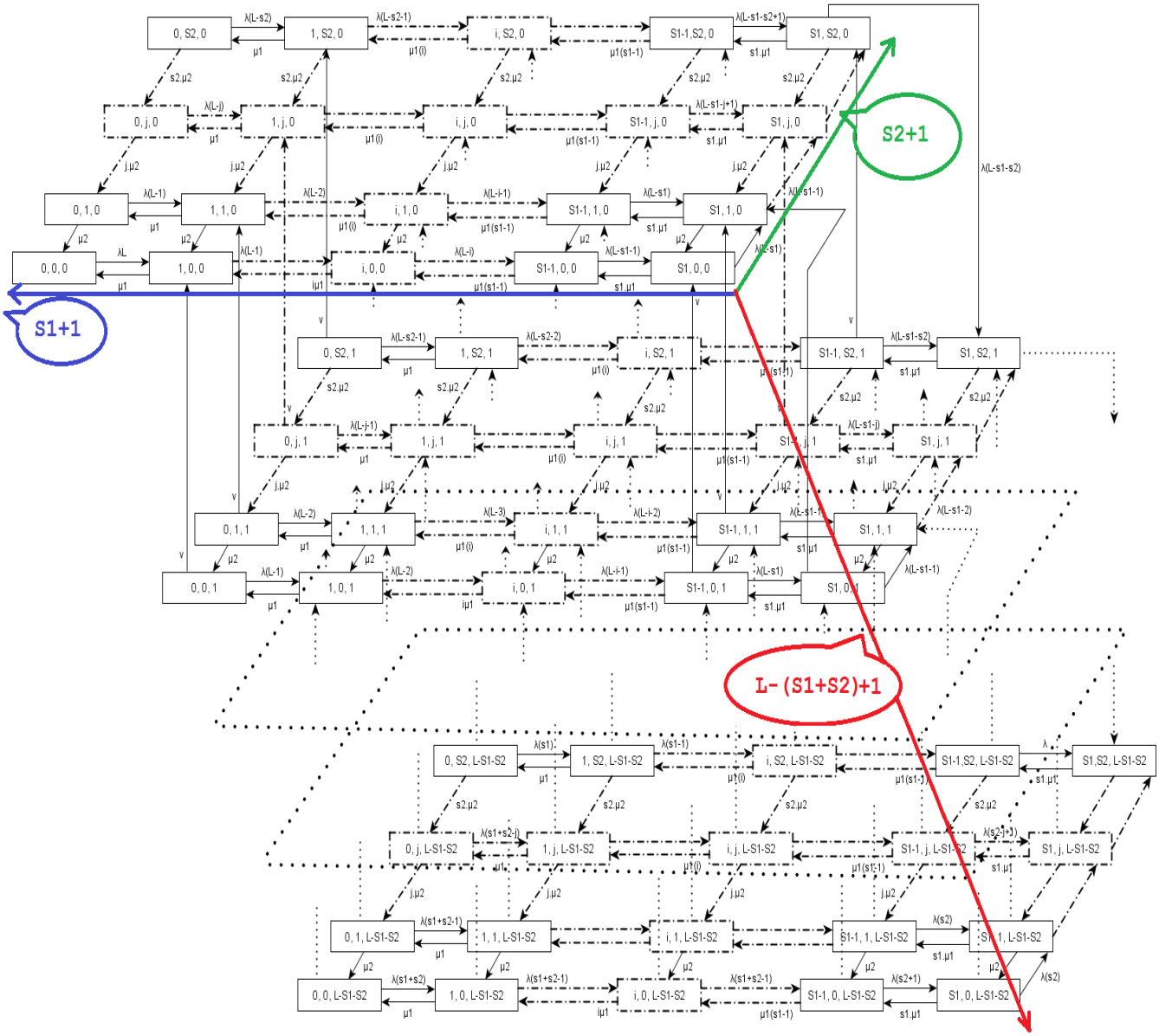


Figure 6. The CTMC describing finite-source retrial networks with two servers classes and Fastest Free Server discipline.

- `Mean number of free customers:` It corresponds to the mean number of tokens in place *Cus_Free*,

$$n_{CusFree} = \sum_{i:M_i \in A} M_i(Cus\_Free).\pi_i = \sum_{k=0}^{L-S} \sum_{i=0}^{S_1} \sum_{j=0}^{S_2} (L-i-j-k).\pi_{i,j,k}$$

- `Mean number of customers in the orbit:` This corresponds to the mean number of tokens in *Orbit*,

$$n_{Orb} = \sum_{i:M_i \in A} M_i(Orbit).\pi_i = \sum_{k=0}^{L-S} \sum_{i=0}^{S_1} \sum_{j=0}^{S_2} k.\pi_{i,j,k}$$

- `Mean number of busy servers of class`

Algorithm 2 Infinitesimal Generator Construction - Fastest Free Server Policy

for $k \leftarrow 0, L-S$ **do**
  for $i \leftarrow 0, S1-1$ **do**
    for $j \leftarrow 0, S2$ **do**
      $Q[(i,j,k),(i+1,j,k)] \leftarrow (L-i-j-k)\lambda$
    end for
  end for
  for $j \leftarrow 0, S2-1$ **do**
    $Q[(S1,j,k),(S1,j+1,k)] \leftarrow (L-S1-j-k)\lambda$
  end for
end for
for $k \leftarrow 0, L-S-1$ **do**
  $Q[(S1,S2,k),(S1,S2,k+1)] \leftarrow (L-S-k)\lambda$
end for
for $k \leftarrow 0, L-S$ **do**
  for $i \leftarrow 1, S1$ **do**
    for $j \leftarrow 0, S2$ **do**
      $Q[(i,j,k),(i-1,j,k)] \leftarrow i\mu_1$
    end for
  end for
  for $j \leftarrow 1, S2$ **do**
    for $i \leftarrow 0, S1$ **do**
      $Q[(i,j,k),(i,j-1,k)] \leftarrow j\mu_2$
    end for
  end for
end for
for $k \leftarrow 1, L-S$ **do**
  for $i \leftarrow 0, S1-1$ **do**
    for $j \leftarrow 0, S2$ **do**
      $Q[(i,j,k),(i+1,j,k-1)] \leftarrow k\nu$
    end for
  end for
  for $j \leftarrow 0, S2-1$ **do**
    $Q[(S1,j,k),(S1,j+1,k-1)] \leftarrow k\nu$
  end for
end for

C_1: Note that this is also the mean number of customers under service by Class C_1, it corresponds to the mean number of tokens in place *Cus_Serv1*,

$$\begin{aligned} n_{busyC_1} &= \sum_{i:M_i \in A} M_i(Cus\_Serv1).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2} i.\pi_{i,j,k} \end{aligned}$$

- `Mean number of busy servers of class` C_2: It is also the mean number of customers in service by Class C_2, and it represents the mean number of tokens in *Cus_Serv2*,

$$\begin{aligned} n_{busyC_2} &= \sum_{i:M_i \in A} M_i(Cus\_Serv2).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2} j.\pi_{i,j,k} \end{aligned}$$

- `Mean number of busy servers:`

$$\begin{aligned} n_{busy} &= n_{busyC_1} + n_{busyC_2} \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2}(i+j).\pi_{i,j,k} \end{aligned}$$

- `Mean number of customers in the system`: Which is the total number of the mean number of customers in the orbit and those under service (by C_1 and C_2),

$$\begin{aligned} n &= n_{Orb} + n_{busy} \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2}(i+j+k).\pi_{i,j,k} \end{aligned}$$

- `Mean number of free servers of class` C_1: This represents the mean number of tokens in place *Ser_Idle1*,

$$\begin{aligned} n_{FreeC_1} &= \sum_{i:M_i \in A} M_i(Ser\_Idle1).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2}(S_1-i).\pi_{i,j,k} \\ &= S_1 - n_{busyC_1} \end{aligned}$$

- `Mean number of free servers of class` C_2: This represents the mean number of tokens in place *Ser_Idle2*,

$$\begin{aligned} n_{FreeC_2} &= \sum_{i:M_i \in A} M_i(Ser\_Idle2).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2}(S_2-j).\pi_{i,j,k} \\ &= S_2 - n_{busyC_2} \end{aligned}$$

- `Mean number of free servers (of both classes):`

$$n_{Free} = n_{FreeC_1} + n_{FreeC_2} = S - n_{busy}$$

- `Effective customer arrival rate:` This represents the throughput of the transition *Arrival*,

$$\begin{aligned}\bar{\lambda} &= \sum_{i:M_i \in A(Arrival)} \lambda.M_i(Cus\_Free).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2} \lambda.(L-i-j-k).\pi_{i,j,k} \\ &= \lambda.n_{CusFree}\end{aligned}$$

- `Effective customer retrial rate:` It corresponds to the throughput of *Retrial* transition,

$$\begin{aligned}\bar{\nu} &= \sum_{i:M_i \in A(Retrial)} \nu.M_i(Orbit).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2} \nu.k.\pi_{i,j,k} \\ &= \nu.n_{Orb}\end{aligned}$$

- `Mean rate of` C_1 `service:` This corresponds to the throughput of the transition *Serv_End1*,

$$\begin{aligned}\bar{\mu_1} &= \sum_{i:M_i \in A(Serv\_End1)} \mu_1.M_i(Cus\_Serv1).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2} \mu_1.i.\pi_{i,j,k} \\ &= \mu_1.n_{busyC_1}\end{aligned}$$

- `Mean rate of` C_2 `service:` This corresponds to the throughput of *Serv_End2*,

$$\begin{aligned}\bar{\mu_2} &= \sum_{i:M_i \in A(Serv\_End2)} \mu_2.M_i(Cus\_Serv2).\pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2} \mu_2.j.\pi_{i,j,k} \\ &= \mu_2.n_{busyC_2}\end{aligned}$$

- `Total mean rate service:`

$$\bar{\mu} = \bar{\mu_1} + \bar{\mu_2}$$

- `Availability of` s `servers of class` C_1 $(1 \leq s \leq S_1)$`:` It's the probability that at least s servers of class C_1 are available

$$A_{sC_1} = \sum_{i:M_i(Ser\_Idle1) \geq s} \pi_i = \sum_{k=0}^{L-S}\sum_{i=0}^{S_1-s}\sum_{j=0}^{S_2} \pi_{i,j,k}$$

- `Availability of` s `servers of class` C_2 $(1 \leq s \leq S_2)$`:` It's the probability that at least s servers of class C_2 are available

$$A_{sC_2} = \sum_{i:M_i(Ser\_Idle2) \geq s} \pi_i = \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0}^{S_2-s} \pi_{i,j,k}$$

- `Availability of` s `servers in the system (among both classes):`

$$\begin{aligned}A_s &= \sum_{i:M_i(Ser\_Idle1)+M_i(Ser\_Idle2) \geq s} \pi_i \\ &= \sum_{k=0}^{L-S}\sum_{i=0}^{S_1}\sum_{j=0,i+j\leq S-s}^{S_2} \pi_{i,j,k}\end{aligned}$$

- `Utilization of at least` s `servers of the class` C_1`:` This corresponds to the probability that at least s servers of class C_1 are busy

$$U_{sC_1} = \sum_{i:M_i(Cus\_Serv1) \geq s} \pi_i = \sum_{k=0}^{L-S}\sum_{i=s}^{S_1}\sum_{j=0}^{S_2} \pi_{i,j,k}$$

- `Utilization of` s `servers at least, of the class` C_2`:` This corresponds to the probability that at least s servers of class C_2 are busy

$$U_{sC_2} = \sum_{i:M_i(Cus\_Serv2)\geq s} \pi_i = \sum_{k=0}^{L-S} \sum_{i=0}^{S_1} \sum_{j=s}^{S_2} \pi_{i,j,k}$$

- Utilization of s servers at least in the system (among the two classes):

$$\begin{aligned} U_s &= \sum_{i:M_i(Cus\_Serv1)+M_i(Cus\_Serv2)\geq s} \pi_i \\ &= \sum_{k=0}^{L-S} \sum_{i=0}^{S_1} \sum_{j=0,i+j\geq s}^{S_2} \pi_{i,j,k} \end{aligned}$$

- The blocking probability of a primary customer:

$$B_p = \frac{\sum_{k=0}^{L-S} (L-k-S).\lambda.\pi_{S_1,S_2,k}}{\bar{\lambda}}$$

- The blocking probability of a repeated call:

$$B_r = \frac{\sum_{k=1}^{L-S} k.\nu.\pi_{S_1,S_2,k}}{\bar{\nu}}$$

- The blocking probability:

$$B = B_p + B_r$$

- The mean waiting time: It's the mean period between the arrival of the customer and its service beginning. Using the Little's formula, the mean waiting time is given by :

$$\bar{W} = \frac{n_{Orb}}{\bar{\lambda}}$$

- The mean response time:

$$\bar{R} = \frac{n}{\bar{\lambda}}$$

- The mean service time:

$$\bar{S} = \bar{R} - \bar{W} = \frac{n_{busy}}{\bar{\lambda}}$$

VII. Validation and Numerical Examples

In order to test the feasibility of our approach, we developed a $C\#$ code to implement the above algorithms 1 and 2 as well as the performance indices formulas. Next, we tested it for a large number of examples. In particular, in the homogeneous case, by assuming $\mu_1 = \mu_2$, the results were validated by the Pascal program given in [9]. From Table I, we can see that both models give exactly the same results up to the sixth decimal digit.

Table I
Validation in the Homogeneous Case

	Homogeneous case	Two servers classes system
Number of servers	4	S_1= 1,S_2= 3
Size of source	20	20
Primary call generation rate	0.1	0.1
Service rate	1	μ_1= 1, μ_2= 1
retrial rate	1.2	1.2
Mean number of busy servers	1.800 748	C_1: 0.521 865 C_2: 1.278 883 Tot.: 1.800 748
Mean number of source in orbit	0.191 771	0.191 771
Mean primary call generation rate	1.800 748	1.800 748
Mean waiting time	0.106 495	0.106 495

In the following, we present sample numerical results to illustrate graphically the impact of some system parameters, namely, the primary call rate, retrial rate, and the servers number in both classes, on some performance indices, which are the mean response time and the blocking probability, in both policies cases; Random Server and Fastest Free Server. We also consider the *Averaged Random case*, where we assume the same number of servers in the network ($S = S_1 + S_2$), all homogeneous with the average service rate μ, whose formula is given by:

$$\mu = \frac{\mu_1.S_1 + \mu_2.S_2}{S_1 + S_2} \quad (7)$$

The customers requests are assigned to idle servers randomly.

The input parameters of Random and Fastest Free Server policies are collected in Table II, while those of the Averaged Random case are summarized in Table III.

Table II
INPUT NETWORK PARAMETERS

	L	S_1	S_2	λ	ν	μ_1	μ_2
Figure 7, 11	50	5	2	x axis	0.1	8	2
Figure 8, 12	50	4	2	0.5	x axis	8	2
Figure 9, 13	30	x axis	4	2	1	6	1
Figure 10, 14	30	4	x axis	2	1	6	1

Table III
INPUT NETWORK PARAMETERS FOR THE AVERAGED RANDOM CASE

	L	S	λ	ν	μ
Figure 7, 11	50	7	x axis	0.1	6.28
Figure 8, 12	50	6	0.5	x axis	6

In Figure 7, we study the primary call generation rate variation effect on the mean response time. As we can see, this latter increases with the intensity of the flow of primary calls. This is due to the increase of waiting time of customers in the orbit. Furthermore, the mean response time of Fastest Free Server discipline is always shorter than the one of Random Server discipline. The curve of the Averaged Random case is situated in the middle.

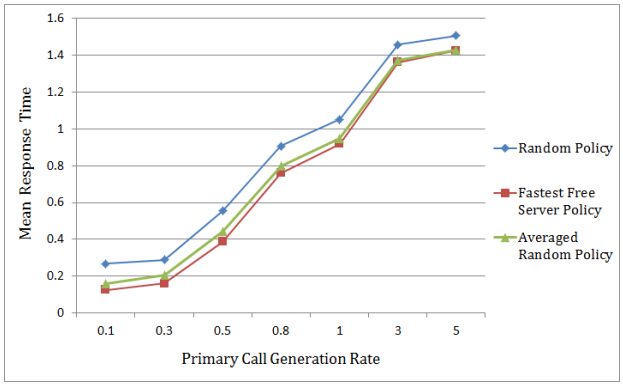


Figure 7. Mean response time versus primary call generation rate.

The Figure 8 shows the sensitivity of the mean response time to the retrial generation rate, for both service disciplines. Indeed, the response time decreases with the intensity of the flow of repeated calls, particularly when the retrial intensity is low (between 0.01 and 0.3), beyond the value 0.3, the influence becomes less significant. In addition, performance obtained with the discipline of the Fastest Free Server and Averaged Random case are always better than the one obtained by Random Server discipline.

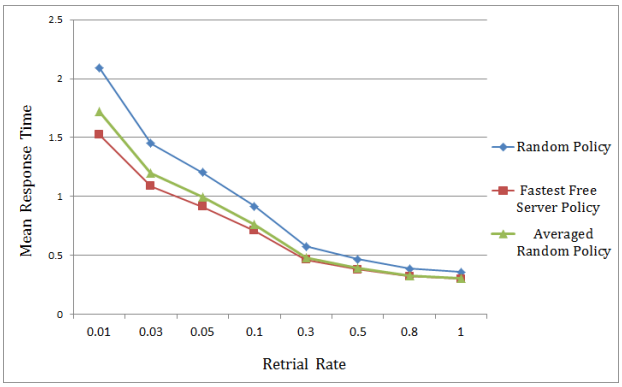


Figure 8. Mean response time versus retrial generation rate.

In Figure 9, (10 respectively), we show the influence of the number of servers of C_1 (C_2 respectively) class, on the mean response time.

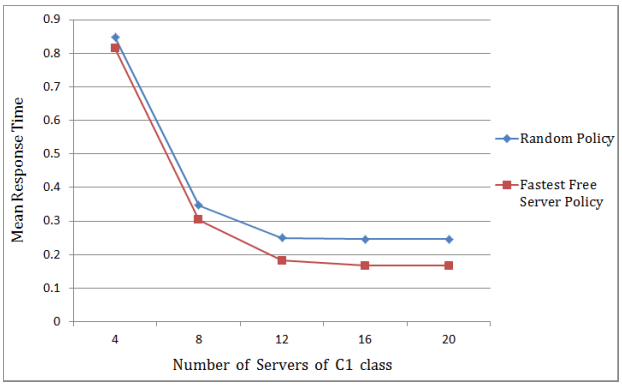


Figure 9. Mean response time versus C_1 class servers number.

We conclude that the mean response time decreases with the increase of the number of servers. However, the rate of influence of the number of servers in the C_1 class is faster than the influence due to increasing number of servers of C_2 class, because the former is faster ($\mu_1 = 6$ vs $\mu_2 = 1$). In Figure 9, the response time reached the optimum and stabilized after a certain time (number of servers = 12). Hence, it is not interesting to invest in new servers in the C_1 class.

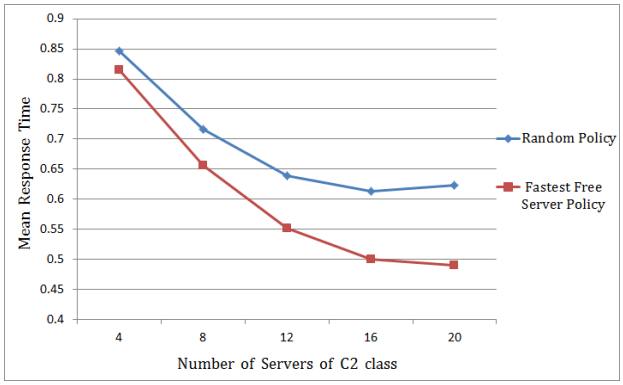


Figure 10. Mean response time versus C_2 class servers number.

In Figure 10, the surprising increase in the mean response time in Random Service policy case when having more C_2 class servers (from 16 to 20 servers) reveal another weakness of this policy. Actually, the slowest servers number becomes much greater than the fastest one, and as customers requests are assigned to servers randomly, fastest servers have less chance to catch a customer request (4 servers in C_1 class vs 20 of C_2 class), and C_2 class servers tend to take most of customers requests, this results in increasing the mean service time, and consequently the mean response time.

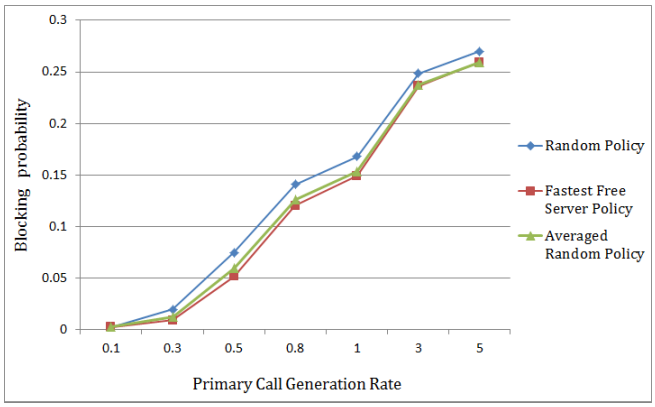


Figure 11. Blocking probability versus primary call generation rate.

As it can be seen in Figures 11 and 12, the blocking probability depends on both primary call generation rate and retrial rate, the increase of these latter involves the increase of the blocking probability in both Random Server and Fastest Free Server policies. But, as we can see on the curves, Fastest Free Server policy gives always values slightly better than those obtained in the Averaged Random case, and the results of this latter are better than those given by Random Server policy, which means that the Fastest Free server policy gives better performance.

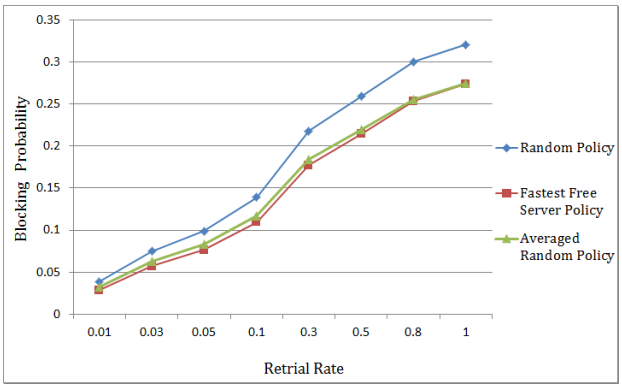


Figure 12. Blocking probability versus retrial generation rate.

In Figures 13 and 14, the blocking probability is displayed as a function of servers number in C_1 and C_2 classes respectively. As to be expected, the blocking probability is higher when having a few number of servers, and it decreases as the number of servers rises in the system, the decrease is more significant in case of C_1 class, because it is supposed to be faster. Moreover, the performance of Fastest Free Server discipline and Random Server discipline are almost the same.

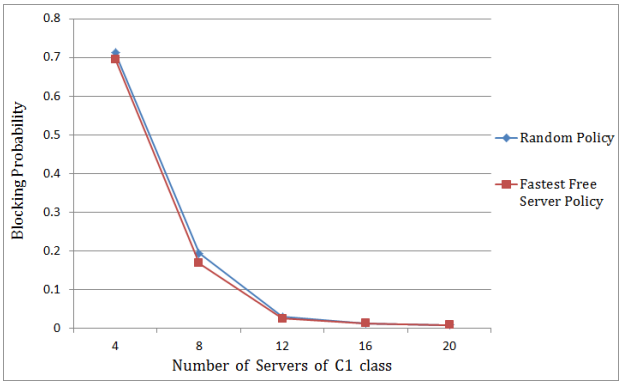


Figure 13. Blocking probability versus C_1 class servers number.

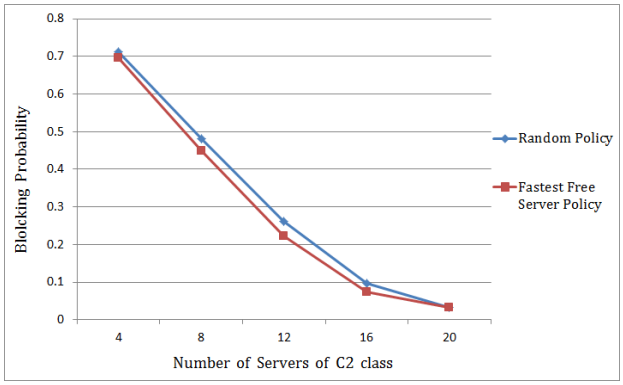


Figure 14. Blocking probability versus C_2 class servers number.

VIII. Conclusion and Future Work

In [1], we have proposed a technique based on GSPNs to analyze finite-source retrial networks with two servers classes, using Random Server discipline. In the current paper, we have extended this idea by considering another service discipline, which is the Fastest Free Server. Hence, we investigated and compared the two service disciplines.

The advantage of our approach is the automatic computation of the infinitesimal generator for both disciplines, applying the given algorithms, and without need to generate neither the reachability graph nor the underlying Markov chain. We have also developed formulas of the main stationary performance indices based on stationary probabilities and network parameters. Furthermore, we studied the effect of network parameters on performance indices, and proved through some numerical examples, that Fastest Free Server discipline gives more favorable system performance than both Random Server discipline and Averaged Random case, which is the equivalent homogeneous network with the average service rate.

References

[1] N. Gharbi and L. Charabi, *An Algorithmic Approach for Analyzing Wireless Networks with Retrials and Heterogeneous Servers*, The seventh International Conference on Wireless and Mobile Communications, ICWMC, 2011, Luxembourg, pp. 151-156, ISBN:978-1-61208-008-6.

[2] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *Modelling with Generalized Stochastic Petri Nets*, 1994, New York, NY, USA, John Wiley & Sons, Inc., ISBN-13: 9780471930594.

[3] J.R. Artalejo and A. Gómez-Corral, *Retrial Queueing Systems: A Computational Approach*, 2008, Berlin, Springer Berlin Heidelberg, ISBN-13: 978-3642097485.

[4] J.R. Artalejo, *Accessible bibliography on retrial queues: Progress in 2000-2009*, Mathematical and Computer Modelling, 2010, vol. 51, pp. 1071-1081.

[5] J.R. Artalejo and M.J. Lopez-Herrero, *Cellular mobile networks with repeated calls operating in random environment*, Computers & operations research, 2010, vol. 37, no. 7, pp. 1158-1166.

[6] M. Diaz, *Les réseaux de Petri - Modèles Fondamentaux*, 2001, Paris, Hermès Science Publications, ISBN-13: 978-2-7462-0250-4.

[7] D. Efrosinin and L. Breuer, *Threshold policies for controlled retrial queues with heterogeneous servers*, Annals of Operations Research, 2006, vol. 141, pp. 139-162.

[8] D. Efrosinin and J. Sztrik, *Performance Analysis of a Two-Server Heterogeneous Retrial Queue with Threshold Policy*, Quality Technology and Quantitative Management, 2011, vol. 8, no. 3, pp. 211-236.

[9] G.I. Falin and J.G.C. Templeton, *Retrial Queues*, 1997, London, Chapman and Hall, ISBN-13: 978-0412785504.

[10] N. Gharbi, C. Dutheillet, and M. Ioualalen, *Colored Stochastic Petri Nets for Modelling and Analysis of Multiclass Retrial Systems*, Mathematical and Computer Modelling, 2009, vol. 49, pp. 1436-1448.

[11] J. Roszik and J. Sztrik, *Performance analysis of finite-source retrial queues with nonreliable heterogeneous servers*, Journal of Mathematical Sciences, 2007, vol. 146, pp. 6033-6038.

[12] J. Sztrik, G. Bolch, H. De Meer, J. Roszik, and P. Wüchner, *Modeling finite-source retrial queueing systems with unreliable heterogeneous servers and different service policies using MOSEL*, Proc. of 14th Inter. Conf. on Analytical and Stochastic Modelling Techniques and Applications, ASMTA'07, 2007, Prague, Czech Republic, pp. 75-80.

[13] T. V. Do, *A new computational algorithm for retrial queues to cellular mobile systems with guard channels*, Computers & Industrial Engineering, 2010, vol. 59, pp. 865-872.

[14] P. Tran-Gia and M. Mandjes, *Modeling of customer retrial phenomenon in cellular mobile networks*, IEEE Journal on Selected Areas in Communications, 1997, vol. 15, pp. 1406-1414.

[15] P. Wüchner, J. Sztrik, and H. De Meer, *Modeling Wireless Sensor Networks Using Finite-Source Retrial Queues with Unreliable Orbit*, Proc. of the Workshop on Performance Evaluation of Computer and Communication Systems, PERFORM'2010, 2010, pp. 73-86.

Wireless Cooperative Relaying Based on Opportunistic Relay Selection

Tauseef Jamal[1], Paulo Mendes[1], André Zúquete[2]
[1]SITI, R&D Unit of Informatics Systems and Technologies, Universidade Lusofona de Humanidades e Tecnologias (ULHT), COFAC, Campo Grande 376, Lisbon, Portugal
{tauseef.jamal,paulo.mendes}@ulusofona.pt
[2]Dep. Electronics, Telecommunications and Informatics, IEETA, University of Aveiro Campus Universitário de Santiago 3810–193 Aveiro, Portugal
andre.zuquete@ua.pt

***Abstract*—Advances in wireless technologies, including more powerful devices and low cost radio technologies, have potential to drive an ubiquitous utilization of Internet services. Nevertheless wireless technologies face performance limitations due to unstable wireless conditions and mobility of devices. In face of multi-path propagation and low data rate stations, cooperative relaying promises gains in performance and reliability. However, cooperation procedures are unstable (rely on current channel conditions) and introduce overhead that can endanger performance especially when nodes are mobile. In this article we describe a framework, called RelaySpot, to implement cooperative wireless solutions in large mobile networks, based upon opportunistic relay selection methods. RelaySpot based solutions are expected to minimize signaling exchange, remove estimation of channel conditions, and improve the utilization of spatial diversity, minimizing outage and increasing reliability.**

***Index Terms*—Cooperative Relay Scheduling, Opportunistic Relay Selection, Wireless Resource Management, Space-Time Diversity.**

I. Introduction

Over the past decade, Internet access became essentially wireless, with 802.11 technologies providing a low cost broadband support for a flexible and easy deployment. However, channel conditions in wireless networks are subjected to interference and multi-path propagation, creating fading channels and decreasing the overall network performance. While fast fading can be mitigated by having the source retransmitting packets, slow fading, caused by obstruction of the main signal path, makes retransmission useless, since periods of low signal power lasts for the entire duration of the transmission.

Extensive research has been done to mitigate the impact of shadowing in wireless networks, being mostly focused on *Multiple-Input Multiple-Output* (MIMO) systems. Recently, cooperative relaying techniques have been investigated to increase the performance of wireless systems by using diversity created by different single antenna devices, aiming to reach the same level of performance of MIMO systems.

Cooperation occurs when overhearing relays assist the transmission from source to destination, by transmitting different copies of the same signal from different locations, allowing the destination to get independently faded versions of the signal that can be combined to obtain an error-free signal.

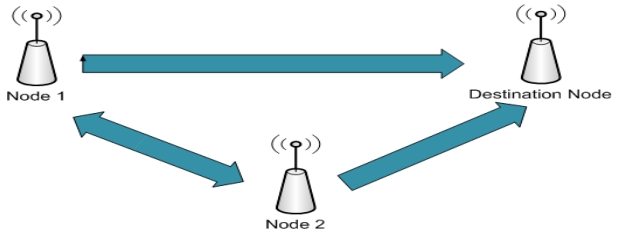


Figure 1. Cooperative relaying

Figure 1 shows a pair of single antenna devices able to act as relays of each other by forwarding some version of "overheard" packets along with its own data. Since the fading channels of two different devices are statistically independent, this generates spatial diversity. The development of cooperative relaying systems, of which Figure 1 illustrates a simple scenario, raises several research issues including the performance impact on the relay itself, and the interference on the overall network, leading to a potential decrease in network capacity and transmission fairness.

In this paper, we present our arguments in favor of a new type of cooperative relaying scheme based upon local decisions that do not rely on unstable information e.g., *(Channel State Information*) CSI collected over multiple links. We describe an 802.11 backward compatible cooperative relaying framework, called *RelaySpot* [1], which aims to ensure accurate and fast relay selection, posing minimum overhead and reducing the dependency upon CSI estimations, which is essential to increase system performance in scenarios with mobile nodes. The basic characteristic of any RelaySpot-based solution is the capability to perform local relaying decisions at potential relay nodes (can be more than one), based on a combination of opportunistic relay selection and cooperative relay scheduling. Intermediate nodes take the opportunity to relay in the presence of local favorable conditions (e.g., no concurrent traffic). Cooperative scheduling is used to compensate unsuccessful relay transmissions. To the best of our knowledge RelaySpot is the first framework that aims to create the basic conditions to allow relay selection to be done without relying on CSI estimation.

The remaining of this paper is organized as: section II describes the concept of cooperative relaying. Section III describes the prior-art. In sections IV and V we describe the proposed RelaySpot mechanism. Section VI provides an operational comparison with an example of source-based relaying approach (CoopMAC [2]) with RelaySpot. While RelaySpot implementation is discussed in section VII. Section VIII concludes the paper.

II. Cooperative Relaying

The basic problem of wireless communication systems is the delivery of information from one network node to another in a resource-efficient manner. While, wireless links always had orders of magnitude less bandwidth than their wired counterparts, newer technologies, such as multiple-input multiple-output (MIMO) systems, are starting to improve the performance of wireless network. However such improvements come at the cost of multiple Radio Frequency (RF). Furthermore, the size of mobile devices may limit the number of antennas to be deployed.

In 802.11 networks, the low quality (throughput and reliability) and short coverage of the direct link between a source and a destination are mainly due to the shadowing and fading effects of the wireless environment [3]. There are however other constrains in wireless networks such as limited power, size of devices, and distance. Due to the distance from the Access Point (AP), a mobile node can observe a bad channel as compared to other nodes that are closer to the AP. Figure 2 shows the transmission characteristics of some nodes, as a result of the rate adaptation functionality of IEEE 802.11: stations closer to the AP transmit at high data rates, while stations far away from the AP decrease their data rate after detecting missing frames.

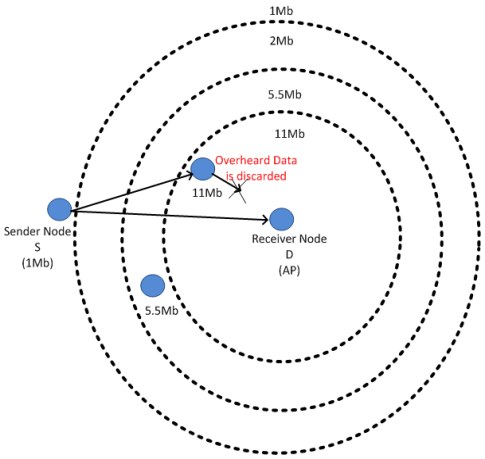


Figure 2. 802.11 rate adaptation

The usage of rate adaptation schemes results in a degradation of the overall network performance, since low data rate stations grab the wireless medium for a longer time. This occurs since each station has the same probability to access the channel, which means that high data rate stations will not be able to keep the desirable throughput.

As illustrated in Figure 2, the station at the cell edge adapts its data rate to 1Mbps, yet its frames are overheard by the high data rates stations. The latter ignores this overheard information and drops the frames. Cooperative relaying is a very simple, and yet effective solution, to mitigate the problems raised by the presence of low data rate stations. With cooperative relaying, high data rate stations help low data rate stations to release the medium earlier, by relaying their data over channels with higher data rates. This way high data rate stations will be able to transmit earlier, increasing the overall system performance. In cooperative communications, nodes in a wireless network work together to form a virtual antenna array.

The basic ideas behind cooperative communication can be traced back to the relay channel model in information theory extensively studied in the 1970s by Cover and El Gamal [4]. Recent research on cooperative communication [5], [6], [7] demonstrates the benefits of cooperative relaying in a wireless environment by achieving spatial diversity. Moreover, most of the research being done focuses on the physical layer (cooperative communications), by exploiting spatial diversity to increase system reliability of cellular networks. Recently, the exploitation of link-layer diversity (cooperative relaying) in cellular and multi-hop wireless networks has attracted considerable research attention. Cooperative techniques utilize the broadcast nature of wireless signals: the source node sends data for a particular destination, and such data can be "overheard" at neighboring nodes; these neighboring nodes, called relays, partners, or helpers, process the data they overhear and transmit it towards the destination; the destination receives the data from the relay or set of relays (on behalf of the source) enabling higher transmission rate, or combines the signals coming from the source and the relays enabling robustness against channel variations. Such spatial diversity arising from cooperation is not exploited in current cellular, wireless LAN, or ad-hoc systems. Hence, cooperative relaying is different from traditional multi-hop or infrastructure based methods. Therefore, for cooperation to be implemented at the link layer, link layer needs to be changed in order to allow indirect transmission between source and destination.

At the link layer, IEEE 802.11 uses the CSMA/CA algorithm to control medium access, being the *Distributed Coordination Function* (DCF) the most common operation mode. In scenarios with fading channels and low data rate stations, high throughput, reliability, and coverage may be possible to achieve with an efficient cooperative *Medium Access Control* (MAC) layer based on a modifying version of the DCF signaling scheme. Like Ethernet, it first checks to see that the radio link is clear before transmitting. To avoid collisions, stations use a random back-off after each frame, with the first transmitter (the one with shortest random time) seizing the channel. Carrier sensing is used to determine if the medium is available. Two types of carrier sensing functions in 802.11 manage this process: the physical carrier-sensing and virtual carrier-sensing functions [8]. If either carrier-sensing function indicates that the medium is busy, the MAC reports this to higher layers. Virtual carrier-sensing is provided by the Network Allocation Vector (NAV). Most 802.11 frames

carry a duration field, which can be used to reserve the medium for a fixed time period. The NAV is a timer that indicates the amount of time the medium will be reserved. Stations set the NAV to the time for which they expect to use the medium, including any frames necessary to complete the current operation. Other stations count down from the NAV to zero. When the NAV is not zero, the virtual carrier-sensing function indicates that the medium is busy; when the NAV reaches zero, the virtual carrier-sensing function indicates that the medium is idle. Figure 3 shows the virtual carrier sensing with usage of optional RTS/CTS signaling.

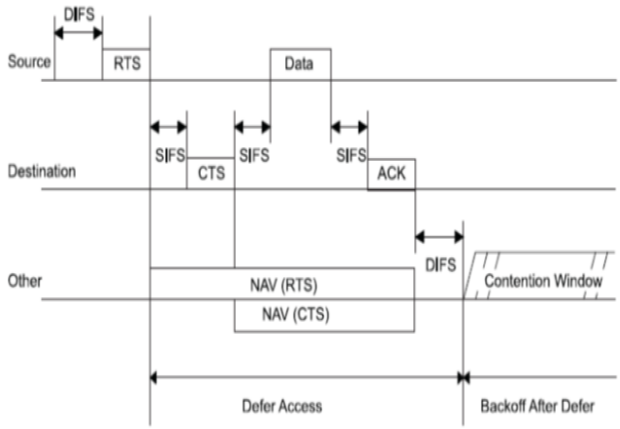


Figure 3. NAV propagation mechanism [8]

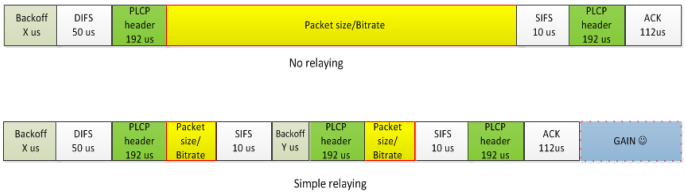


Figure 4. Simple relaying gain

Relaying involves transmission of two data frames separated in time and space; therefore, it introduces overhead, which increases due to additional control messages. However significant gain can be achieved by a careful selection of reservation duration and back-off timings. Figure 4 shows the gain of cooperative relaying in 802.11 (when there is no extra control message). As seen in Figure 4 a regular data transmission with acknowledgment takes longer to send data when compared to the data transmission based on a relay protocol. With a relay protocol the relatively slow stations would reserve the channel for a duration of *frame_size/(fast_data_rate=11Mbps)* instead of *frame_size/(slow_data_rate=1Mbps)* and the other stations will benefit from this with higher probability of accessing the channel.

Cooperative relaying can be divided into two major parts: i) relay transmission protocol (relaying protocol); ii) relay selection. Cooperative relaying protocols can be classified into proactive schemes and reactive schemes. In the former, the cooperation from relay(s) is always provided either by a prearranged or a random set of relay(s) before the acknowledgment (ACK or NACK) from the receiver. In the latter, the help from the relay(s) is initiated only when the direct transmission fails (lack of ACK or overheard NACK). Irrespective of the class of relaying protocol, the operation can be opportunistic or cooperative. Cooperative relaying protocols are normally initiated by source or destination, where relays are selected prior to data transmission. Such protocols require additional control/handshake messages which pose additional overhead. In the case of opportunistic relaying protocols, the relay(s) opportunistically forward the overheard data to destination, and the destination acknowledges the reception of data by sending ACK to the source. Source and destination may not have prior knowledge of selected relay(s). Such mechanisms are prone to collision as there is no coordination between nodes.

The definition of MAC cooperative schemes poses several challenges, specially in the presence of mobile nodes. A major challenge is related to relay selection, which aims to identify the most suitable relay(s) for assisting transmissions between any pair of nodes. Research is ongoing to devise efficient relay selection at MAC layer, being the proposed approaches mostly source or destination based. In the former case, the source maintains a table with *Channel State Information* (CSI) of neighboring devices to support relay selection. In destination-based approaches, the destination decides whether to use relaying or not, based on thresholds and CSI kept on the destination and on potential relays. Both approaches incur in some overhead (specially source-based) and are not efficient reacting to network changes, mainly in the presence of mobile nodes.

III. State Of The Art

This section provides an analysis of the most significant contributions aiming to devise efficient cooperative relaying mechanisms, able to take advantage of available relay nodes. First of all, a study of backward compatible 802.11 cooperative MAC proposals (relay transmission protocol) is presented: such proposals can serve as a basic ground for further developments. Second, central aspects of cooperative relaying (relay selection) are analyzed.

A. Cooperative MAC

Initial work in cooperative networking was mainly focused on physical layer approaches aiming to achieve higher spatial diversity. Although previous work shows the benefit of cooperation in wireless networks, it does not define medium access methods that would support new cooperative schemes. To take full advantage of physical layer cooperative techniques, new MAC schemes must change the transmitter-receiver communication model to include a transmitter-relay(s)-receiver model. Common examples of MAC source-based cooperative relaying schemes are the ones that use one relay [2], [9] or two relays in parallel [10]. Source-based relaying approaches require the sources to maintain a table of CSI that is updated by potential relays based upon periodic broadcasts. As an example, with

CoopMAC [2], the source can use an intermediate node (called helper) that experiences relatively good channel with the source and the destination. Instead of sending frames directly to the destination at a low transmission rate, the source makes use of a two-hop high data rate path to the destination via a helper. In case of CoopMAC, potential helpers overhear ongoing RTS/CTS transmissions for measuring the source-helper and helper-destination CSI. Based on the CSI broadcasted by potential helpers, sources update a local table (cooptable) used to select the best relay for each transmission. Another example of source based relaying is CODE[10], which uses multiple relays based on network coding. In CODE all nodes overhear RTS/CTS frames, and if they find that they can transmit data faster than the source, they add the identity of source and destination to their willingness list. Once the source finds its address in the willing list of relay(s), it adds those relay(s) into its cooperative table. The major difference between CoopMAC and CODE is that with the latter, a source selects two relays with latest feedback time, forming a cooperative diamond. The usage of RTS-CTS frames is also different. Source-based approaches undergo two main problems: channel estimation and periodic broadcasts, which introduce overhead that is problematic in mobile scenarios.

While source-based proposals follow a proactive approach, reactive cooperative methods [11], [12] rely on relays to retransmit on behalf of the source when the direct transmission fails. An example is PRO [11], which selects relays among a set of overhearing nodes in two phases: first, a local qualification process takes place at potential relays, during which the link quality is compared with some predefined threshold, leading to the identification of qualified relays. In a second phase, qualification information is broadcasted, allowing qualified relays to set scheduling priorities. Reactive approaches face the same challenges of source-based methods. CoRe-MAC [13] is another reactive Cooperative MAC protocol. In CoRe-MAC, when a NACK is overheard, candidate relays send an AFR (Apply For Relay) message to the destination within a fixed number of slots. After receiving non colliding AFRs, the destination elects the best relay in term of the highest received SNR. However the destination does not know which is the suitable number of AFR messages to wait for, in order to reach a good decision. Moreover, the extra handshake messages introduce significant overhead in case of relay failure.

N. Marchenko et al. propose a mechanism [14] where all overhearing nodes estimate the *Signal-to-Noise Ratio* (SNR) for both source-relay and relay-destination channels, based on which they can nominate themselves as potential relays. Potential relays send a nomination message to the destination, by selecting a slot in the contention window, and the destination selects a most suitable relay among all the nominated nodes. This proposal has several drawback: i) geographic position of nodes is assumed to be known; ii) the size of the contention window has great influence in selecting the best relay; iii) the destination node is not aware of the number of nominated relays.

In the case of multi-hop networks the performance gain of cooperative relaying may be exploited by finding a node that assists the transmission for every hop. Although the gain achieved through cooperative diversity increases robustness, it requires retransmissions reducing network capacity. Such a hop base cooperation scheme neglects a crucial evidence: not only the destination of a data might be in need of help but also the next hop. An alternative approach may be to use two-in-one cooperation [12], in which a single retransmission can improve the success probability of two ordinary transmissions (source to next-hop and next-hop to destination), leading to a better usage of the network capacity. In two-in-one cooperation all potential relays react after detecting a missing *Acknowledgment* (ACK) from the destination. Although two-in-one cooperation can achieve a diversity gain of three, the most suitable relay selection scheme is not investigated.

B. Relay Selection

In what concerns relay selection mechanisms, the basic mechanism defines an opportunistic behavior in which all overhearing nodes estimate the CSI of sender-node and node-destination links, based on which they set a timer such that nodes with better channel conditions broadcast first their qualification as relays, or even data to be relayed [15]. Such mechanisms present a high probability of collision, as well as low efficiency in mobile scenarios due to CSI measurements. Nevertheless, opportunistic relaying has been modified aiming to increase its efficiency level [16], [17]. Although most of the related work considers opportunistic relaying, it may lead to data collision if more than one relay is selected [18]. Collisions may be avoided by using a suitable resource allocation scheme, or by using a relay only when needed, which lead to the need to devise a relay on demand mechanism. For instance, with relaying on demand [19], the basic relay selection mechanism [15] is modified with the introduction of a receiver threshold aiming to improve energy savings. With on-demand approaches nodes with bad channel conditions do not participate in relay selection. However, such approaches still rely upon RTS/CTS for channel estimation, leading to high overheads.

Other kind of relay selection mechanisms rely on geographical information [20]. Such approaches assume that users' location is known, based for example on information from GPS, and Packet Error Rate (PER) is used as metric for selecting relays. It relies on constant/known channel statistics in terms of fading Probability Density Function (PDF), fading auto correlation function, and path loss exponent. In scenarios where the users are moving fast such parameters cannot be assumed to be known, which limits the potential of this type of approaches.

A proposal to group and select set of relays for cooperative networks is presented by A. Nosratinia et al. [21], in which each node has data of its own to transmit, and cooperation may be non-reciprocal. The study of non-reciprocal approaches to relay allocation brings several benefits, since with distributed algorithms nodes make individual decisions about cooperation. A. Nosratinia et al. [21] investigate the effect of allocation policies on system performance, and how the cooperative gain scales with the number of cooperating nodes, such that each node can decode message with high probability. In terms of

the outage probability, it assumes that each node may help n other nodes, and the selection strategy guarantees diversity n+1 for all transmissions. However, as n+1 nodes take part in one transmission, the system complexity is considerably high. Moreover, this work assumes that small scale fading is not dominated by path loss, which points to networks of up to a certain coverage area.

For better understanding of the different type of relay selection schemes, T. Jamal and P. Mendes [22] devised a comprehensive analysis and taxonomy.

IV. RELAYSPOT

Relay selection is a challenging task, since it greatly affects the design and performance of a cooperative network. On the one hand, cooperation is beneficial for the network, but on the other hand it introduces extra overhead (e.g., CSI estimation). The major goal of *RelaySpot* is to minimize overhead introduced by cooperation, with no performance degradation.

Unlike previous work, RelaySpot does not require maintenance of CSI tables, avoiding periodic updates and consequent broadcasts. The reason to avoid CSI metrics is that accurate CSI is even harder to estimate in dynamic networks, and periodic broadcasts would need to be very fast to guarantee accurate reaction to channel conditions. Moreover, relay selection faces several optimization problems that are difficult to solve, which means that the best relay may be difficult to find. Hence, for dynamic scenarios, the solution may be to make use of the best possible relaying opportunity even if not the optimal one (e.g., in terms of CSI). By achieving the best performance over the faced conditions, RelaySpot aims to target a fair balance between relay selection and additional resource blockage.

In summary, RelaySpot aims to select relay(s) based only on information local to potential relays, with minimum computational effort and overhead. The remaining of this section describes RelaySpot opportunistic relay selection, cooperative relay scheduling, and chain relaying mechanisms.

A. Opportunistic Relay Selection

The relay selection process only takes into account nodes that are able to successfully decode frames sent by a source. This ensures that potential relays are closely bounded with the source, with which they have good channel conditions. The qualification of a node as a relay depends upon local information related to node degree, load, mobility and history of transmissions to the specified destination, and not to CSI.

Node degree, estimated by overhearing the shared wireless medium, gives an indication about the probability of having successful relay transmissions: having information about the number of neighbors allows the minimization of the collision risk as well as blockage of resources. However, it is possible that nodes with low degree are overloaded due to local processing demands, leading to delay.

Equation 1 estimates the interference level that a potential relay is subjected to as a function of node degree and load. Let N be the number of neighbors of a potential relay, T_d and T_i the propagation time of direct and indirect transmissions involving such potential relay, respectively, and N_i and N_d the number of nodes involved in such indirect and direct transmissions (indirect transmissions are the ones overheard by the potential relay, and direct transmissions are the ones ending and starting at the potential relay). Adding to this, T_p is the time required for a potential relay to process the result of a direct transmission. The interference factor (I) affecting a potential relay has a minimum value of zero corresponding to the absence of direct or indirect transmissions.

$$I = \sum_{j=1}^{N_d} (T_{dj} + T_{pj}) + \sum_{k=1}^{N_i} T_{ik}, \quad I \in [0, \infty[\qquad (1)$$

The goal is to select as relay a node that has low interference factor, which means few neighbors (ensuring low blockage probability), short transmissions and few direct transmissions (ensuring low delays).

Figure 5 shows a scenario where node R is selected as a potential relay. Node $N1$ is the direct neighbor of node R, while there are several other indirect neighbors ($N2$,$N3$,$N4$, X). Apart from R, node X also seems to be a relay candidate due to its low interference level. But it may be difficult to select R or X due to the similar interference levels: while R has a short transmission from a neighbor and a long transmission from the source, X is involved in an inverse situation. The selection of R or X as a relay can be done based on two other metrics of the RelaySpot framework: history of successful transmissions towards destination; stability of potential relays.

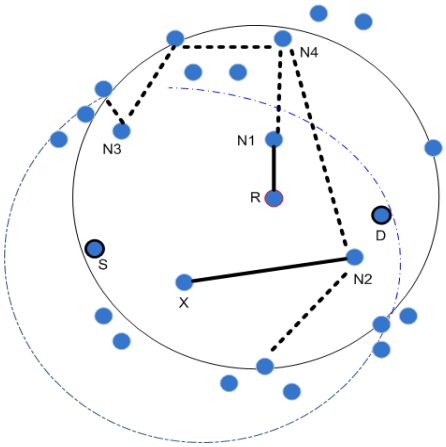


Figure 5. Opportunistic relay selection scenario

Although it is ensured that potential relays have good channel with the source, the quality of the relay-destination channel is unknown. Without performing measurement of CSI for the relay-destination channel, channel conditions are estimated based on the successful ratio of previous transmissions towards the destination (history factor) and the current stability of a potential relay (mobility factor). The history factor (H), is estimated as a ratio between an exponential moving average of the duration of successful transmissions and the maximum duration of any successful transmission (H_M), variable that is initiated to a time unit. The factor H aims to tell whether the intended relay has probabilistically a good channel with

the required destination, without the need to estimate and broadcast channel information.

The mobility factor (M) is estimated as a ratio between an exponential moving average of the pause time of the node and the maximum detected pause time (M_M), which is initiated to a time unit. The factor M aims to select more stable nodes as relays.

Based on the interference factor of a node, as well as its history and mobility factors, the probability of selecting a node as relay for a given destination is given by Equation 2, which shows that the selection factor (S) is proportional to the history of successful transmissions to the destination and the pause time, and inversely proportional to the interference level of the node.

$$S = \frac{H * M}{1 + I}, \quad S \in [0, 1[\tag{2}$$

Lets go back to Figure 5 to illustrate the usage of Equation 2. Lets assume that R is a node that moves frequently around the destination with a good history of successful transmissions. While X is a node with long pause times but that is new near the destination. In this case, Equation 2 may give preference to node R, although it presents a higher mobility factor than X.

After overhearing data frames or RTS towards a destination, a potential relay uses the estimated selection factor (S) to compute the size of its contention window (CW), between a predefined minimum and maximum values of CW_{min} and CW_{max}, as given by Equation 3.

$$CW = CW_{\text{min}} + (1 - S)\,(CW_{\text{max}} - CW_{\text{min}}) \tag{3}$$

From a group of nodes that present good channel conditions with the source, the opportunistic relay selection mechanism gives preference to nodes that have low degree, low load, good history of previous communication with the destination, as well as low mobility. In scenarios with highly mobile nodes, we expect opportunistic relay selection to behave better than source-based relay selection (e.g., CoopMAC), since with the latter communications can be disrupted with a probability proportional to the mobility of potential relays, and relays may not be available anymore after being selected by the source.

As illustrated in Figure 6 the selection mechanism may lead to the qualification of more than one relay (R1, R2, R3), each one with different values of S, leading to different sizes of CW (e.g., R3 transmits first). Selected relays will forward data towards the destination based on a cooperative relay scheduling mechanism.

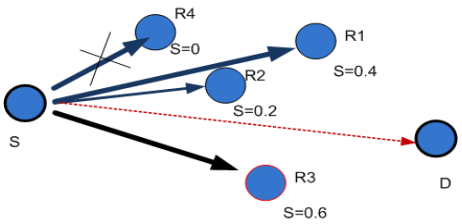


Figure 6. Opportunistic relay selection

B. Cooperative Relay Scheduling

This section describes the functionality proposed to allow self-elected relays to avoid high interference and to guarantee high data rates to a destination while preventing waste of network resources.

The contention window (computed in Equation 3) plays an important role in scheduling relay opportunities. The goal is to increase the probability of successful transmissions from relays to the destination by giving more priority to relays that are more closely bounded to the destination, while not neglecting the help that secondary relays may give. Increasing diversity, by allowing the destination to receive multiple copies of the same frame, aims to construct error free frames while avoiding re-transmissions.

Based on the quality of the frames received from all self-elected relays, the destination estimates which of the involved relays are more suitable to help in further transmissions (to get multiple copies the destination only process received frames after a predefined time window). By sending a list of priority relays embedded in ACK messages, the destination allows potential relays to improve the accuracy of the back-off time computation in next transmissions (relay with highest priority sends and the others back-off but keep overhearing the transmission). This functionality leads to a space-time diversity, which leverage the space diversity used by prior art (e.g., CoopMAC). Space-time diversity is achieved by allowing the usage of different relays over time, helping the same source-destination communication.

Figure 7 illustrates the cooperative relay scheduling, in a situation where R1, R2 and R3 are self-elected as relays, with R3 having smaller CW than R1 and R2 (as illustrated in Figure 6). If the destination receives good frames from multiple relays during a predefined time window, it decides for priorities (primary and secondary relays) on basis of SNR between well decoded frames. As an example, Figure 7, shows a situation where the destination is only able to decode the data by combining partial frames received from R1 and R2, in a scenario in which no data is received in good shape.

In this situation the destination sends an ACK having R1 and R2 as primary relays and R3 as secondary one i.e., ACK(R1, R2; R3). This means that in the next transmission R1 and R2 will transmit (diversity 2) and R3 will back-off and overhear the transmission.

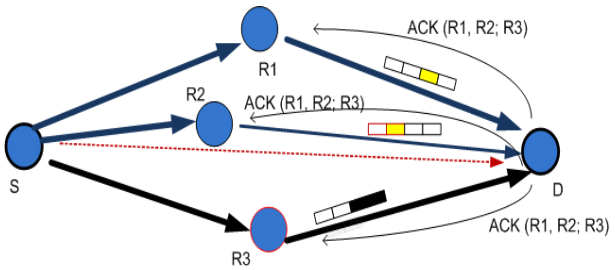


Figure 7. Cooperation relay scheduling

Cooperative scheduling allows to keep a source-destination transmission in a good shape even when the primary relay is

not useful anymore. Cooperation between selected relays (primary and secondary), identified by the priority list embedded in ACK message, aims to ensure a high probability of having the best set of relays over time. This means that based on current conditions, primary and secondary relays may switch their priorities.

Figure 8 illustrates the relay switching operation between a selected primary relay (R1) and secondary relay (R2): Destination chooses R1 as primary relay on basis of signal strength, while R2 is a secondary relay; in the next transmission R1 will transmit (diversity 1) and R2 will back-off. Suppose that after some time R1 move away and detects a deterioration of the conditions of the Source-R1 channel. In this situation R1 notifies the secondary relay (R2) with a Relay-Switch message. This means that R2 will become a primary relay, starting to transmit frames received from source.

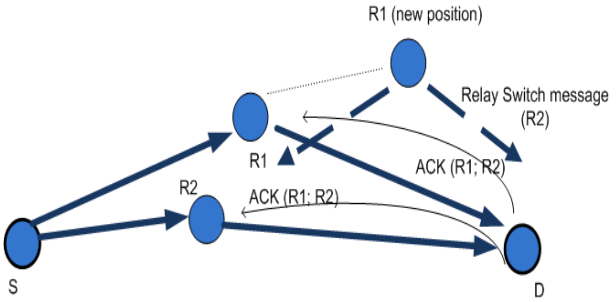


Figure 8. Cooperative relay switching

C. Chain relaying

The proposed opportunistic relay selection and cooperative relay scheduling mechanisms aim to increase throughput and reliability, as well as to reduce transmission delay by increasing the diversity adjusting the relaying order. Nevertheless, the presence of mobile nodes, as well as unstable wireless conditions, may require higher levels of diversity achieved based on nodes that are closed to the destination (higher probability of successful transmissions). Hence, RelaySpot includes the possibility of using recursive relay selection and retransmissions in case of poor performance. This functionality is called chain relaying (c.f. Figure 9). Nodes that are able to successfully decode MAC data frames sent by a relay to a destination may trigger the RelaySpot operation on that relay-destination channel in case the channel conditions are so bad that the node will overhear two consecutive NACK (or the absence of ACK's/ NACKs) during a predefined time window. This means that relays closer to the destination can help the transmission when the destination does not get any (acceptable) data frames from any relay in contact with the source.

With chain relaying, the relaying process is repeated for the relay-destination channel (R1-D and R2-D in Figure 9), by having another relay (R4) or set of relays helping the transmission from each of the previously selected relays to the destination. R4 may not receive correct frames from source, but it is closely bounded to R1 as well as to the destination. R4 can trigger chain relaying when both primary and secondary relays fail. Chain relaying aims to minimize the outage and to increase the overall throughput by complementing the cooperative scheduling functionality.

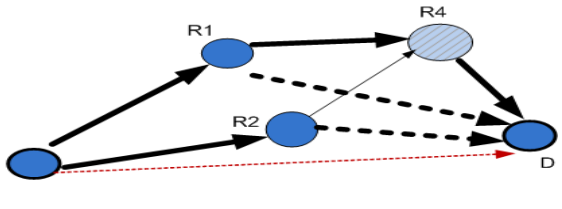


Figure 9. Chain relaying

V. RelaySpot Algorithm

RelaySpot is a hybrid relaying scheme, which means that it allows relays to retransmit data when: i) NACKs are overheard in the direct transmission; ii) relays detect that the performance of a direct link can be improved by relaying. RelaySpot operation, for a specific source-destination pair ends when there are no more packets to be send or when the destination informs the relays to stop relaying MAC data frames, after detecting a decrease in the number of damaged frames received through the direct channel below a predefined threshold. This action aims to increase network capacity by allowing relays to help other endangered transmissions.

RelaySpot operation has two modes: RelaySpot on potential relays and RelaySpot on destination (or gateway). Figure 10 shows the RelaySpot sequence of operations on potential relay nodes, which including the computation of the selection factor and relaying of MAC data frames.

Since the opportunistic relay selection process can lead to several relays being selected, self-elected relays may adjust their priority based on the information collected from the ACK sent by the destination. The primary relay (the one with highest priority) will continue sending frames, while other relays will back-off. Figure 10 shows that before sending data frames, the relay checks SNR for the signal received from source. If the SNR is below certain threshold (i.e., data rate is degraded) the relay stop participating as a relay by sending Relay-Switch message; otherwise it continue sending data frames until last frame. The primary relay then goes to back-off mode.

Figure 11 shows the RelaySpot operation at destination node. The destination keeps receiving good frames via relays until reception window expire. If the destination receives a good frame from a single relay it ACK with relay identification and send the frame to application to avoid further delays. However, if more than one good relay exists, the destination computes the priority list by using received SNR, acknowledging the priority list to self-elected relays. If there is not any good relay during reception window, the destination tries to combine the received partial frames. If the destination is able to decode the data by combining the received frames, it computes the priority list accordingly. However, if the destination is unable to decode the data even with combining, it sends NACK to indicate failure.

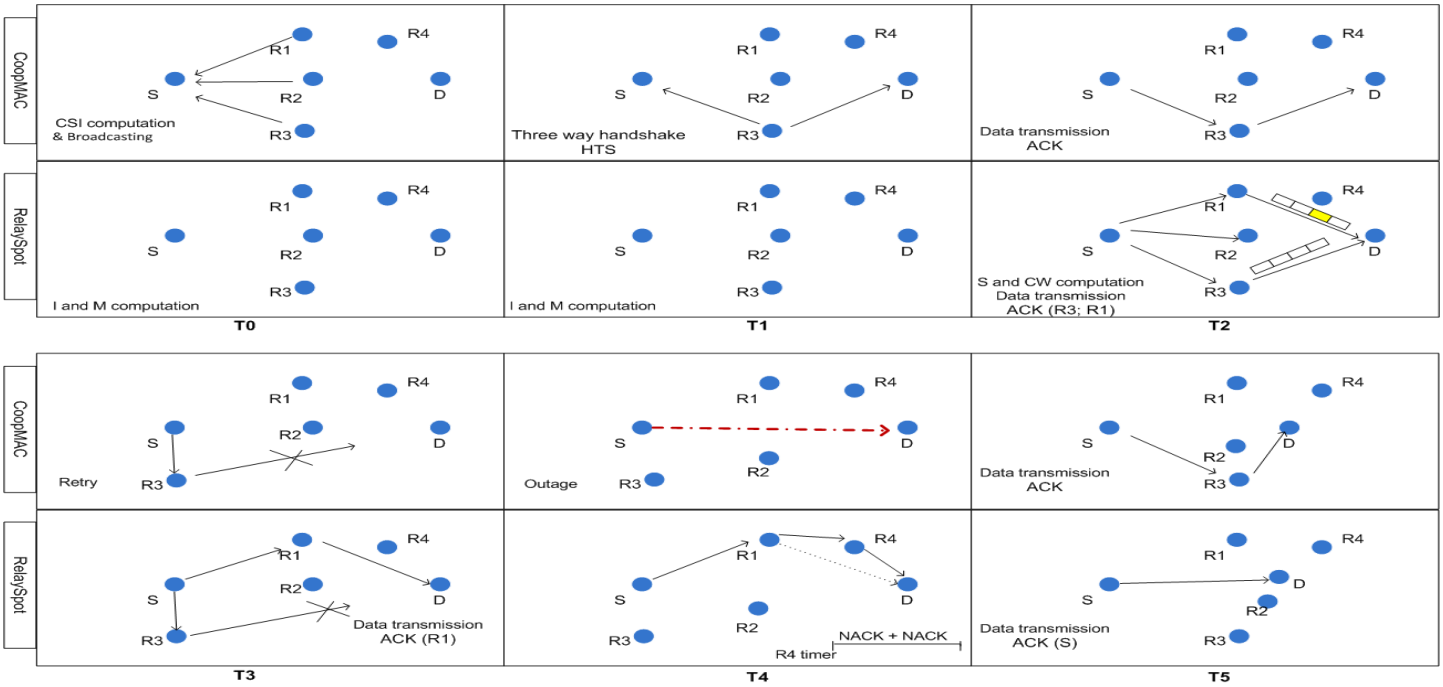


Figure 12. Illustration of the RelaySpot algorithm with chain relaying

VI. RelaySpot Vs CoopMAC Operational Comparison

Figure 12 illustrates the phases of the RelaySpot algorithm in comparison to CoopMAC. Lets consider that we have three potential relays (R1, R2, and R3), where R3 is the best (primary) relay. Figure 12 starts by showing that with CoopMAC at time T0 potential relays do some CSI computation and then broadcast it to source, while at that time RelaySpot potential relays do local computations of I and M factors without any transmission.

At time T1 CoopMAC relays undergo three way handshake using "Helper ready To Send" (HTS) messages, while RelaySpot potential relays updates local factors I and M without any transmission.

At time T2, CoopMAC sends data via the selected helper i.e., R3. RelaySpot potential relays first computes the selection factor S and CW after the reception of data from source, selecting R3 and R1 as relays, which then transmit data to the destination, achieving higher diversity than CoopMAC. The destination notifies the relays (in ACK message), about the priority order for future transmission i.e., ACK(R3; R1). After receiving the ACK, R1 backs-off since R3 seems to be suitable to provide reliable transmissions.

At time T3, R3, the primary relay, moves away. In such case CoopMAC repeats the complete relay selection procedure after a maximum number of retries. While in RelaySpot, the secondary relay R1 (in this example) tries to help the transmission and ends up sending data to destination on behalf of source, after detecting the missing ACK for R3 transmission (or detecting NACK). If this is successful, destination sends ACK(R1).

At time T4 we suppose that R1 is unable to cooperate. In this situation R4 overhears two consecutive NACKs during a predefined time frame. Thus chain relaying will occur as other nodes (R1, R2, and R3) are not suitable anymore. In case of CoopMAC, when there is no suitable relays, poor direct transmission takes place leading to outage.

At time T5 the destination moves closer to source and the direct link between source and destination becomes stronger. In RelaySpot when the destination starts receiving the correct frames from source, it notifies the relays to stop cooperation (i.e., ACK(s)) and continues receiving the direct data, while in CoopMAC the data will be still relayed over the selected relay (R3 in this example).

From this comparison it is clear that CoopMAC always uses additional control messages, such as periodic broadcast and HTS for handshaking. While RelaySpot does not have an overhead related to additional control messages. CoopMAC uses one relay only, while in RelaySpot multiple relays can be utilized in parallel or in sequence base on quality of received frames. CoopMAC does the CSI computation for relay selection, which incurs complexity; moreover the decision for relay is based on historic information. RelaySpot on the other hand, have fast reaction to network dynamics.

VII. Implementation and Analysis

In this section we start by describing the relaying protocol implementation in OMNET++, and then we discuss the simulation results: first we describe the initial analysis of RelaySpot, which serve as a reference point for further investigation; then we analyze the impact that interference has on relay performance. We also describe the analysis of the proposed cooperative relay switching.

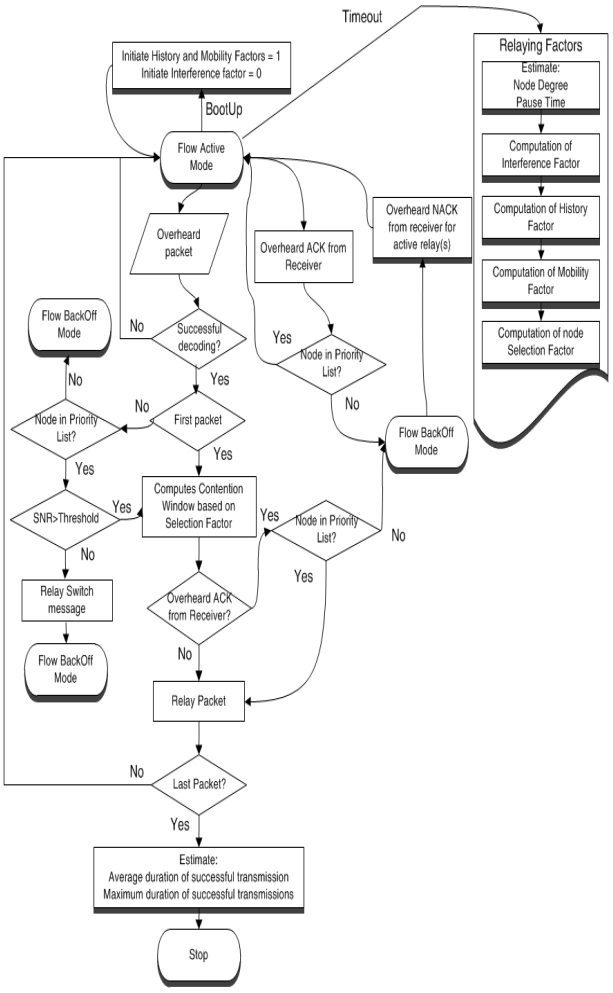


Figure 10. RelaySpot sequence of operations on potential relay nodes

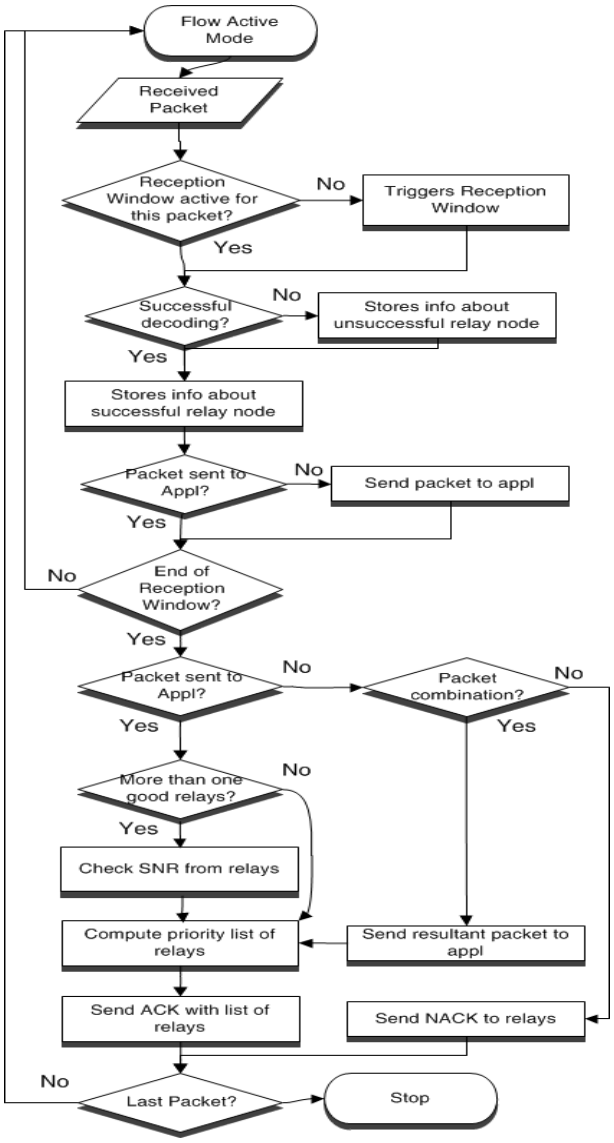


Figure 11. RelaySpot sequence of operations on destination nodes

A. RelaySpot implementation

In this section we describe the steps to implement the RelaySpot protocol, which is serving as a prototype for further implementation. We use OMNET++ 4.1 simulator and the MiXim 2.1 framework. As discussed before, relaying protocol is a MAC layer protocol. Therefore, most of the modifications were done in MAC layer. In MiXim framework, whenever a message arrives from physical layer (i.e., data is overheard), the MAC layer invokes a function "handleLowerMsg()". This function analyzed the incoming message and passes the message to either msgForMe() or msgNotForMe(); if the message (overheard frame) is not for the node, it invokes msgNotForMe(). Normally a node discards data frames that are not intended for itself, but we modified this method to allow a node to keep and send data frames (to behave as a relay). Similarly, when a message arrives from upper layer (i.e., application layer), "handleUpperMsg()" is invoked. This function analyze the message and if it is a data frame to send, the node sends channel sense request and schedule the *Contention* timer. If the node wins contention it invokes sendDATAframe() to send down the data to physical layer, after which it set the MAC state to Wait For ACK (WFACK).

We have added an additional timer "*RelayContention*", which is used to schedule the contention period for the relay and to send down the channel sense request. The modifications to msgNotForMe() are as follows: when an overheard message arrives, first a node checks if it is a data frame or not. Then it checks the reservation duration (NAV timer) with message arrival time to be sure that it can relay the data frame. To start relaying, first a node adds its MAC address into address4 field of the data frame; it cancels the NAV timer as the node cannot send or receive data until NAV expires; then it schedules RelayContention timer and sends down a channel sense request. If there is no other ongoing transmission on channel, it calls sendDATAframe() to send the received frame to destination with necessary modification.

B. Initial Analysis

We start by performing an analysis to test the general relaying framework, in order to setup the performance reference points in what concerns throughput and latency in a scenario

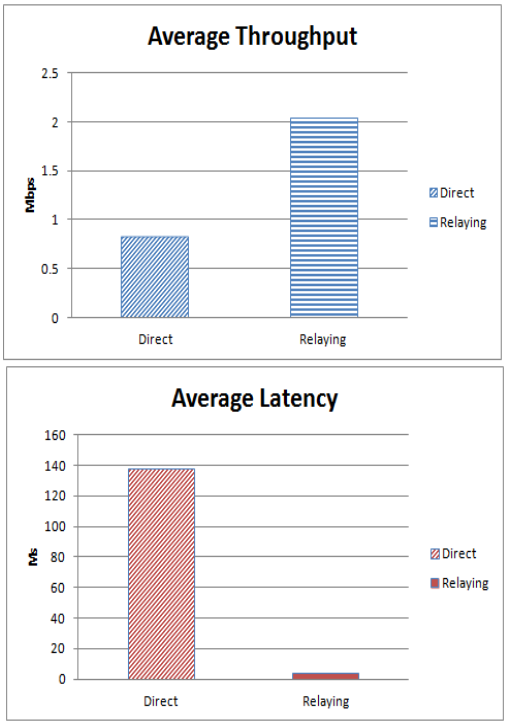


Figure 13. Throughput and latency gains of relaying

without interference, in which data frames can have different sizes. We also run simulations for a first evaluation of the impact of interference on relaying.

In order to test RelaySpot we create a scenario where source and destination is placed at a distance of more than 150 meter with a direct link of 1Mb. Figure 13 shows that when one relay is used improvements can be achieved in term of throughput and latency, reaching an average throughput near 2.1 Mbps and insignificant latency. In the same scenario the direct transmission provides only an average throughput of 0.82 Mbps, which is close to the average capacity of the direct link, and an average latency of 137.8 ms. The improvement in throughput and latency, illustrated in Figure 13, refers to a scenario that is free of interference. However the introduction of interference (different direct and indirect traffic) is expected to lead to a degradation of performance (we analyze this later on this article).

We also analyze the impact of frame size over gain in throughput. Figure 14 shows that RelaySpot has a gain in throughput for a frame of size of 1 Kbits or more in relation to the direct transmission. The gain is negative when the size of frame is less than 1 Kbit, however such frame size is rarely used. The frame size strongly influences the throughput as for smaller frame size the throughput drops due to the domination of the transmission overhead.

In order to have a first glimpse about the impact of interference over relayed data, we run a set of simulations with 25 nodes (other than relay) randomly generating between 1 and 10 Mbps of traffic (inducing indirect interference). Figure 15 shows that the throughput of relayed data dropped to a maximum of 1.8 Mbps instead of 2.1 Mbps as shown in Figure 13. In this situation the interfering node is in competition with the relay node. Therefore the throughput gain depends upon transmission opportunities. Figure 15 shows that at interference (traffic at interfering node) up to 2 Mbps the relay throughput drop linearly while throughput at interfering node reaches to its maximum. Further increase in interference (application traffic of interfering node) does not increase the throughput of interfering node because the relay is blocking this node. This benefits the relay throughput.

A node, when operating as a relay also has an impact on the system: on the relay node itself and on neighbor transmissions. Hence we also analyzed the impact that relaying data has on the data generated and consumed by the node acting as relay. Figure 16 shows that due to interference the number of frames dropped at the relay node increases significantly. Hence, by avoiding interference we can improve not only the performance of the flow being relayed, but also of the overall network performance. This motivates a further analysis about the impact that direct and indirect interference have on relaying based on RelaySpot, which uses interference-aware relay selection metrics.

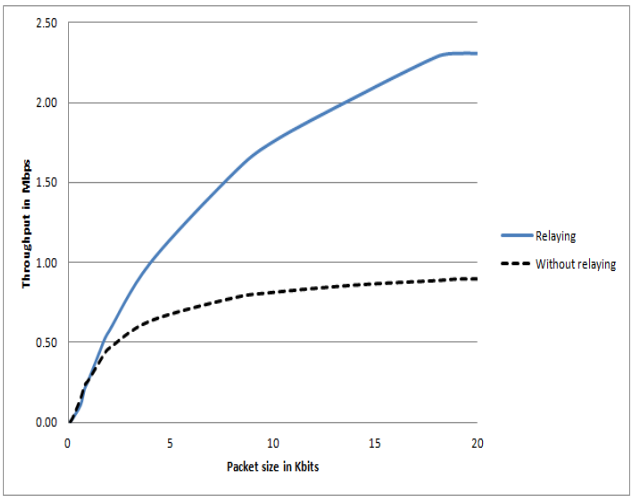


Figure 14. Frame size impact on throughput (with and without relaying)

C. Analysis of Impact of Interference

In this section we evaluate the performance of RelaySpot in the presence of different levels of direct and indirect interference. Several simulations are run based on the MiXim framework of the OMNET++ 4.1 simulator. Each simulation has a duration of 300 seconds and is run ten times, providing a 95% confidence interval for the results.

Simulations consider a scenario where all nodes are static and have similar stochastic history of transmissions among them, thus the mobility factor and history factors are assumed to be 1. The source and destination are at a distance of more than 150 meters from each other with a poor direct link, with an average of 1 Mbps. Depending on the level of interference needed in each simulation, potential relays may operate also as sources sending data to the same destination at different traffic rates.

First we did simulations by selecting a relay based on node degree and distance towards the destination in an interference

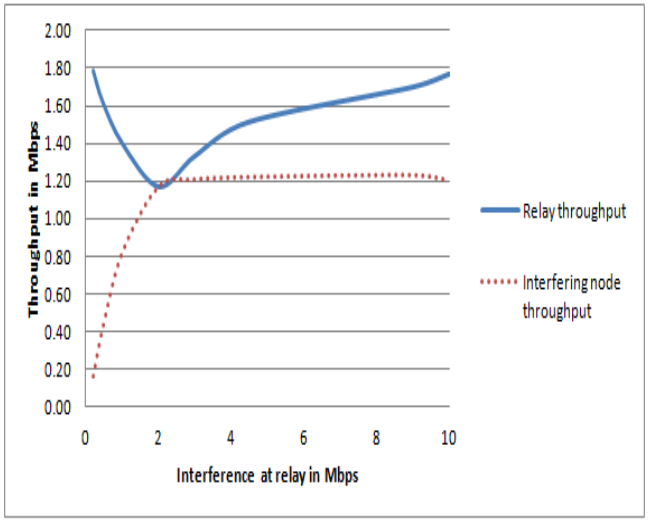


Figure 15. Relay throughput with indirect interference

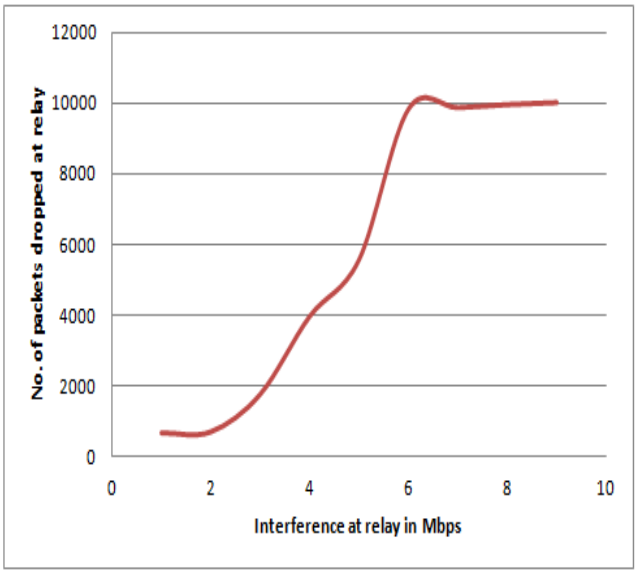


Figure 16. Number of frames dropped at relay node

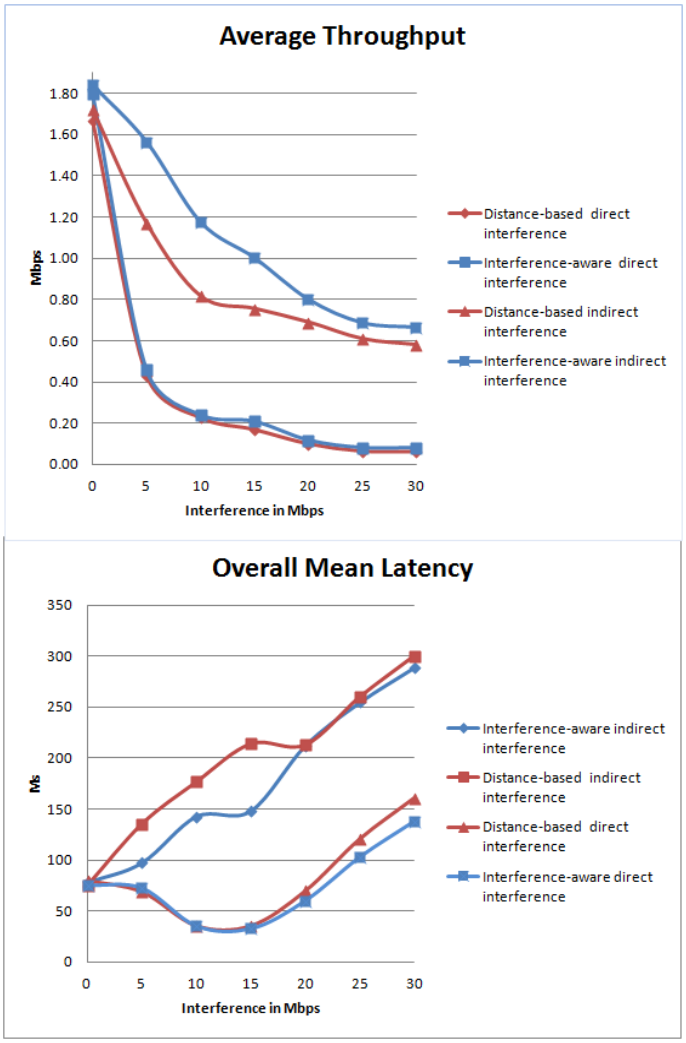


Figure 17. Throughput and latency analysis [23]

free scenario. Using node degree as metric lead to the selection of isolated nodes, with high probability, being such nodes far away from the source and destination which were reflected in low throughput and big latency. Therefore, distance-based relay selection achieved significant improvement in term of both throughput and latency when compared to degree-based. Therefore, we consider distance-based as a reference point for our further evaluation of RelaySpot as an interference-aware relaying algorithm.

Figure 17 shows that by introducing interference, the performance of degree-based solutions starts degrading. As the direct interference increases, the relay starts blocking the source-destination communication, since it has its own processing delay. By using the proposed RelaySpot metric (Interference-aware) for direct interference, we achieved improvements in term of throughput and latency, as RelaySpot selects a relay which has less load. However, the gain is not considerable, as the direct interference at other potential relay nodes is still affecting the source-destination pair. Therefore, it is always better to chose a dedicated relay (a relay without its own traffic). Figure 17 shows that the gain of the interference-aware approach with direct interference is more visible in the case of latency, since the interference-aware approach selects a relay from a set of nodes that present higher availability for retransmission (lower number of local generated traffic), even if placed further away from the destination, leading to lower latency. This gain is clearer with high traffic load, since distance-based approach keep selecting overloaded nodes near the destination.

In a scenario with indirect interference, the throughput gain of RelaySpot is significant (e.g., 33% with a load of 10 concurrent flows) because the indirect traffic does not affect the source-destination pair: only the chosen relay is affected. Nevertheless RelaySpot is able to choose a relay with low probability of being blocked by additional transmissions, leading to an improvement in performance. With an increase of traffic load this performance gain diminishes, because at some level of indirect interference it is hard to avoid interference, but it is always higher than the distance-based approach. The

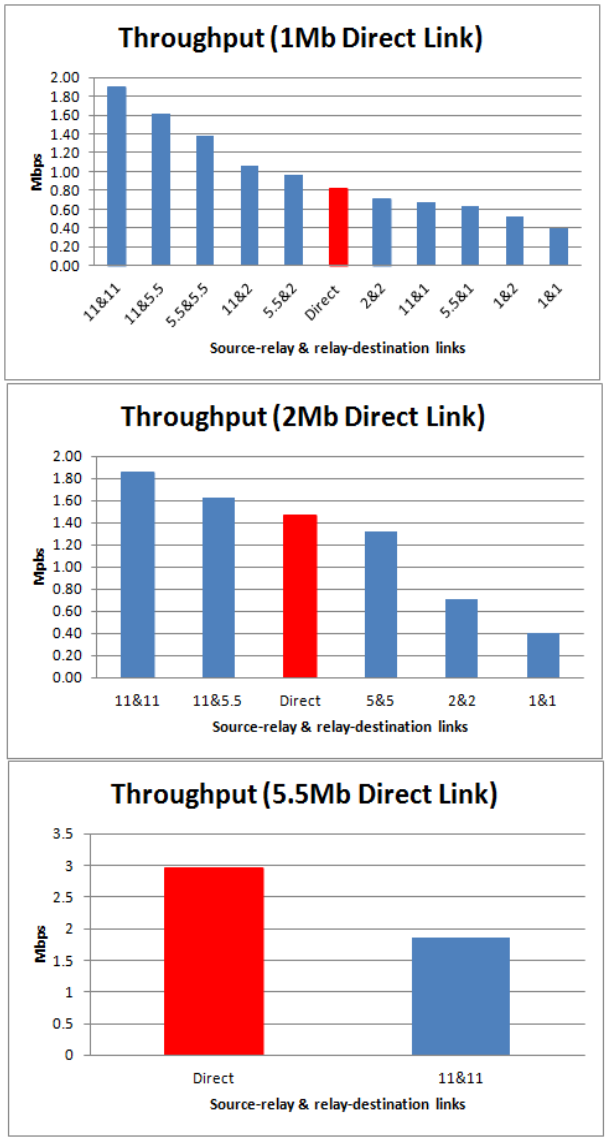


Figure 18. Analysis with different data rates

advantage of our interference-aware approach is also visible in terms of latency, since selecting a relay with low interference (lower number of concurrent neighbor flows) leads to higher transmission opportunities. The gain in latency decreases with a load of 20 concurrent flows, mainly due to increasing number of concurrent flows placed far away from the destination, which benefits distance-based approaches. Nevertheless, results show that even with a random placement of concurrent flows, the interference-aware solution keeps a lower latency with high traffic loads. By including the history factor we can achieve further improvement in both throughput and latency [24].

D. Cooperative Switch Analysis

In what concern the switching between relays as discussed in section IV-B, a relay can give up relaying if it does not ensure acceptable conditions anymore. To analyze this idea we run simulations with different source-destination pairs, relayed by relays in different location, and with different combination of data rates in the source-relay and relay-destination links. It is observed from Figure 18 that relaying is not always useful. In order to achieve performance improvement the direct link must be replaced by relays with both source-relay and relay-destination links that present a data rate higher than the direct link, and one of the links must have a data rate at least twice higher than the direct link. For instance, 1 Mbps direct links can be replaced by relays with 11 Mbps and 5.5 Mbps, or even with 5.5 Mbps and 2 Mbps, but not with 2 Mbps and 2 Mbps. For example if a direct link of 5.5 Mbps is replaced by relays with 11-11 Mbps links, the gain will be negative.

Hence, to ensure performance gain the cooperative relay switching operation of RelaySpot provides the following operation, as illustrated in Figure 10: in MiXim the data rates are decided on bases of received SNRs. Therefore, the primary relay collects SNRs of different links by overhearing to decide if switching is required or not. If a primary relay observe that its signal strength (SNR value) is below certain threshold (which means it does not support fast bit rate anymore), it notifies the secondary relay for help with a Relay-Switch message. The secondary relay starts relaying data while the primary relay goes to back-off mode.

VIII. Conclusions and Future Work

Most of the current cooperative relaying approaches use only one relay, selected based on CSI estimations, without exploiting different relays in parallel or in sequence. The proposed *RelaySpot* framework provides a set of functional building blocks aiming to opportunistically exploit the usage of several relays to ensure accurate and fast relay selection, posing minimum overhead and reducing the dependency upon CSI estimations in scenarios with mobile nodes. The proposed building blocks are related to opportunistic relay selection, cooperative relay scheduling, and chain relaying. Moreover, RelaySpot does not have any additional control overhead and its functional blocks allow fast reactions to network conditions. We also observed that interference have great impact over relay network. After analyzing the behavior of RelaySpot in a scenario with interference our findings show that selecting a relay with low interference (lower number of concurrent neighbor flows) leads to higher transmission opportunities. The impact of direct and indirect interference is different in relation to throughput and latency: indirect interference has higher impact over latency, while direct interference leads to lower throughput. Interference-aware solution as RelaySpot ensures also low resource blockage.

As a future work, we will analyze the performance of a version of RelaySpot that would be aware of the type of traffic in order to further investigate the behavior of the cooperative relay switching and relay scheduling functionalities. We will also further evaluate how RelaySpot can contribute to increase the overall network capability in the presence of mobile nodes.

Acknowledgment

Thanks are due to FCT for PhD grant number SFRH/BD/60436/2009. The research leading to these results

has received funding from the European Commission's Seventh Framework Programme (FP7) under grant agreement n° 257418, project ULOOP (User-centric Wireless Local Loop).

REFERENCES

[1] T. Jamal, P. Mendes, and A. Zúquete, "RelaySpot: A framework for Opportunistic Cooperative Relaying," in *Proc. of IARIA ACCESS*, Luxembourg, June 2011.

[2] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. Panwar, "CoopMAC: A Cooperative MAC for Wireless LANs," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 2, pp. 340–354, Feb. 2007.

[3] W. Elmenreich, N. Marchenko, H. Adam, C. Hofbauer, G. Brandner, C. Bettstetter, and M. Huemer, "Building Blocks of Cooperative Relaying in Wireless Systems," *Electrical and Computer Engineering, Springer*, vol. 125, no. 10, pp. 353–359, Oct. 2008.

[4] T. M. Cover and A. E. Gamal, "Capacity Theorems for the Relay Channel," *IEEE Trans. Info. Theory*, vol. IT-25, p. 57284, Sep. 1979.

[5] A. Sendonari, E. Erkip, and B. Aazhang, "User Cooperation Diversity-Part II: Implementation Aspects and Performance Analysis," *IEEE Journal on Selected Areas in Communications*, vol. 51, no. 11, pp. 1939–1948, Nov. 2003.

[6] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior," *IEEE Trans. Inform. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.

[7] E. Erkip, A. Sendonaris, A. Stefanov, and B. Aazhang, "Cooperative Communication in Wireless Systems," *Advances in Network Information Theory*, vol. DIMACS Series, 2004.

[8] M. Gast, "802.11: Wireless Networks: The Definitive Guide," *O'Reilly*, Apr. 2002.

[9] Z. Hao and C. Guohong, "rDCF: A Relay-Enabled Medium Access Control Protocol for Wireless Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 5, Mar. 2006.

[10] K. Tan, Z. Wan, H. Zhu, and J. Andrian, "CODE: Cooperative Medium Access for Multirate Wireless Ad Hoc Network," in *Proc. of IEEE SECON*, California, USA, Jun. 2007.

[11] L. Mei-Hsuan, S. Peter, and C. Tsuhan, "Design, Implementation and Evaluation of an Efficient Opportunistic Retransmission Protocol," in *Proc. Of IEEE MobiCom*, Beijing, China, Apr. 2009.

[12] H. S. Lichte, S. Valentin, H. Karl, I. Aad, L. Loyola, and J. Widmer, "Design and Evaluation of a Routing-Informed Cooperative MAC Protocol for Ad Hoc Networks," in *Proc. of IEEE INFOCOM*, Phoenix, USA, Apr. 2008.

[13] H. Adam, W. Elmenreich, C. Bettstetter, and S. M. Senouci, "CoRe-MAC: A MAC-Protocol for Cooperative Relaying in Wireless Networks," in *Proc. of IEEE GLOBECOM*, Honolulu, Hawaii, Dec. 2009.

[14] N. Marchenko, E. Yanmaz, H. Adam, and C. Bettstetter, "Selecting a Spatially Efficient Cooperative Relay," in *Proc. of IEEE GLOBECOM*, Honolulu, USA, Dec 2009.

[15] A. Bletsas, A. Khisti, D. Reed, and A. Lippman, "A simple Cooperative Diversity Method Based on Network Path Selection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 659–672, Mar. 2006.

[16] K.-S. Hwang and Y.-C. Ko, "An Efficient Relay Selection Algorithm for Cooperative Networks," in *Proc. of IEEE VTC*, Baltimore, USA, Oct. 2007.

[17] Y. Chen, G. Yu, P. Qiu, and Z. Zhang, "Power-Aware Cooperative Relay Selection Strategies in Wireless Ad Hoc Networks," in *Proc. of IEEE PIMRC*, Helsinki, Finland, Sep. 2006.

[18] Y. Zhao, R. Adve, and T. J. Lim, "Improving amplifying-and-forward relay networks: Optimal power allocation versus selection," in *Proc. of IEEE ISIT*, Seattle, USA, Jul 2006.

[19] H. Adam, C. Bettstetter, and S. M. Senouci, "Adaptive Relay Selection in Cooperative Wireless Networks," in *Proc. of IEEE PIMRC*, Cannes, France, Sep. 2008.

[20] Z. Lin, E. Erkip, and A. Stefanov, "Cooperative Regions and Partner Choice in Coded Cooperative Systems," *IEEE Transactions on Communications*, vol. 54, no. 7, pp. 1323–1334, Jul. 2006.

[21] A. Nosratinia and T. E. Hunter, "Grouping and Partner Selection in Cooperative Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 2, Feb. 2007.

[22] T. Jamal and P. Mendes, "Relay Selection Approaches for Wireless Cooperative Networks," in *Proc. of IEEE WiMob*, Niagara Falls, Canada, Oct. 2010.

[23] T. Jamal, P. Mendes, and A. Zúquete, "Interference-Aware Opportunistic Relay Selection," in *Proc. of ACM CoNEXT student workshop*, Tokyo, Japan, Dec 2011.

[24] T. Jamal, P. Mendes, and A. Zúquete, "Opportunistic Relay Selection for Wireless Cooperative Network," in *Proc. of IEEE IFIP NTMS*, Istanbul, Turkey, May 2012.

Community Telephone Networks in Africa

Bridging the gap between poverty and technology

Curtis Sahd
Department of Computer Science
Rhodes University
Grahamstown, South Africa
curtissahd@gmail.com

Hannah Thinyane
Department of Computer Science
Rhodes University
Grahamstown, South Africa
h.thinyane@ru.ac.za

***Abstract*—Many new cell phones on the market come with 802.11 enabled, along with standard Bluetooth functionality. A large percentage of working class people in South Africa typically cannot afford 802.11 enabled cell phones, and thus the most applicable form of wireless data transfer is achieved through the Bluetooth protocol. This paper investigates bridging Bluetooth and 802.11 protocols on low cost wireless routers equipped with a Broadcom chip and a Universal Serial Bus port, as well as bridging on high end cell phones. For the router component of this research, the BlueZ protocol stack will be implemented on top of the OpenWrt platform and experiments relating to the feasibility and scalability of Session Initiation Protocol voice calls between clients on the Bluetooth network and clients on the wireless mesh network will be investigated. For the cell phone component of this bridging, Java Mobile will be used as the development platform of choice, and a comparison between bridging on the cell phone and on the wireless router will be conducted, with metrics such as latency, scalability, and minimum throughput will be considered. This paper also investigates Bluetooth throughput achieved at varying distances, as well as the relationship between the average time and the average expected time with variations in the transmission unit size. This paper provides an overview of the Mobile Media Application Programming Interface, its shortcomings, and how to overcome them. This paper proposes a low cost solution to building community telephone networks in rural South Africa, through the bridging of 802.11 and Bluetooth interfaces.**

Keywords – Wireless; SIP; Community telephone networks; BlueZ; Community polling systems.

I. INTRODUCTION

The Bluetooth protocol has been around since 1994, and its primary function is to replace wires and serve as lightweight wireless implementation for data transfer. Even though most high end cell phones are equipped with 802.11, Bluetooth still serves as the primary data transfer protocol between cell phones in South Africa. Based on a survey conducted on the streets of Grahamstown, South Africa, it was discovered that most people called someone in Grahamstown or in the surrounding region on a daily basis. Currently, the only way to make phone calls, whether local or inter-town, is to make use of a fixed landline, which the vast majority of the underprivileged do not have access to, or to make use of the ever increasingly expensive mobile service providers. Paying sky high cellular network rates to make local phone calls places an enormous burden on already financially constrained rural communities. Bluetooth alone cannot be used in a full scale implementation which would enable free local phone calls. However, the combination of Bluetooth and 802.11 mesh networks could, in the context of South African rural communities, create a system which saves rural communities millions of Rand each year. A worldwide study of 24 000 participants from 35 markets (undertaken in November 2009) showed that 86% of respondents owned a mobile phone, while only 55% owned a desktop computer. Of those who owned a mobile phone, 55% used it for media purposes, and 42% used it for transferring files and made use of the Bluetooth connection [22]. Even though these statistics show the trends of people in urban areas, people in rural areas, according to the survey conducted, will most certainly make use of these technologies to reduce communication costs.

Wireless mesh networks (WMNs) are a crucial component to this research as they bridge the gap between the Bluetooth access points and provide the throughput necessary for handling all the calls/traffic. WMNs are dynamic, self-configuring networks which are designed to span large geographical areas. WMNs could therefore be used to span the geographical area of the rural community, and possibly even connect remote rural communities to one another.

This paper aims to explore inexpensive means to creating low cost community telephone networks with existing technology in rural areas. We propose a system which enables the seamless integration of Bluetooth and 802.11 on the OpenWrt and Java Mobile (JavaME) platforms. We begin with an introduction to Bluetooth and in particular, Bluetooth networking with Piconets and Scatternets. We then investigate the throughput achieved by the Bluetooth protocol at varying distances between two class two devices, followed by a comparison between the average time and the average expected time for Bluetooth transmissions with varying transmission unit sizes. We then provide a brief overview of the OpenWrt platform and focus on mesh networking, as well as reviewing related work in this area. Section IV then describes the Mesh Potato, and the possibilities it presents in rural areas. Section V introduces

the Mobile Media API (MMAPI) as well as the concept of double buffering. Section VI then provides an in depth analysis of the proposed infrastructure of the Blue Bridge, and the associated advantages and disadvantages of various implementations. Section VI also provides a high level understanding of the components and techniques as well as the challenges and constraints involved in transmitting voice data from one device to another. Section VII then describes the context of this paper and how the proposed technology can be beneficial to rural communities and coincides with objectives of various social reconstruction programmes. Section VIII describes possible future works on this concept, as well as the challenges and constraints involved in implementing these extensions. Section VIII then concludes this paper.

II. BLUETOOTH PERSONAL AREA NETWORKS

A. Overview

At initial conception Bluetooth was considered the future of Personal Area Networks (PANs), due to it being a lightweight protocol and the inexpensive manufacturing of Bluetooth chips [2]. The Bluetooth specification clearly defines PANs and associated roles of the nodes in the PAN in the case where two devices are communicating directly. The Bluetooth specification also defines the roles of nodes in multi-hop environments, but less research has been conducted in this field [2]. Asthana and Kalofonos [3] have developed a custom protocol which enables the seamless communication of existing Piconets within a Scatternet. Specifically, their research allows for the creation of Ethernet and IP local links on top of scatternets through the use of a standard PAN profile implementation, without the need for ad hoc forwarding protocols [3].

Plenty of research has been done in the field of providing Internet Access to rural communities. There has been little to no research in the field of making use of low cost hardware infrastructure to bridge Bluetooth and 802.11 which enables large scale service provision to local and remote rural communities. Bluetooth Piconets and Scatternets are an important component of bridging Bluetooth and 802.11 mesh networks, as in some cases devices will be able to communicate directly with one another (Piconets) and in other cases devices may only be able to communicate by sending traffic through a number of other nodes before reaching the desired node (more applicable to Scatternets). With that said, many researchers have investigated the formation and limitations of Mobile Ad-hoc Networks (MANETs) with the Bluetooth protocol [2].

B. Piconets and Scatternets

According to Bisdikian [4] a piconet is simply defined as a collection of Bluetooth devices which can communicate with one another. A Piconet consists of one master node and one or more slave nodes, and exists for as long as the master communicates with the slaves. Piconets are formed in an ad-hoc manner, and need a minimum of one master node and maximum of seven active slave nodes. Although only seven active nodes are able to transmit based on coordination of the master node, other nodes are able to connect to the Piconet, and are said to be in a parked state [4].

Scatternets are based on Piconets and are said to exist when one device is a member of multiple Piconets. In the case of Scatternets, a node can only serve as the master node for one Piconet.

For the purposes of this research it is important to understand the functioning of Piconets and Scatternets in order to handle the association of clients with Bluetooth access points.

As mentioned above, Piconets consist of a maximum of eight active nodes, consisting of the master node, and seven active slave nodes. In the context of this research as well as the broader context of community telephone networks, the participation of a maximum of seven active clients poses a very real problem in terms of scalability as well as feasibility.

Apart from the issue with scalability, another concern is the throughput achieved by the Bluetooth protocol at varying distances. Subsection C provides an overview of the relationship between throughput and distance for class two Bluetooth devices.

C. Bluetooth throughput at distance

A major consideration for the successful implementation of Blue Bridge is the throughput achieved at varying distances between access points and clients. Although the maximum theoretical transmission distance specified by the Bluetooth specification for class two Bluetooth devices is 10m, Sahd conducted experiments which resulted in an increase in the maximum transmission distance. Sahd [13] conducted throughput tests at 1m intervals until such point was reached where the file transfer was unsuccessful. A 6.6 MB audio file was transmitted four times with an MTU of 668 bytes, and the results averaged to achieve the results depicted in Figure 1. Sahd [13] also found that reliable transmission was feasible for distances up to 15m with class two Bluetooth devices. Figure 1 highlights the results obtained from distance variation during class two Bluetooth transmission:

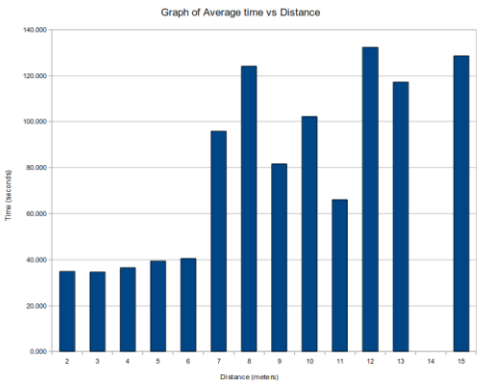


Figure 1. Graph of average time vs distance

It can clearly be seen that the maximum distance achieved by the class two Bluetooth transmitter in the Nokia N95 8GB is capable of achieving distances of up to 15m. There are no results for the transfer at the 14m mark, as link disconnection was prevalent. This could be as a result of electrical piping and/or other environmental factors which adversely affected the connection clarity at the 14m mark. An interesting observation is the throughput achieved at the 11m distance interval, which exceeded that achieved at the 7m, 8m and 9m distance intervals. The result set is theoretically inconsistent, and atmospheric and/or environmental conditions could be the only possible explanation for these differences in throughput. With an understanding of the relationship between distance and throughput for class two Bluetooth devices, Subsection D investigates throughput optimization by determining the optimum range of transmission unit sizes.

D. Relationship between transmission unit size and throughput

JavaME enables variation of the Bluetooth transmission unit, which proved to greatly influence the transfer speed, and hence the transmission distance and quality of the transmission. Although larger transmission unit sizes logically achieve the fastest transfer speed and the quality of voice/media playback, this is however not the case with audio streaming on Java Mobile devices. Sahd performed an experiment which tested the transfer speed and quality of playback by reducing the transmission unit size from 668 bytes to 67 bytes (10% decrements). This experiment shows the difference between the expected time and the average expected time, thus enabling a throughput comparison between the various transmission unit sizes. This experiment also shows the differences between theoretical Bluetooth throughput and actual Bluetooth throughput. The experiment was conducted by sending a 6.6 MB audio file from a Nokia N95 8GB to a Nokia N82. The following formula was used in calculating the average expected time:

x = time taken to transfer file using the maximum MTU
n = percentage of the maximum MTU
Expected time = (x * (100% - n%)) + x

Figure 2 shows the relationship between the average time and the average expected time for each transmission unit size while transferring a 6.6 MB file between two cell phones using Bluetooth:

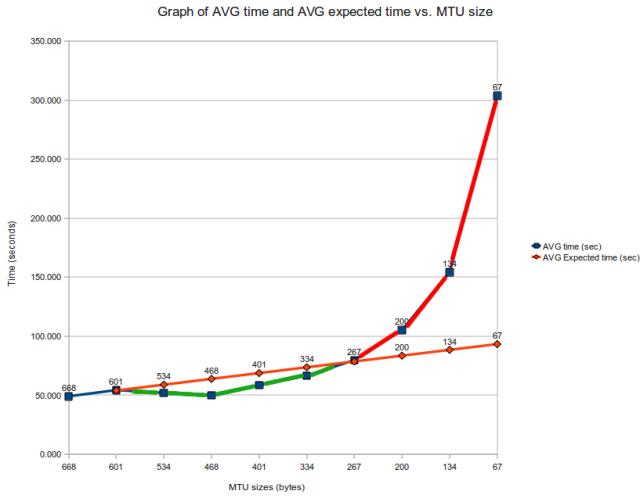


Figure 2. Graph of average time and average expected time vs. MTU size

Figure 2 shows the average time vs the average expected time for a given transmission unit size. The green portion of the graph represents the portion where the average time is less than the average expected time. The orange portion of the graph represents the case where the difference between the average time and the average expected time is greater than or equal to 0 seconds, and less than or equal to 2 seconds. The red portion of the graph represents the case where average expected time is less than the average time. From Figure 2 it can be seen that average time was less than the average expected time for transmission units 534 through 334. The average expected time however is less than the average time for transmission units 200 through 67. The two most obvious points on the graph are those of 601 and 267 where the average time is equivalent to the average expected time. With that said, the transmission units 601 and 267 are ideal for testing Bluetooth throughput which closely matches theoretical throughputs.

Section II introduced the Bluetooth protocol; the formation of PANs as well as their limitations; and the optimization of the Bluetooth protocol by means of distance and MTU variation. Section III introduces the OpenWrt platform and how it can be used in the context of mesh networking and service provision.

III. OPENWRT

OpenWrt is defined as Linux for embedded devices [5]. OpenWrt provides a plethora of opportunities for robust application development and service provision on embedded devices, and for the purposes of this research, specifically on wireless routers. In order to grasp the functioning of OpenWrt it is necessary to understand the various components of the software which manages wireless routers and for that matter any embedded device. The software which runs on computer chips or on embedded device chips is known as firmware [6]. The following are a few of the many types of chips which have firmware installed on them: read-only memory (ROM); programmable read-only memory (PROM); erasable programmable read-only

memory (EPROM). PROMs and EPROMs are designed to allow firmware updates through a software update [6]. In order to compile custom Linux firmwares on embedded devices, a technique known as cross compiling is used, where a new compiler is produced, which is capable of generating code for a particular platform, and this compiler is then able to compile a linux distribution customized for a particular device [7]. Generally, the cross-compiling process begins with a binary copy of a compiler and basic libraries, rather than the daunting task of creating a compiler from scratch [7]. The remainder of this section describes mesh networking principles and practices on the OpenWrt platform, as well as the state of the art in rural mesh networks.

OpenWrt contains a number of packages which assist with the implementation of mesh networks. Optimized Link State Routing Protocol (OLSR) is an example of a routing protocol developed by Andreas Tønnesen which has been implemented in the form of a package for OpenWrt [8]. Another Open Source mesh networking implementation known as ROBIN (ROuting Batman Inside) has been developed on top of OpenWrt Kamikaze [9]. ROBIN is self-configuring and self-maintaining, which enables the seamless creation of wireless mesh networks. ROBIN requires a minimum of one Digital Subscriber Line (DSL) connection, a Dynamic Host Configuration Protocol (DHCP) enabled router which is connected to the DSL line and serves as the gateway node [5]. Client repeater nodes simply have to be powered on and a mesh network is dynamically configured [5]. With that said, open mesh networking protocols, which simplify the creation and extension of mesh networks, can be utilized in rural communities. Mesh networks thus serve as a low cost alternative to information technology service provision in rural areas, providing significantly more benefits than drawbacks. The benefits of mesh networks in rural communities have been extensively discussed [9] [10]. Reguart et al. [9] suggest that mesh networking technologies in urban areas are often unsuited to rural areas due to the high cost of equipment and maintenance. They proposed and tested Wireless Distribution System (WDS) by making use inexpensive wireless hardware (Linksys WRT54AG and Linksys WRT54G). Through a prototype deployment of their infrastructure they found that inexpensive wireless equipment is capable of providing fourty people with internet access, and at any one point in time there are between fifteen and thirty active clients [9]. The aforementioned implementation performs surprisingly well for sparsely situated rural communities, but would not suffice for the purposes of South African rural communities for the following reasons: Rural communities in South Africa are densely populated; laptops are seldom found in rural areas, as most of the people are living below the breadline and cannot afford such equipment; even if everyone had access to laptops, the use of inexpensive wireless equipment as used above would be overloaded and the end result would most likely be malfunction; also the use of secured outdoor equipment is imperative in the context of South Africa due to crime levels.

This Section provides an introduction to the OpenWrt platform, its versatility, and the benefits it provides in service provision. Section IV introduces the Mesh Potato, and how it utilizes the OpenWrt firmware.

IV. THE MESH POTATO

The Mesh Potato is a new device which merges the ideas of current telephony (analog phones) and future technology (reliable wireless communications). The Mesh Potato combines a wireless access point (AP) with an Analog Telephony Adapter (ATA), and thus enables cheap communications using existing technology [18]. Routers by Meraki [19] and OpenMesh [20] are gaining popularity due to their low cost and robustness, but they however lack the functionality contained within the Mesh Potato in terms of integration with existing telephonic infrastructure.

Although rural areas in South Africa are often on the outskirts of town, plenty of remote and isolated settlements exist, and more often than not, these settlements lack infrastructure such as running water, sewage and waste removal, and electricity. In such cases where electricity is scarce or non-existent, the Mesh Potato is ideal since it can be powered by a 10w solar panel [18].

The Mesh Potato is powered by Open Source firmware (Linux, OpenWrt, B.A.T.M.A.N and Asterisk) which removes vendor lock in and makes the Mesh Potato cost effective and highly configurable [18]. The Mesh Potato enables the seamless connection of analog telephones, as well as wired and wireless IP phones. Cellular technology is the primary form of communication in rural areas in South Africa, and although analog phones are inexpensive and could be subsidized by the government, the Mesh Potato is currently unable to cater for the existing needs of people in rural areas.

This section shows how OpenWrt can be used in service provisioning scenarios. Section V introduces the Mobile Media API (MMAPI); highlights its benefits in the realm of media streaming; and describes its shortcomings and possible solutions in overcoming them.

V. THE MOBILE MEDIA API (MMAPI)

The Mobile Media API (MMAPI) is an optional API which enables advanced multimedia capabilities on the Java enabled devices [25]. The MMAPI enables the playback of different audio and video formats from the network, from a record store (persistent storage on JavaME devices), from a JAR file, or from dynamic buffers. The MMAPI also enables audio and video capturing, as well as streaming on Java enabled devices.

One of the major problems with streaming on the JavaME platform is the lack of RTP and RTSP support. One of the more implemented methods of streaming live audio (voice transmissions) is the use of the SIP protocol for signaling; the RTP protocol for transmitted the media; and the RTSP protocol for describing the media being transmitted with the RTP protocol. Although the JavaME

platform supports SIP communication, the lack of the other two aforementioned protocols renders streaming somewhat more complicated. With that said other methods of describing the media and transmitting it have to be devised.

There are three main components of the MMAPI: The Manager; the Player; and the DataSource. The Manager class is responsible for Player instantiations, which in turn sources the data from the DataSource, thus enabling playback. The Manager class essentially bridges the gap between the Player and the DataSource. Figure 3 shows the MMAPI architecture:

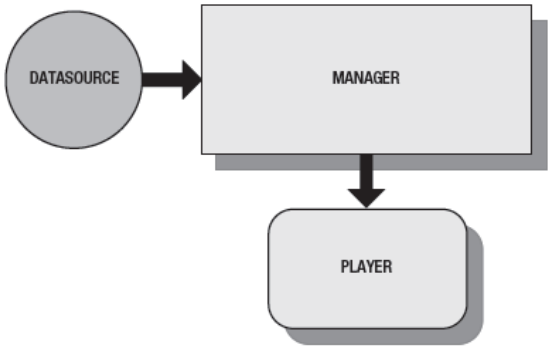


Figure 3. MMAPI Architecture

Without the MMAPI streaming on the JavaME platform would be impossible without the development of other APIs or frameworks which make it possible. The lack of widespread support for RTP and RTSP on the JavaME platform, changes the streaming from the traditional structured and proven method, to one consisting of: record; compress and serialize; and transmit. The Player class contained within the MMAPI is capable of playing media from various sources, one of which includes network streaming. The MMAPI specification however does not mention that streaming is only possible once the entire media file (audio or video) has been downloaded. According to Costello [26], streaming is defined as the process of transferring data from source to destination, where the destination device decodes the data before all the data has been transferred. With that said the MMAPI not only lacks support for streaming protocols such as RTP and RTSP, but there are also limitations with the way in which the Player class needs the entire media stream before being able to render any media playback.

Vazquez-Briseno and Vincent [27] propose a media streaming architecture consisting of a streaming server; a multicast proxy; and a mobile client. Their architecture utilizes the RTP; RTSP; and Session Description Protocol (SDP) protocols to stream an AMR audio file from the streaming server to the mobile client. Their client made use of the MMAPI, and more specifically, the Player class which played back the audio before the entire media file had been downloaded/streamed. The technique which they used to accomplish this partial media playback, is known as double buffering.

As such we applied the same double buffering technique to overcome the limitations of the Player class in the MMAPI. Table I shows the double buffering technique involved in streaming media on the JavaME platform:

TABLE I. AUDIO STREAMING WITH DOUBLE BUFFERING TECHNIQUE

Time	Buffer 1	Buffer 2	Player 1	Player 2
T_1	Receiving			
T_2		Receiving	Play Buffer 1	
T_3	Receiving			Play Buffer 2
T_4		Receiving	Play Buffer 1	

From Table I it can be seen that the double buffering technique starts with buffer 1 receiving part of the media stream at T_1. Once there are enough bytes to begin playback (normally around 150 KB), buffer 2 then receives the next part of the media stream, and player 1 then plays the bytes contained in buffer 1. At T_3 the bytes in buffer 1 are then replaced by the next part of the media stream, and the bytes contained in buffer 2 are then played using player 2. This process of alternating buffers and players then continues until the end of the media stream is reached.

Even though the double buffering technique is the only possible way of streaming on the JavaME platform short of developing RTSP and RTP streaming functionality at the application level, this technique has its fair share of disadvantages. One of the most obvious disadvantages is the jitter encountered during playback between the alternating players. The time it takes for the mobile phone to switch from one player to the other is negligible, but enough to cause a slight delay in handing over playback to the other player.

This section introduced the MMAPI and its role in media streaming. This section also showed how the MMAPI can be adapted in order to cater for the lack of support for certain protocols on the JavaME platform. Section VI describes the proposed infrastructure for bridging the 802.11 and Bluetooth protocols by means of the JavaME and OpenWrt platforms. Section VI also investigates streaming on the JavaME platform, and highlights the challenges and constraints, as well as possible solutions. Section VI also provides a costing analysis of the proposed infrastructure, as well as a means of funding.

VI. PROPOSED BRIDGING INFRASTRUCTURE

After extensive literature reviews we found that there is a lack of knowledge in the field of Bluetooth and 802.11

bridging in the context of rural communities in Africa, and as such we propose a system (Blue Bridge) which not only deals with remote access to such communities, but also enables service provision through the use of inexpensive and readily available technology thus connecting the unconnected. The system will be centered around the OpenWrt firmware, which is to be installed on the Ubiquiti AirRouter [11]. The AirRouter will not only serve as an interface for 802.11 connections, but will also become a Bluetooth access point through the use of the BlueZ protocol stack which controls the functioning of the Bluetooth dongle inserted into the USB port of the AirRouter. Asterisk [12] will be installed as a package on the OpenWrt platform, and will serve as the SIP controller. A package will be developed for the OpenWrt platform which will bridge the connections between the 802.11 and Bluetooth interfaces. Figure 4 shows the proposed infrastructure involving one AirRouter:

Figure 4. Proposed OpenWrt infrastructure for low cost community telephone network.

Any cell phone on the Bluetooth interface of the Blue Bridge should be able to place SIP calls to any other phone on the Bluetooth interface, as well as to any phone on the 802.11 interface. Of course as mentioned in Section B a maximum of seven active connections can exist on the Bluetooth interface, which clearly places limitations on the scalability of the proposed system.

With the aforementioned, the components of the proposed system include the AirRouter (running OpenWrt); the USB Bluetooth dongle; and a JavaME enabled cell phone, which the majority of the surveyed population possesses. The aim of this research is to provide Bluetooth access (via the connected Bluetooth USB dongle) as well as 802.11 access to multiple geographically dispersed routers which in turn enables the creation of community telephone networks, thus connecting the unconnected, and significantly decreasing the burden of expensive cellular calls.

The ideal scenario is the use of minimal equipment, while still maintaining an acceptable level of service provision. This translates to decent quality voice calls, with minimal downtime. In order to achieve this, an understanding of the Bluetooth protocol and its scalability limitations is vitally important. Sahd [13] conducted a study which investigated the real throughput achieved by the Bluetooth protocol on mobile devices. Sahd [13] found that the average transfer speed of the Logical Link Control and Adaptation Protocol (L2CAP) when transferring a 6.6 MB M4A audio file twice between two cell phones is 136.39 KBps [13]. If a maximum of seven clients are connected to the Bluetooth interface each client would be allocated a bandwidth of 19.48 KBps. Based on the assumption that seven simultaneous connections are active on the Bluetooth interface, the minimum accumulated bandwidth for these connections is 27.35 KBps, which would allow a theoretical number of thirty five clients to be connected [14].

This research will also investigate the differences in performance of Blue Bridge implementations on the JavaME platform and on the OpenWrt platform. Of course the most prominent difference between implementations on the two platforms is the class of Bluetooth device. The OpenWrt platform implementation of the Blue Bridge will make use of a class one Bluetooth device which is capable of a distance of 100m, whereas cell phones typically contain class two Bluetooth chips which enables transmission at distances of 10m. There are three classes of Bluetooth of which class three achieves a distance of up to 1m [23]. Sahd [13] found that even though the Bluetooth specification states a distance of 10m, transmission is possible at distances as high as 15m. Figure 5 shows the proposed Blue Bridge infrastructure on the cell phone:

Figure 5. Mobile phone based implementation for community telephone networks.

From Figure 5 it can be seen that an external asterisk server would have to substitute the asterisk server contained within the OpenWrt packages. The scalability of the internal asterisk server would have to be researched and compared to that of the external asterisk server. On the other hand, the Nokia N95 8GB could pose to be a severe bottleneck under load.

In order to determine which platform will serve as the basis for a community telephone network, a number of metrics would have to be compared. These metrics can be seen in Table II:

TABLE II. KNOWN METRICS OF PROPOSED BLUE BRIDGE PLATFORMS

Metrics	OpenWrt	Blue Bridge on cell phone
Cost	Cheap	More expensive
Compactness	Average	Very compact
Complexity	High	Medium
Platform	Linux	Java Mobile

Based on the information currently available, assumptions from the data in Table II could lead one to believe that the Blue Bridge on the cell phone would be the better alternative as a whole. However, metrics such as performance under load, scalability, and multi-hop capability can only be determined once the implementation and necessary research has been completed.

Even though Figures 4 and 5 depict the bridging infrastructure necessary for connecting previously disconnected Bluetooth devices, one crucial component is that of interconnecting the bridging nodes at varying distances. Ideally, the interconnection of these nodes should be extendable from one rural community to another, possibly for distances above 50km.

With the above overview of the equipment needed for the implementation of the Blue Bridge, Subsection A provides information pertaining to the techniques used to transmit and receive audio transmissions. Subsection B highlights the costs involved, and a means for funding the Blue Bridge.

A. *JavaME streaming infrastructure*

As mentioned in Section V., streaming on the JavaME platform involves the adaptation of existing classes and APIs. This section deals with the techniques involved in successful implementation of the client side application for this research.

Due to JavaME supporting the SIP protocol, and its trivial implementation on this platform, SIP has been omitted from the diagrams and can be assumed present. Figure 6 shows the interaction between two mobile phones participating in an audio call, and the procedures involved in ensuring its success:

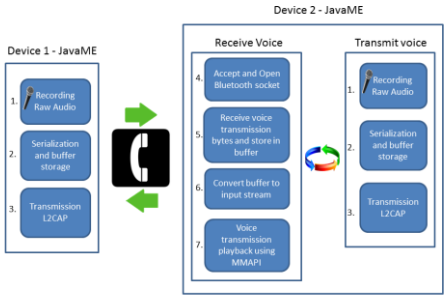


Figure 6. Sequence of events for mobile phone based implementation of community telephone networks.

As mentioned in Section V., the MMAPI lacks support for the major streaming protocols on the JavaME platform. From Figure 6, it can be seen that Device 1 initiates the call, and the procedures necessary in doing so are: Recording the audio via the MMAPI into a temporary buffer, from which the audio data is streamed to Device 2. In order for Device 2 to accept the voice call and receive the subsequent audio transmission, an *accept and open* method is continually running, ready for any incoming transmissions. With the inability of the Player class in the MMAPI to play partially downloaded media, it is necessary to store the incoming media in a temporary buffer, and play back the audio from a static buffer. A static buffer in this context is a buffer which contains media content which does not change, thus fooling the Player class into thinking that the media contained in the buffer is in actual fact an entire media file which is not being streamed. As described in Section V., once the media contained in the static buffer is rendered/played back, the media from the temporary buffer is then transferred to the static buffer, thus enabling constant media playback.

Of course throughout this process of receiving the streamed audio, storing it in buffers, and playing it back, Device 2 might also want to transmit audio back to Device 1. With that said, this then gives rise to an even greater problem, which is intrinsic to the Bluetooth protocol itself, and that is the fact that transmission using the L2CAP protocol is half-duplex. This translates to only one device/party being able to transmit at a time, essentially creating a push-to-talk system which can by no means be considered a phone call.

Clearly, when compared to full-duplex communication, which is that of traditional phone calls, this system is limited. However in a country such as South Africa where poverty is so widespread, the benefits of being able to communicate for free, far outweigh the limitations of half-duplex communication. In the context of instant messaging and community polling, half-duplex communication is completely acceptable.

B. *Costs and implementation considerations*

There are two important factors to consider when determining the cost, and the number of units necessary for the implementation of community telephone networks: the geographical area and the proposed number of connected clients. The geographical area plays a large role in determining the strength of the devices needed to transmit a good quality signal. Mountains, trees, buildings, and other obstructions have to be considered. The number of connected clients dictates the scalability of the system, and thus the overall cost of implementation. Table III provides an overview of the costs involved:

TABLE III. KNOWN METRICS OF PROPOSED BLUE BRIDGE PLATFORMS

Device	Cost	Means of funding
AirRouter 150Mbps WiFi Router	R313.50	Government
Mecer Class 1 USB Bluetooth (ENUBT-C1EM)	R169.00	Government

Device	Cost	Means of funding
Basic machine for Asterisk server (1.8 GHz, 2GB RAM, 500GB HD)	R2700.00	Government

Based on the costs in Table III, the maximum total cost for a prototype system catering for seven connected nodes will come to a total of R3182.50. This value is of course inclusive of the Asterisk server machine, which would not be necessary if the Asterisk server were to be implemented on the AirRouter itself.

The average voice call from the Vodacom cellular network to another network costs R2.75 per minute [21]. Based on the assumption that seven people spend five minutes on the phone each day for one month, the total cost incurred is R2983.75. Even though the Bluetooth protocol only permits seven active clients, more than seven people could connect to one AirRouter, due to the unlikeliness of everyone placing calls simultaneously. With that said, it can be seen that in just one month, the costs incurred by impoverished communities can be drastically reduced. This rate is the highest rate per minute rate on the Vodacom prepaid plan, and was chosen to estimate the maximum amount of money spent on cell phone calls.

Section VII provides an overview of government initiatives to introduce equality in impoverished areas, as such all equipment and implementation costs would be government subsidized.

VII. CONTEXT

The reconstruction and development program (RDP) of South Africa is a program implemented by the African National Congress (ANC) which addresses socioeconomic problems which exist as a result of the Apartheid regime [15]. The RDP program is of great benefit to all South Africans and in particular, South Africans living in rural areas without basic necessities such as adequate housing, water and electricity. Traditionally RDP housing was built on plots of 250m^2 which placed tremendous strain on the fair land distribution due to special constraints [16]. Recently, there has been a movement from traditional RDP housing to more cost effective multi-storey RDP housing which reduces plot sizes from 250m^2 to 80m^2 [16][17]. With that said this poses as an ideal situation for the successful implementation of the Blue Bridge, as signal penetration will be higher and this type of RDP housing would prove more effective from a point of view of device mounting as well as line of sight access for surrounding residents. The Blue Bridge will benefit such communities immensely in terms of cost savings, and possible expansions could include educational resources and Internet access.

Section VIII concludes this paper and provides possible extensions to this research.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we proposed an inexpensive means to creating a community telephone network, which utilizes existing technology and infrastructure. We demonstrated an innovative approach to merging two independent technologies to achieve maximum penetration in all spheres of society. We proposed an infrastructure for the implementation of the Blue Bridge on the OpenWrt platform, as well as on the JME platform, and determined the metrics necessary for large scale implementation. This paper demonstrated an understanding of the social inequality and the effects of overpriced communications on impoverished communities.

The Mesh Potato lacks functionality which caters for the existing needs of people in rural areas. Similarly, the Blue Bridge lacks the functionality of providing an analog telephony interface, which is still widely used. As such, future work which adds functionality to the OpenWrt component of the Blue Bridge could involve connecting the Mesh Potato to the AirRouter via cable, and ensuring that both devices are on the same subnet, thus enabling the utilization of the analog interface of the Mesh Potato. In terms of the cell phone component of the Blue Bridge, the Mesh Potato could be connected to the cell phone via the wireless interface.

Another proposal for future work regarding this research could involve the use of low cost, high powered wireless equipment which could solve the need for large numbers of AirRouters or similar devices, since one device could provide access to a larger area. Future implementations of the aforementioned could involve connecting powerful wireless equipment to the AirRouter via the LAN interface, and in the case of the cell phone based Blue Bridge, via the wireless interface. The proposed expansion of the original infrastructure can be seen in Figure 7:

Figure 7. Proposed wireless expansion of OpenWrt based Blue Bridge.

Figure 7 shows the expansion of the OpenWrt based infrastructure through the use of an external high powered wireless device, which is connected to the AirRouter via cable. This device then expands the wireless network, which then enables a larger number of clients to connect to the mesh and reap the benefits of a community telephone network. Of course the AirRouter will still serve as an access point for nearby 802.11 and Bluetooth clients.

Figure 8. Proposed wireless expansion of cell phone based Blue Bridge.

Figure 8 depicts the expansion of the cell phone based infrastructure for the Blue Bridge. Since the cell phone is unable to connect to the external wireless device via LAN cable, a connection needs to be made wirelessly. As such, the external high powered wireless device will transmit signal over a greater distance accomplished by the cell phone and will serve as the primary AP for 802.11 based clients.

Blue Bridge has a plethora of benefits in the context of South Africa and its diverse demographic spread. Not only does this research provide a platform for cheap local calls, but it also gives birth to instant messaging applications, community polling systems, and possibly even crime reporting and emergency response.

An instant messaging expansion of Blue Bridge not only provides an alternative to the proposed voice communication, but also provides a means to conserve precious bandwidth. With the introduction of instant messaging capability to Blue Bridge, other possibilities arise, such as text-based community polling systems and social networks.

Although community polling systems have been researched in the context of e-voting and similar applications, the use of such systems (and their implementation) as a means for service delivery are largely unexplored. With somewhat limited resources and often overburdened government departments, the reporting and subsequent implementation and follow-up of services proves to be a very real problem in South Africa. Future uses for this research could implement community polling systems as a means to report, attend to, and track progress of issues relating to refuse removal, road maintenance, security risks and other vital services which the community depends upon. Members of the community will use existing cell phones to access the community network through a series of Bluetooth and 802.11 access points, thus enabling people with less modern cell phones to connect to the community network, post new service tickets and view current service tickets. Any community member wanting to participate in community polling system will register with their ID number through the web-browser on their mobile device, thus eliminating potential "prank service requests". Another means of reducing the frequency of illegitimate requests and hence the overall burden on the system is to only allow people above the age of 21 to register.

Community polling systems are a great example of how Blue Bridge can be used to alleviate pressure on government departments, and essentially provide the community with a means of ensuring that municipalities are aware of issues in the surrounding area. Such systems also provide a more managed approach for government departments to track open service tickets as well as a means of informing the community of pending changes and/or service disruptions.

According to Kumar and Sinha [24], e-Government is defined as the set of technological tools which are used in enhancing the functioning of government to better serve its citizens. With the wide scale adoption of Blue Bridge on mobile devices, government officials and citizens could improve the efficiency of municipalities. One of the major concerns in the implementation of such systems in a country such as South Africa where the literacy rate pales in comparison to first world countries, is of course the inability to use text-based systems. In such cases, service requests would have to be tracked by means of pictorial interfaces.

ACKNOWLEDGMENT

I would like to thank my sponsors, the Telkom Centre of Excellence at Rhodes University, funded by Telkom SA, Business Connexion, Verso Technologies, THRIP, Stortech, Tellabs and the National Research Foundation.

REFERENCES

[1] Sahd, C., Thinyane, H., "Connecting the Unconnected," The Fourth International Conference on Advances in Mesh Networks, Mesh 2011.

[2] Zaruba, G.V., Basagni, S., and Chlamtac, I., "Bluetrees-Scatternet formation to enable Bluetooth-based ad hoc networks," IEEE International Conference on Communications, ICC 2001, vol. 1, 2001, pp. 273-277.

[3] Asthana, S. and Kalofonos, D., "Secure ad-hoc group collaboration over bluetooth scatternets," Applications and Services in Wireless Networks, 2004. ASWN 2004. 2004 4th Workshop, pp. 199-124.

[4] Bisdikian, C., "An overview of the Bluetooth wireless technology," IEEE Communications Magazine, vol. 39, 2001, pp. 86–94.

[5] OpenWrt. Available at: http://openwrt.org, 2011. [Accessed 04-04-2011].

[6] Apple. What is Firmware?. Available at: http://support.apple.com/kb/ht1471, 2008. [Accessed 01-04-2011].

[7] Fainelli, F., "The OpenWrt embedded development framework," 2008.

[8] OpenWrt. Available at: http://wiki.openwrt.org/inbox/mesh.olsr, 2011. [Accessed 06-04-2011].

[9] Reguart, A., Cano, J.C., Calafate, C.T., and Manzoni, P., "Providing Internet Access in Rural Areas: A Practical Case Based on Wireless Networks," The 2006 IFIP WG 6.9 Workshop on Wireless Communications and Information Technology in Developing Countries (WCIT 2006), 20-25 August 2006, Santiago, Chile.

[10] Parikh, T.S., and Lazowska, E.D., "an architecture for delivering mobile information services to the rural developing

world," Proceedings of the 15th international conference on World Wide Web, 2006, pp. 791-800.

[11] UBNT. Available at: http://ubnt.com, 2011. [Accessed 04-04-2011].

[12] Asterisk. Available at: http://asterisk.org, 2011. [Accessed 07-04-2011].

[13] Sahd, C. (2010). "Bluetooth Audio and Video Streaming on the J2ME Platform." Unpublished master's thesis, Rhodes University, Grahamstown, South Africa.

[14] AsteriskGuru. Available at: http://www.asteriskguru.com/tools/bandwidth_calculator.php, 2011. [Accessed 09-04-2011].

[15] Metagora. Reconstruction and Development Programme (RDP) of South Africa. Available at: http://www.metagora.org/training/encyclopedia/rdp.html, 2006. [Accessed 10-04-2011].

[16] Alexandra. Another RDP first from the Alexandra Renewal Project. Available at: http://www.alexandra.co.za/05_housing/article_0610_rdp_housing.htm, 2006. [Accessed 09-04-2011].

[17] Joshco. Sol Plaatje. Available at: http://www.joshco.co.za/solplaatje.html, 2011. [Accessed 10-04-2011].

[18] VillageTelco. Mesh Potato. Available at: http://www.villagetelco.org/mesh-potato/, 2011. [Accessed 11-04-2011].

[19] Meraki. Meraki. Available at: http://meraki.com/, 2011. [Accessed 11-04-2011].

[20] Open-Mesh. Open-Mesh. Available at: http://www.open-mesh.com/, 2011. [Accessed 11-04-2011].

[21] Vodacom. 4U Prepaid. Available at: http://www.vodacom.co.za/vodacom/Deals/Prepaid/Prepaid+Price+Plans/4U+Prepaid, 2011. [Accessed 23-05-2011].

[22] Curtis, S. Global telecoms insights 2010 focus report. Available at: http://www.tnsglobal.com, 2010. [Accessed 08-04-2011].

[23] NOKIA. Bluetooth technology overview. Available at: www.forum.nokia.com, April 2003. [Accessed 09-04-2011].

[24] Kumar, M. and Sinha, O., "M-Government – Mobile Technology for e-Government," Proceedings of the 5th international conference on e-governance. ICEG 2007. Hyderabad, India, 2007.

[25] Goyal, V., "Pro Java ME MMAPI: Mobile Media API for Java Micro Edition," Apress, 2006.

[26] Costello, S. "What is streaming?." Available at: http://ipod.about.com/od/glossary/g/streamingdef.htm, 2010. [Accessed 09-04-2011].

[27] Vazquez-Briseno, M., and Vincent, P. "An Adaptable Architecture for Mobile Streaming Applications," IJCSNS 7, 9 (2007), 79.

Performance Isolation Issues in Network Virtualization in Xen

Blazej Adamczyk, Andrzej Chydzinski
Institute of Informatics
Silesian University of Technology
44-100 Gliwice, Poland
{*blazej.adamczyk,andrzej.chydzinski*}*@polsl.pl*

***Abstract*—Resource virtualization has been known and used for a while as a mean of better hardware utilization and cost reduction. Recently, the idea of virtualization of networking resources has become of vital importance to networking community. Among other things, this is connected with the fact that the virtulization principle is built in many discussed Future Internet (*FI*) architectures. In this study we deal with the virtualization of networking resources offered by Xen virtual machine monitor. We are especially interested in the performance isolation across virtual network adapters. Firstly, we demonstrate several problems with the performance isolation. In particular, the results of a number of experiments in which the activity of one virtual machine influences the network performance of any other are presented. We also examine the fairness, predictability and configurability of the network I/O scheduler in Xen. Secondly, we propose solutions to the problems revealed by our experiments. In particular, we introduce prioritization into Xen Netback driver, add a verification mechanism to the output buffer and discuss possibilities of some other improvements.**

***Keywords*-performance isolation; Xen; virtualization; network scheduler.**

I. Introduction

The increasing number of different IT services are making the virtualization idea a very important aspect of computer science. Virtual Machine Monitors (VMMs) bring about the dynamic resource allocation and enable full utilization even of the most powerful servers, while still maintaining good fault isolation between virtual machines (VMs, also called domains in Xen). However, the services provided over the network may require a certain quality, which is not easy to ensure in a virtualized environment. Several VMs can share the same physical network interface as well as other hardware (processor, memory etc.) what likely makes one VM affect other VMs' performance. Therefore, the *performance isolation* is crucial in case of some applications and has to be carefully verified.

In this paper, we focus on Xen VMM, [2], which is one of the most popular virtualization platforms and an Open Source project. Firstly, we present a study of the network performance isolation between Xen virtual machines. Different test scenarios allowed us to identify several problems. Secondly, we carefully analyze the Xen CPU scheduler and the network I/O scheduler to find out their possible source and resolution method.

The motivation behind our study is the fact that virtualization of networking resources has recently became of vital importance to networking community. This is partly connected with the fact that the virtualization principle is built in many projects dealing with propositions on the Future Internet architecture, for example in 4WARD FP7, [3], FIA MANA, [4], AKARI, [5], PASSIVE FP7, [6], GENI, [7], IIP, [8]. As observed in [9], creating high-level abstractions of networking resources, that cover the underlying physical infrastructure and implementation may help to overcome several drawbacks of the current Internet architecture. For instance, virtualization allows coexistence of multiple networking technologies in the network layer and offers a possibility to deploy easily new architectures, protocols and services.

The remaining part of the paper is structured as follows. First, a short account of the literature connected with the subject is given in Section II. In Section III-A, Xen general architecture is overviewed. Then, a detailed Xen networking structure is presented in Section III-B. A description of the Xen schedulers is presented in Section III-C. Section IV-A describes the testing environment and its parameterizations. The results and discussion on them are contained in Section IV-B. Finally, propositions of methods for improving the network performance isolation in Xen are presented in Section V. Conclusions are gathered in Section VI.

II. State of the art

This study verifies that there are problems related to the performance and isolation of virtualized network resources. Network adapter sharing and scheduling without virtualization is well described in literature. Virtualized environments, however, introduce additional software layer what does not allow to apply directly the existing solutions. Considering virtualized environments several previous studies [10]–[16], [27]–[29] focus on analysis of the performance of I/O operations and some of them present partial solutions. Unfortunately, these studies do not examine isolation and manageability in the field of resource sharing in considered virtualization platforms. In [17], however, the authors tried to approach the performance isolation problem focusing on all kinds of resources. Unfortunately, this study was performed on older version of Xen with an older CPU sched-

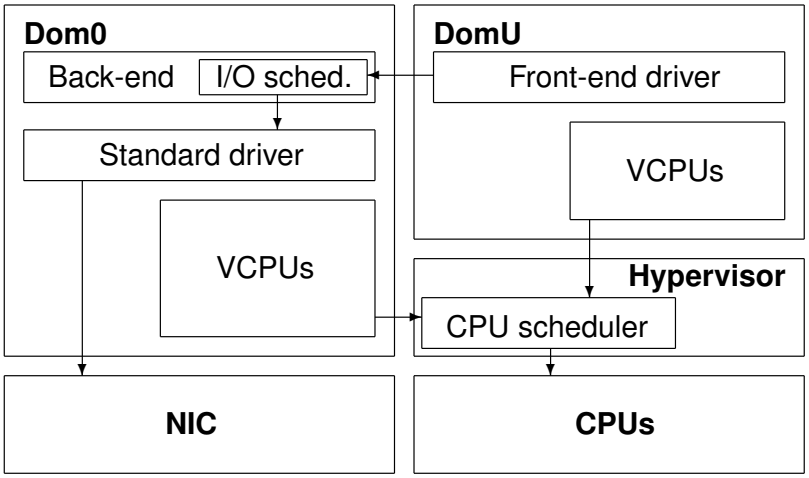


Figure 1. Xen architecture. Dom0 - Xen primary virtual machine, DomU - other Xen virtual machine, Hypervisor - main Xen operating system running directly on hardware, NIC - Network Interface Card, VCPU - virtual CPU.

uler implementation. They assumed that the main source of the problem is connected with CPU assignment and scheduling. As they have proven such general improvement idea can partially increase the performance isolation of all resources. We think, however, that to achieve really good performance isolation across virtual network adapters, the proposed CPU scheduler improvement is not the only change that has to be made because in the existing Xen networking implementation packet scheduling is performed randomly. We present that even on a low CPU utilization the problem is still noticeable and is related to the network scheduler itself. We have verified that applying a modified (for virtualization purposes) Weighted Round Robin (WRR) network scheduler improves the performance isolation and provides better control over virtual network devices.

III. Xen virtualization architecture

A. Xen VMM

Different virtualization environments have been developed throughout the years. Xen, due to its unique architecture (Fig. 1), is one of the leading solutions. The core of Xen, which is responsible for control over all virtual machines, is a tiny operating system called Xen Hypervisor. Its main tasks are CPU scheduling, memory assignment and interrupt forwarding. In contrast to other VMMs, the virtualization of all other resources is moved outside the hypervisor. Such original approach has the following advantages:

- Device drivers are not limited to the hypervisor operating system because they are installed on a virtual machine (any OS),
- Device drivers, as the most vulnerable software, are isolated from the hypervisor, significantly increasing the stability,
- Distributed virtualization of resources allows creation of several driver domains, eliminating the single point of failure,
- Small hypervisor operating system is much more reliable, efficient and stable.

There are two main virtualization methods. The first one allows to run any kind of OS and emulates all the necessary hardware to create an impression that the guest system is running on a physical machine. Second approach is to run a modified guest operating system, which is ”aware” of being virtualized. The latter, called *paravirtualization*, is much more efficient, but limited to some operating systems only. Xen provides both methods, but performs much better in the paravirtualization mode, which will be the only method used further in this paper.

To make the I/O operations as fast as possible, Xen introduced also paravirtualized device drivers. Each guest domain (Xen VMs are also called ”domains”) has the front-end drivers installed. Such drivers, provided with Xen, are communicating with the back-end drivers running on a special driver domain (Dom0 in Fig. 1). All requests addressed to a certain hardware are first scheduled and processed by the back-end driver, then are sent to the standard device driver inside the driver domain and finally reach the hardware. Thanks to Xen internal page-flipping mechanism called XenBus [18], [19] such solution is much more efficient than the standard emulation technique.

B. Xen networking architecture

To perform analysis of the network-related problems it is necessary to explain Xen networking architecture in detail. For each virtual interface the back-end network driver, called *Netback*, creates a virtual network interface in domain0 called *vif*. All virtual interfaces which share

the same physical device are connected with it using a standard Linux level-2 bridge. The Netback process which is responsible for handling traffic of each virtual interface is scheduling this traffic and passing it to the bridge. The existing scheduling scheme implemented in Xen by default is a simple Credit Scheduler and will be described in section III-C2. Finally, the bridge passes the packets to the device driver output queue and the device driver sends the packets to the hardware. Figure 2 presents the outgoing traffic path through Xen virtualization platform.

Such solution has several advantages. Firstly, the administrator has direct control over virtual interfaces from within domain0. Secondly, it is possible to monitor and analyze the packets passing through these interfaces using standard tools like *tcpdump* [20] or *wireshark* [21]. Finally, the traffic can be filtered and manipulated on the bridge level by creating custom ethernet bridge rules using *ebtables* [22] utility. All the above can be applied for both in and outgoing network traffic.

C. Xen schedulers

The main goal of this study is to examine the network performance isolation across Xen guest domains. It means to check if activity of one virtual machine influences the network performance of any other. The resulting knowledge is of great importance from the perspective of many network-related applications.

There are two elements in Xen, which may influence such isolation, namely the CPU scheduler and the network I/O scheduler [23]. Despite the fact that the schedulers are very simple algorithms, their analytical analysis is still far from being solvable. Creating a mathematical model of such systems even with large approximation is a very complex task and very often proves to be impossible. In the following two sections a description of the two schedulers is given.

1) CPU Scheduler: The fundamental part of each multi-tasking operating system is the CPU scheduler. Its aim is to create an impression that all running processes are executed in parallel. Typically, there are much more processes than available physical CPUs and the processes have to share CPU time. The scheduler is responsible for this division.

Inside Xen VMM, the hypervisor is the main operating system running on the physical machine. It is responsible for scheduling physical CPU time among virtual machines. To make the process easier the term *virtual CPU (VCPU)* is introduced. Every VM in Xen can have multiple virtual processors. Also, every domain is running operating system with another scheduler, which divides a VCPU time among processes running inside the guest operating system. The hypervisor on the other hand, schedules the physical CPU time among VCPUs.

The newest version of Xen uses the *credit scheduler* [24], [25]. It assigns two parameters for each domain - *weight* and *cap*. The weight defines how much CPU time a domain gets comparing to other virtual machines. The cap parameter is optional and describes the maximum amount of CPU a domain can consume. Using this two parameters the number of credits can be calculated. As a VCPU runs, it consumes credits. While VCPU has existing credits, its priority is called *under* and it gets CPU time normally. When there are no credits left, the priority changes to *over*. Each physical CPU maintains its own local VCPU queue. In the first place, the VCPU tasks with priority *under* from the local queue are executed. Then, if there are no VCPUs with priority *under*, the scheduler looks for such tasks in other CPU queues. If there are no tasks with priority *under*, the tasks with priority *over* from the local queue are executed. The credit scheduler in Xen can be summarized in the following algorithm and diagram (Fig. 3):

1) Process preemption - the scheduler takes control over CPU.
2) Last taken VCPU inserted back into the local queue according to its credits number.
3) Have the highest priority VCPU from the local queue used all its credits?
   - No: Highest priority VCPU taken from the local queue.
   - Yes: SMP Load Balancing - highest priority VCPU taken from other CPU queues.
4) Switching context to the currently taken VCPU - the VCPU takes control over CPU.

Considering this CPU scheduler in the context of the network performance isolation, it is worth noticing that the scheduler operates on virtual CPUs only, so it should not have a strong impact on I/O performance. The network I/O scheduler, on the other hand, works as a kernel thread inside domain0 using its VCPUs so a fair VCPU scheduling scheme should not influence its performance. However, it may happen that one misbehaving VM will slow down the total responsiveness and performance of other domains. Also, as it was presented in [17], the Xen CPU scheduler does not take into account the amount of CPU consumed by the driver domain on behalf of other VM. This may also have an impact on the network performance isolation, as some domains may use more CPU time than they are allowed. Furthermore, a different type of I/O request (e.g., more demanding, like disk driver requests) can potentially slow down the driver domain and affect the network performance of other VMs.

2) Network I/O scheduler: Looking at Xen architecture and analyzing its source code from the network performance isolation point of view, one can easily note that the most interesting part is the aforementioned Netback driver. It contains another scheduler, responsible for gathering all I/O requests sent to a certain physical network adapter. This network scheduler is not a complex mechanism and probably can be improved. Its only configuration parameter is the

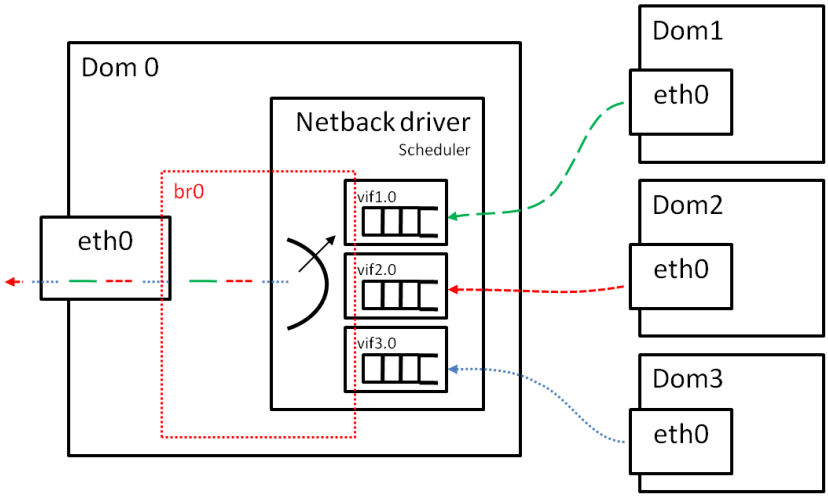


Figure 2. Xen networking architecture.

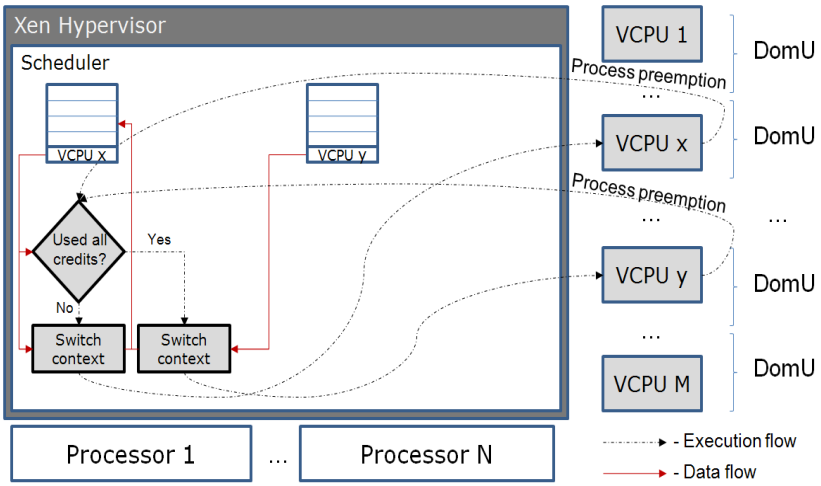


Figure 3. Xen CPU scheduler

maximum rate (parameter *rate*) - in fact it can be perceived as the credits number in the scheduler. The administrator can specify only the maximal throughput achieved by a certain virtual network adapter. Unfortunately, there is no way to prioritize and control the quality of service in more details.

The scheduler itself counts the amount of data sent/received in given periods. If *rate* has been reached, it sets a callback to process the request in next periods. Such solution is efficient, but does not guarantee any fair share or quality. In fact, a misbehaving VM can theoretically flood driver domain with requests because it processes all of them even those which are further rejected.

IV. Experiements

A. Experimental setup

To perform the tests, we installed Linux Gentoo with Xen 4.0.0 on Intel Quad Core 2 (2.83GHz), 4GB RAM, with hardware virtualization support. Two guest domains, each having 1 VCPU and 1GB of RAM, were created. Although there were separate physical CPU available for each VM, both VCPUs were pinned to the same physical CPU. Such configuration was used in order to check the influence of the CPU scheduler on the network performance. All network measurements were taken using *iperf* application. The UDP protocol transferring datagrams of 1500B to an external host over 100Mb link was used. We used the 100Mb link (instead of 1Gb) to demonstrate that the isolation problems are still present without a heavy CPU utilization. Only the outgoing

traffic was measured, as this was our main point of interest. The testing environment is presented in Fig. 4.

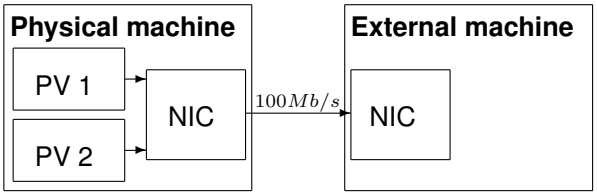


Figure 4. Testbed configuration. PV1, PV2 - Xen paravirtualized machines, NIC - Network Card Interface.

B. Results

In the first experiment, we observed how activity of one VM can affect the performance of another, when both VMs are configured with the same *rate* parameter. Four values of *rate* were used in different test runs: $25Mb/s$, $30Mb/s$, $35Mb/s$ and $40Mb/s$. In every run one machine started its transfer at the very beginning and the other started after 5s of delay. For every *rate* value, the experiment was repeated 10 times and the 0.95 confidence intervals were derived. The results are presented in Fig. 5.

Firstly, we can see that the actual rate is always a little smaller than *rate* parameter. As for the performance isolation, it is not too bad for low values of *rate*. However, with growing *rate*, the confidence intervals are getting larger and larger - in sample runs we can observe stronger variations of the throughput achieved by each VM. For the value of *rate* equal to 35Mb/s, the performance isolation becomes rather weak (although only about 60 percent of the total bandwidth is consumed). It is also worth to mention that a single VM throughput is stable even above $80Mb/s$ what proves that this effect is in fact a performance isolation issue.

Thus the only way to achieve a good isolation is to limit virtual adapters by far, which is not a satisfactory solution. Also, it is worth mentioning that having only the upper limit parameter is not enough in many cases. It would be much better to have any means to prioritize certain virtual adapter or even to have a minimum rate parameter and a scheduler satisfying these requirements.

In the second experiment, different *rate* values per each VM were used. Fig. 6 shows results for $rate = 30Mb/s$ in one VM, and $rate = 40Mb/s$ in another. The isolation problem still remains but, what is worth noticing, both VMs affect each other similarly.

In the presented two experiments the performance isolation problem was either mild or moderate, depending on the configuration. In the following two experiments, we will demonstrate more severe performance isolation issues.

In the third experiment, we verified how Xen divides available bandwidth among two VMs when the maximal rate is not set. A sample path of the throughput achieved by each

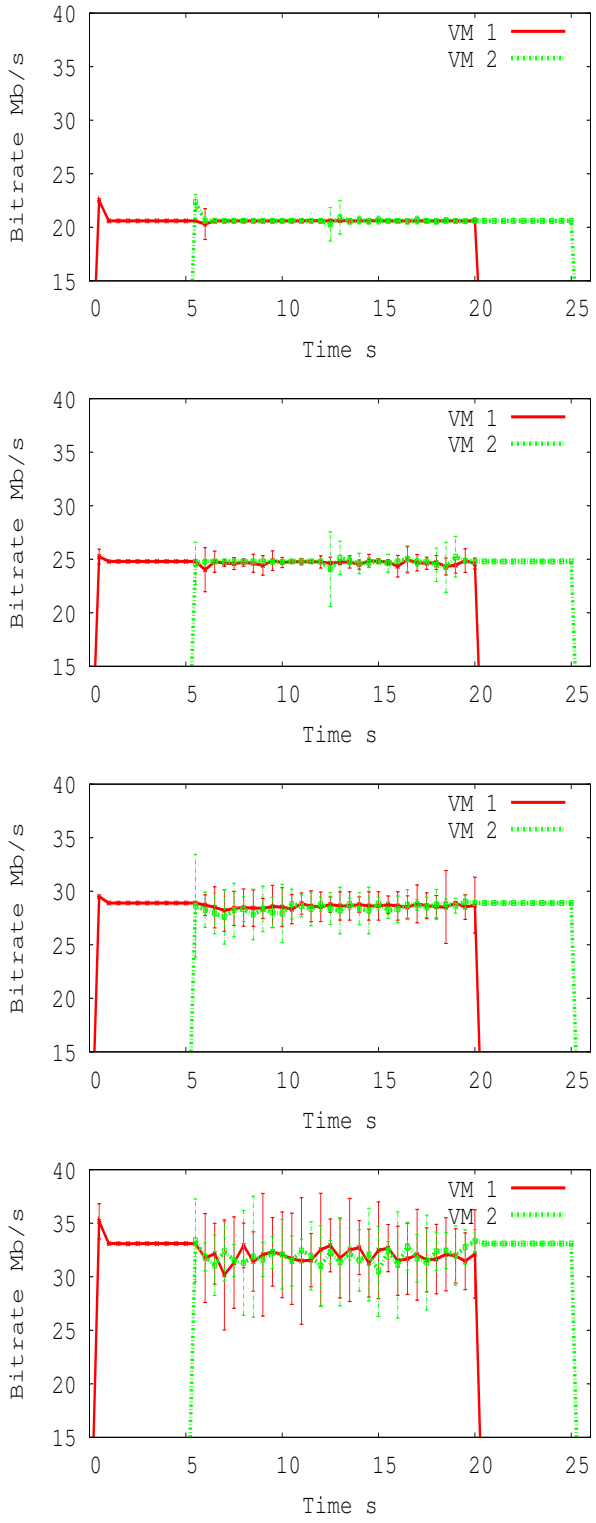


Figure 5. The throughput per VM for different values of *rate* parameter, namely for $25Mb/s$, $30Mb/s$, $35Mb/s$ and $40Mb/s$, counting from the top.

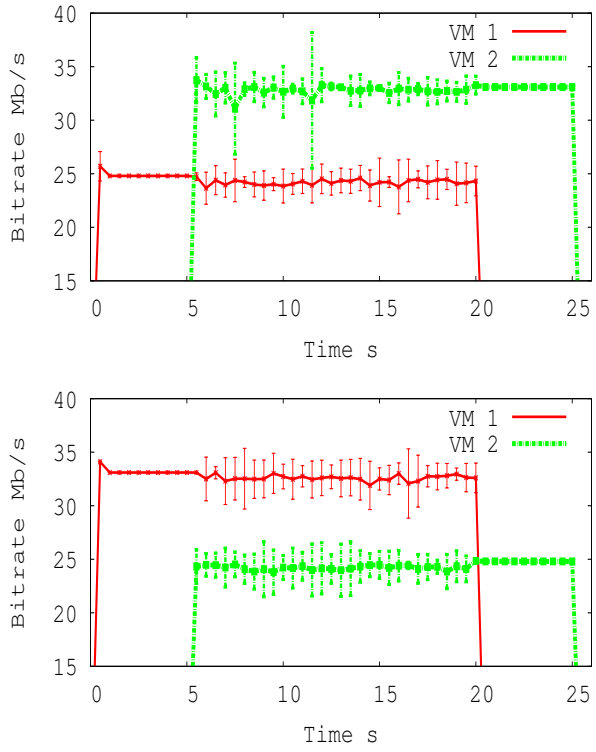


Figure 6. Total throughput per VM for different values of *rate* parameter ($30Mb/s$ and $40Mb/s$.

VM in time is presented in Fig. 7. Surprisingly, sometimes one virtual machine gets the total throughput and the other's throughput decreases to 0. Moreover, there are long periods when one VM dominates the other by far. Therefore, we have in fact no performance isolation at all in this case.

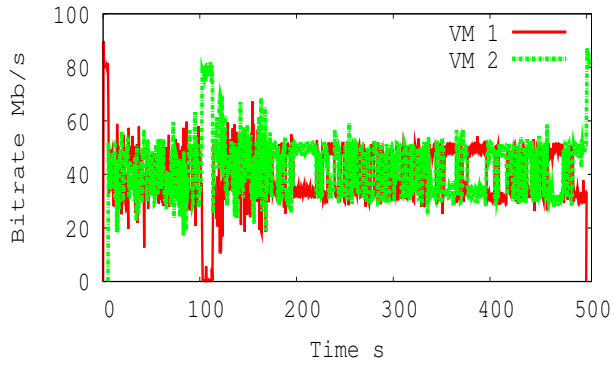


Figure 7. Sample throughput processes in time for two separate VMs without limits

In the fourth set of tests, we wanted to verify if a very abusive virtual machine can take more bandwidth than others. This time we wanted to check the performance isolation of the network I/O scheduler only, therefore we pinned one physical CPU to each VM.

In the first test, one domain was trying to transfer data over one connection using full available speed, while the second domain was using two connections, both of them trying to achieve full available speed. In the next test, the second domain was using three connections at full available speed.

The results are presented in Fig. 8. As it can be observed, the more abusive domain is, the better throughput it achieves. Naturally, if the rate parameter had been set, the overactive domain would never have crossed the maximum rate. In the lower ranges however, the problem remains.

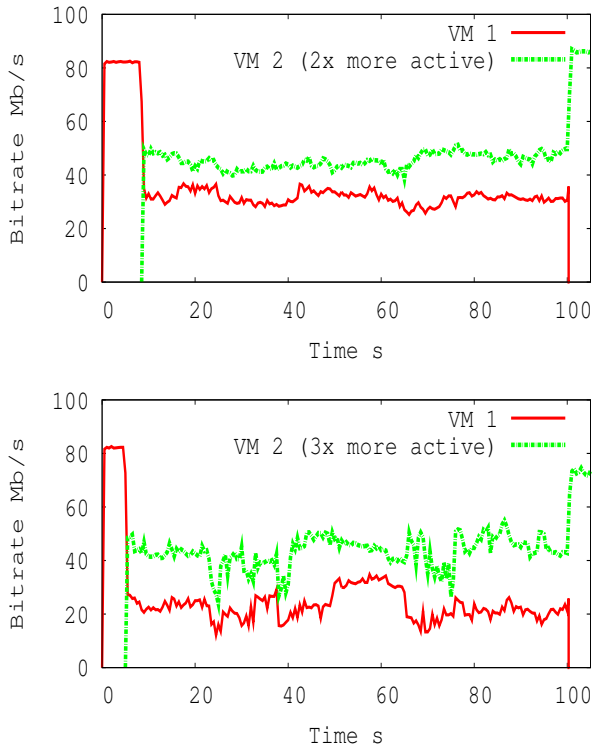


Figure 8. Bandwidth division with one overactive VM.

In the last experiment, we wanted to check if non-network I/O requests can influence the network performance isolation of another domain. During the experiment one VM was constantly sending datagrams at full speed, while the second VM was performing some extensive disk operations (*fio* tool was used for this purpose). The results are presented in Fig. 9; t_0 and t_1 are points in time when the extensive disk operations were initiated and finished, respectively.

We can see that other I/O request can also have a strong impact on the network performance. This is probably caused by driver domain not being able to process all the I/O requests. Block device access is being handled by separate block device back-end drivers. Disk operations are much more demanding in the driver domain than the Netback

drivers because disk is used by many crucial system components and when a disk request is in a blocked state waiting for a response all other mechanisms using disk are blocked as well causing the whole system to perform badly.

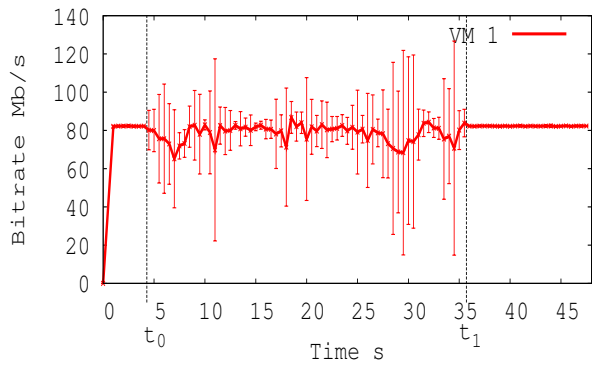


Figure 9. Disk I/O influence on network performance. (t_0 - disk I/O start, t_1 - disk I/O finish)

V. Improvements

After detailed analysis of the problem, we have gathered some ideas on how to modify Xen to improve the network performance isolation. Currently, in the driver domain several Netback kernel threads can be running, depending on the number of VCPUs. Furthermore, several virtual network adapters are mapped with one Netback kernel thread dynamically and this single Netback thread schedules the work using a simple round-robin algorithm, additionally taking into account *rate* parameter (omitting adapters, which used up all their bandwidth in the current period). Our idea is to introduce two additional parameters for every virtual adapter, namely *priority* and *min rate*. To implement the former, it would be necessary to change the round-robin mechanism to a more advanced priority based queue. Of course, we have to remember that the algorithm should not increase significantly the time complexity. The *min rate* parameter could use the same prioritization mechanism, assigning higher priorities to interfaces, which have not yet achieved the minimum rate. Depending on the results, it may be also necessary to introduce a user level application for maintaining the niceness level of each Netback thread inside the driver domain, according to actual needs.

1) Prioritization: The very first step to solve all the aforementioned problems is to introduce a prioritization mechanism into Xen's Netback driver. This will allow for better control over virtual interfaces and additionally schedule the packets in more predictable way thus preventing guests domains to flood the backend driver with requests what should result in improved isolation. To achieve such functionality we implemented the simple *Weighted Round Robin* algorithm [26]. We decided to use the WRR because of its simplicity, low complexity and to present that even the basic scheduling scheme can improve the performance isolation by far comparing to the native random scheduler. The actual implementation is presented in Algorithm 1.

Algorithm 1 The implemented version of WRR scheduler

```
min = infinity
for each vif do
    vif.weight = vif.priority/mean_pkt_size
    if min > vif.weight then
        min = vif.weight
    end if
end for
for each vif do
    vif.packets_to_serve = vif.weight/min
end for
while true do
    for each vif do
        if vif.has_packets_to_send() then
            counter = 0
            while counter < vif.packets_to_serve do
                event = vif.wait_for_event()
                if event = new_packet_to_send then
                    vif.process_packet()
                    set timer to end of transmission time
                else if event = timer_elpased then
                    counter++
                end if
            end while
        end if
    end for
end while
```

In a virtualized environment where a packet passes several virtual adapters before it reaches the actual real interface and each interface has its own input buffer, the WRR scheduler has to be modified to guarantee that the scheduled packets will not be dropped before they reach the wire. Dynamic and real-time priority assignment in this scheduler was created by additional Linux kernel *sysctl* parameters, i.e., *prioritize* and *priorities*. The first parameter defines whether to use the *WRR* scheduler or not. Second parameter is an array of the actual priority values for each virtual adapter.

Each *vif* has a separate queue of data to transfer and a *priority*. The latter corresponds to the weight in the implemented WRR algorithm. Total bandwidth available at the physical link is shared proportionally between all active virtual interfaces according to their weights. Because the packets are scheduled at the virtual driver level, they are processed almost immediately. This may cause wrong behaviour when the queue gets empty and after some time recieves a new packet while the last one is still transmitting. In standard WRR implementation the new packet would be transmitted because of the blocking send operation. This is why we had to introduce a waiting mechanism (in means

of a timer) so that all transmitted packets are actually sent before switching to the next queue.

To test the prioritization we performed simple experiment where two VMs transmit data to an external host. In the meantime the priorities were changed every second. At the beginning VM 1 had much bigger priority, in the end VM 2 was favored in the same proportion (i.e., $30/1$). The results are presented in Figure 10.

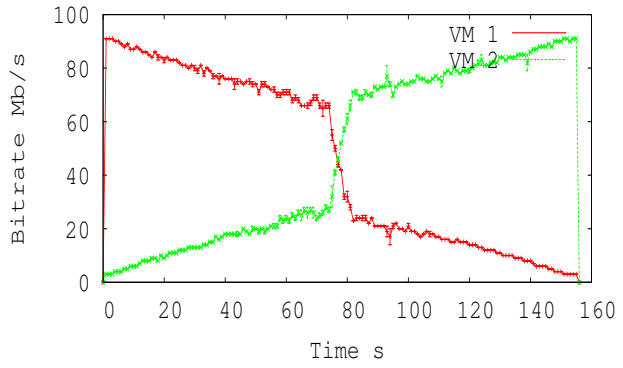


Figure 10. Results of the improved scheduler for changing priorities of each VM.

2) Buffer overflow in Domain 0: After implementing the WRR scheduling scheme the results in terms of performance and isolation were better but still not satisfactory especially at high throughputs (i.e., around the output device bandwidth). Gathering a small sample of output traffic at the physical device led us to the conclusion that the WRR scheduler is working correctly for most of the time however sometimes the packets from different *vifs* are not sent in correct amounts (according to WRR weights).

Knowing the above and that the scheduler itself is implemented correctly, it became obvious that the output traffic is being distorted after scheduling. Furthermore, looking at Xen networking architecture (see section III-B), one can easily notice that the scheduling is applied before packets get to the bridge and finally to the physical device output queue. Normally, when the output queue is getting full the driver informs higher layers and stops packet transmission using *netif_stop_queue* function. Xen Netback driver implementation lacks a mechanism of verification the output buffer of physical device before sending data to bridge. This results in distorted scheduling and larger amount of packet drops.

We modified the Netback module adding such verification. After detecting that the physical device output queue is full the packet is left in the *vifs* output queue. Of course this may lead to situation when the buffer of virtual interface gets full as well what finally results in *netif_stop_queue* at domainU level what is desired and makes the *vif* operation more similar to the operation of real hardware drivers.

3) Improved scheduler results: After applying the modification we have repeated the experiments to verify if the

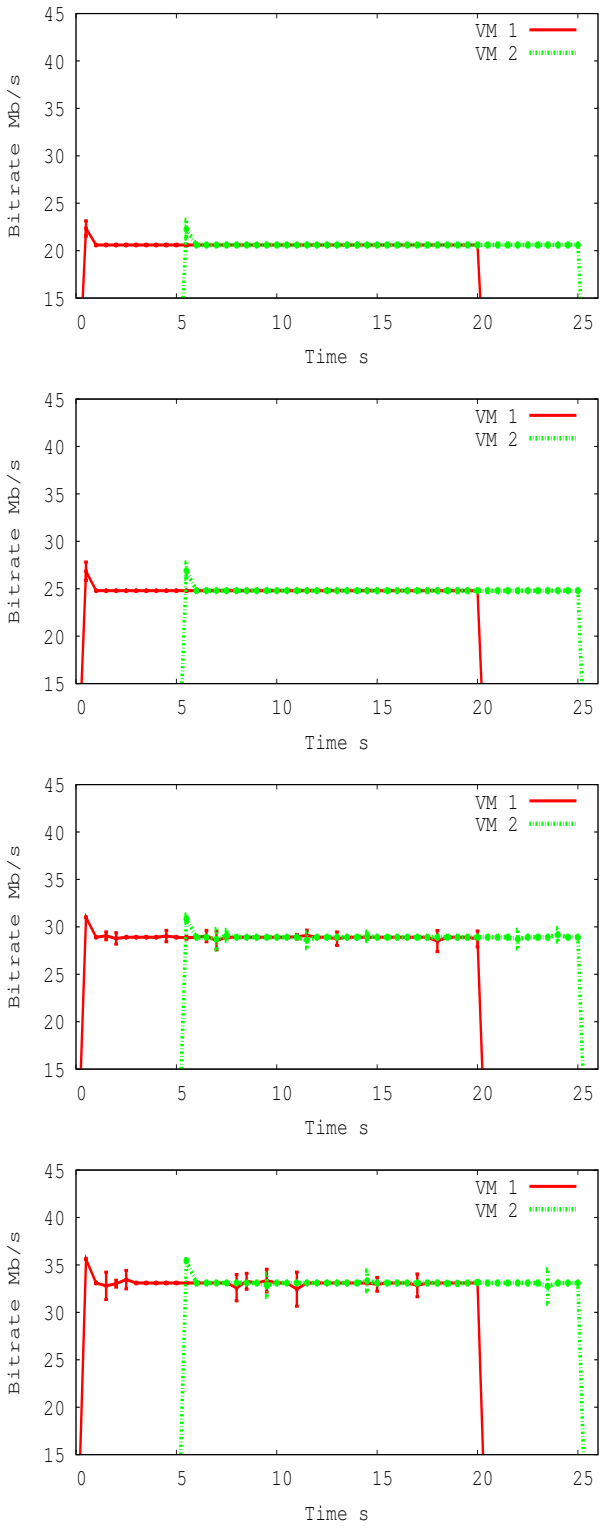


Figure 11. Results of the improved scheduler taking into consideration output buffer for *rate* parameter equal to $25Mb/s, 30Mb/s, 35Mb/s$ and $40Mb/s$.

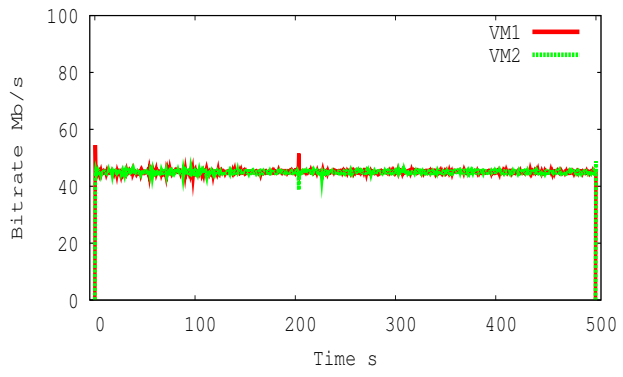


Figure 12. Results of the improved scheduler for one long measurement without $rate$ parameter.

modified scheduler indeed provides better isolation. Figures 11 and 12 present the results.

Comparing Figures 5 and 11 it can be easily seen that the WRR scheduler makes the bandwitdh sharing fair and stable. There does not seem to be any influence of one virtual interface on another. There are, however, some fluctuations (even if only one *vif* is transmitting) which were not noticed in the unmodified version what makes us think this is a minor problem which can be resolved and will be a subject of our further investigation.

Considering Figures 7 and 12 one can see that the modified Netback driver again makes the bandwidth sharing fair and predictable. Both virtual machines get more or less the same result and none is favored nor discriminated. In our opinion, this experiment shows the most significant benefits of applying our modification. In unmodified Xen environment there is no actual network scheduling what makes the outcome (both bandwidth and delays) dependent only on CPU scheduling and assignment. Application of WRR scheduler influences the way virtual interfaces send data and thus require CPU. This, as we can see, significantly decreases the CPU scheduling influence. Unfortunately, the disk I/O influence is still not fully addressed by our solution as we changed only the packet scheduling mechanism. Probably a good idea to minimize this effect would be to additionally use the solution proposed in **??** but unfortunately we could not verify this as we used a newer version of Xen.

It is worth to notice that the applied packet scheduler by improving the fairness and isolation can also positively influence the delays. We think that for our testing scenario, where the packet sizes were constant and the most important factor was the throughput, the WRR algorithm was a good choice. However, for other scenarius and use cases it might be better to implement a different scheduling scheme which may improve the interesting parameters. For example, when a real IP networks are concerned the packets have random sizes and thus it would be good to choose the Deficit Round Robin (*DRR*) scheduling or in case when fairness is concerned the Weighted Fair Queueing could be used. Of course, the more complex scheduling algorithm is used the more overhead is caused by the networking layer thus some schedulers may be hard to implement. In case of our implementation the overhead is minimal and does not affect the overall system performance.

Both the prioritization and buffer modification presented above may be of great use for system administrators who are providing services to external clients and wish to have good control over network resources and at the same time maintain the performance isolation at higher level.

4) Further improvements: Prioritization and the aforementioned buffer modification brings a lot of new possibilities and improves the performance isolation by far. Nevertheless, in high CPU utilization scenarios it may be not sufficient. We may think of much more complicated mechanisms. Virtualization makes the problem very complex, as three different schedulers may affect the isolation: CPU Scheduler, Domain 0 VCPU Scheduler and Netback I/O Scheduler. To achieve best results it might be necessary to synchronize all schedulers. Thus, partial solutions providing the *minimal rate* parameter for given virtual interface may prove very valuable. Further, a modification proposed in [17] may also help to increase the performance isolation taking the aggregate CPU consumption into consideration. Finally, we would like to test the scalability of our solution on a better hardware with more VMs running. All these are subjects of our future study.

VI. Conclusion

Xen is a powerful and stable virtualization platform, what accompanied with its Open Source formula makes it one of the most interesting VMMs, especially for research purposes. However, when the network virtualization is considered, the weak point of Xen is its lack of proper performance isolation. We demonstrated this using five sets of tests. The problems with isolation are caused by several factors mostly connected with CPU and I/O schedulers. We proposed the Netback driver modification using WRR algorithm to provide prioritization. We have also briefly presented an idea for future improvements.

VII. Acknowledgments

This work is partially funded by the European Union, European Funds 2007-2013, under contract number POIG.01.01.02-00-045/09-00 ”Future Internet Engineering”. This is extended version of the paper [1], presented during the International Conference on Cloud Computing, GRIDs, and Virtualization, Rome, September 25-30, 2011.

REFERENCES

[1] B. Adamczyk, A. Chydzinski: On the performance isolation across virtual network adapters in Xen, in Proceedings of the International Conference on Cloud Computing, GRIDs, and Virtualization. Rome, September 25–30, 2011, pp. 222–227.

[2] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield: Xen and the art of virtualization, in Proceedings of the 19th ACM Symposium on Operating Systems Principles, New York, 2003, Vol. 37, pp. 164–177.

[3] The 4WARD Project, http://www.4ward-project.eu/index.php, 30-12-2011

[4] A. Galis, et al., Management and Service-aware Networking Architectures (MANA) for Future Internet. System Functions, Capabilities and Requirements, Position Paper, Version V6.0, 3rd May 2009.

[5] AKARI Architecture Design Project, http://akari-project.nict.go.jp/eng/index2.htm, 30-12-2011

[6] The PASSIVE Project, http://ict-passive.eu/about/, 30-12-2011

[7] Global Environment for Network Innovations Project, http://www.geni.net/, 30-12-2011

[8] Future Internet Engineering, http://iip.net.pl, 30-12-2011

[9] T. Anderson, L. Peterson, S. Shenker, J. Turner: Overcoming the Internet Impasse through Virtualization, Computer, Volume 38, Issue 4, April 2005, pp. 34–41.

[10] P. Padala et al.: Adaptive control of virtualized resources in utility computing environments, ACM SIGOPS Operating Systems Review, Vol. 41, No. 3, 2007, pp. 289–302.

[11] Y. Song, Y. Sun, H. Wang, and X. Song: An adaptive resource flowing scheme amongst VMs in a VM-based utility computing, in Proceedings of the 7th IEEE International Conference on Computer and Information Technology (CIT), 2007, pp. 1053–1058.

[12] J. Liu, W. Huang, B. Abali, and D. K. Panda: High performance VMM-bypass I/O in virtual machines, in Proceedings of the annual conference on USENIX, 2006, Vol. 6, pp. 3–3.

[13] V. Chadha, R. Illiikkal, R. Iyer, J. Moses, D. Newell, and R. J. Figueiredo: I/O processing in a virtualized platform: a simulation-driven approach, in Proceedings of the 3rd International Conference on Virtual Execution Environments, 2007, pp. 116–125.

[14] D. Ongaro, A. L. Cox, and S. Rixner: Scheduling I/O in virtual machine monitors, in Proceedings of the 4th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments, 2008, pp. 1–10.

[15] G. Liao, D. Guo, L. Bhuyan, and S. R. King: Software techniques to improve virtualized I/O performance on multi-core systems, in Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, San Jose, California, 2008, pp. 161–170.

[16] S. R. Seelam and P. J. Teller: Virtual I/O scheduler: a scheduler of schedulers for performance virtualization, in Proceedings of the 3rd International Conference on Virtual Execution Environments, 2007, pp. 105–115.

[17] D. Gupta, L. Cherkasova, R. Gardner, and A. Vahdat: Enforcing Performance Isolation Across Virtual Machines in Xen; In Proceedings of the 7th ACM/IFIP/USENIX Middleware Conference, 2006, pp. 342–362.

[18] Y. Xia, Y. Niu, Y. Zheng, N. Jia, C. Yang, and X. Cheng: Analysis and Enhancement for Interactive-Oriented Virtual Machine Scheduling, in Proceedings of the IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, 2008, Vol. 2, pp. 393–398.

[19] Xen Wiki, http://wiki.xensource.com/xenwiki/XenBus, 29-06-2011.

[20] Van Jacobson, Craig Leres and Steven McCanne: tcpdump, Lawrence Berkeley National Laboratory, University of California, Berkeley, http://www.tcpdump.org, 30-12-2011.

[21] Gerald Combs, et al.: Wireshark, http://www.wireshark.org/about.html, 30-12-2011.

[22] B. De Schuymer, et al.: ebtables, http://ebtables.sourceforge.net/, 30-12-2011.

[23] J. Matthews, E.M. Dow, T. Deshane, W. Hu, J. Bongio, P.F. Wilbur, and B. Johnson: Running Xen: A Hands-on Guide to the Art of Virtualization; Prentice Hall; April 2008.

[24] L. Cherkasova, D. Gupta, and A. Vahdat: Comparison of the three CPU schedulers in Xen, SIGMETRICS Performance Evaluation Review; September 2007, Vol. 35, No. 2., pp. 42–51.

[25] G. W. Dunlap: Scheduler development update, Xen Summit North America 2010, http://www.xen.org/files/xensummit_intel09/George_Dunlap.pdf, 29-06-2011.

[26] A. K. Parekh and R. G. Gallager: A generalized processor sharing approach to flow control in integrated services networks: The single-node case; IEEE/ACM Transactions on Networking; 1993, Vol. 1, pp. 344–357.

[27] G. Somani and S. Chaudhary: Application Performance Isolation in Virtualization in Cloud Computing; CLOUD 09. IEEE International Conference, 2009; pp. 41 48.

[28] N. M. M. K. Chowdhury and R. Boutaba: Network virtualization: state of the art and research challenges; Communications Magazine, IEEE, vol. 47, no. 7, pp. 20 26, Jul. 2009.

[29] P. Yuan, C. Ding, L. Cheng, S. Li, H. Jin, and W. Cao: VITS Test Suit: A Micro-benchmark for Evaluating Performance Isolation of Virtualization Systems; in e-Business Engineering (ICEBE), 2010 IEEE 7th International Conference on, 2010, pp. 132 139.

Detailed Analysis for Implementing a Short Term Wind Speed Prediction Tool Using Artificial Neural Networks

Aubai Alkhatib
University of Kassel
REMENA
Kassel, Germany
alkhatibaubai@yahoo.com

Siegfried Heier
University of Kassel
REMENA
Kassel, Germany
heier@uni-kassel.de

Melih Kurt
Fraunhofer IWES
Kassel, Germany
mkurt@iset.uni-kassel.de

Abstract **- Wind speed forecasting is an essential prerequisite for the planning, operation, and maintenance works associated with wind energy engineering. This paper attempts to forecast fluctuations based only on observed wind data using the data-driven artificial neural network approach. Wind fluctuations with varying lead times ranging from a half year to a full year are predicted at Al-Hijana, Syria with the pre-preparation for the available data. Two layers of feed-forward back-propagation networks were used along with the conjugate gradient algorithm and other tested training functions. The results show that artificial neural network models perform extremely well as low values of errors resulting between the measured and predicted data are obtained. The present work contributes to previous work in the field of wind energy independent power producer market and may be of significant value to Syria, considering that the country is currently in the process of transitioning into a free energy market. It is likely that this modeling approach will become a useful tool to enable power producer companies to better forecast or supplement wind speed data. Two main types of wind speed prediction tool is discussed in this paper. One prediction tool with no time shift and the other prediction tool with time shift, where in the second type two different time periods were used to show the different between long term prediction and short term prediction.**

Keywords - Artificial Neural Networks; Wind Speed; Mean root square error; Training functions; Short term prediction; Long term prediction.

I. INTRODUCTION

The wind-energy spread usage in recent years is an attempt to address the environmental problems that result from the consumption of energy and especially from nuclear power plant disasters like the one that recently occurred in Fukushima, Japan [1]. The IPCC (Intergovernmental Panel on Climate Change) indicated that [2] human activities are directly related to increased atmospheric levels of greenhouse gasses, i.e., carbon dioxide, methane, chlorofluocarbons, and carbon monoxide. Additionally, a correlation also exists between global warming involving greenhouse gas and environmental problems. It is generally agreed that of those harmful greenhouse gasses, carbon dioxide contributes the most to global warming. The main artificial source of carbon dioxide discharge is derived from fossil fuels (conventional power plants). Therefore, much recent research has focused on reducing the consumption of fossil fuels and replacing those with renewable, environment-friendly energy sources. Currently, wind energy is considered as one of the most promising energy sources. However, since wind is difficult to manage, generating wind energy is still a challenge. Due to a variety of factors, the wind speed characteristic curve can change with time, location and height. Wind blows as a result of an imbalance in the quantity of heat on the earth by the energy from the sun. Experimentally, it is known that wind speed is intermittent, irregular, and frequently fluctuates in the short term. Since wind energy is directly related to the cubic value of the wind speed, any changes in the wind speed will greatly impact the amount of the energy. In order to better support the transition to a free energy market, a more accurate means of estimating the energy generated from the wind farm and pumped in the grid is needed.

This paper thus introduces an ANN (Artificial Neural Networks) for wind speed predictions to estimate the wind speed in a suggested location in Syria that involves two main approaches.

1- No time shift approach.
2- Time shift approach: which contain:
   a. A one year prediction tool (old approach – long term-).
   b. A half year prediction tool (new approach, see the next point -3-).
3- Time shift new approach (short term): is suggested in is work as result of analysis the outputs generated from the previous tools.

Also, the different possible ways that can be used in order to improve the prediction output (e.g., choosing different training functions, which are introduced by Matlab). This paper also introduces a model for energy estimation using the output of the wind speed prediction tool as an input for the energy model. Using the Matlab computing program for building the suggested ANN is one of the future computing methods for wind prediction.

Finally the steps needed for building the suggested wind speed prediction tool & the energy module are presented in the same order as done is this research, from the part of getting the row data and analyzing the importance of every used input using speerman analysis, till getting the final results obtained from the energy module. A suggestion for new approach is ales presented in this paper for predicting

the wind speed (with time shift) taking into consideration the unexpected changes in the wind speed function

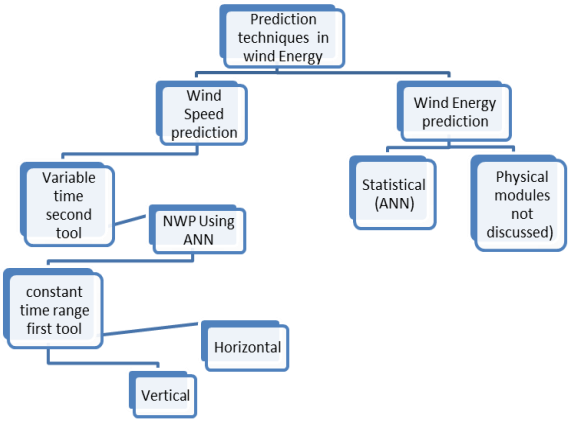


Figure 1. Prediction techniques

Figure 1 shows the different prediction techniques used in the wind energy field. The physical module in wind energy prediction section is not discussed in this paper and the variable time prediction technique is the time shift tool (old and new approach). The constant time range is the not time shift approach.

II. WIND SPEED PREDICTION TECHNIQUES

In this part a detailed description will be illustrated for the actual steps used in this research in order to build an acceptable wind speed prediction tool using the artificial neural networks approach.

A. state of the art

The wind speed characteristic can be considered a non-linear fluctuation. Therefore, the forecasting of this function using traditional methods (e.g., numerical weather prediction (NWP) which is used in Germany nowadays) can be very difficult and time consuming. In this case, the intelligent engineering represented by a neural network, a chaos fractal, and a genetic algorithm, etc. can be applied. While these techniques are already adopted in numerical predictions, the usage of the ANN gives a better performance in terms of pattern recognition and finding location peculiarities, especially when information on the used wind turbine and power curve is given [3]. That is why the focus in this work will be on developing an acceptable wind speed prediction tool using ANN's and showing the different possibilities of sizing this tool with a new approach for minimizing the errors resulting from the used prediction tool.

B. The usage of the proposed wind speed prediction tool

There are two different types of wind speed predictions [4]:

The vertical wind speed prediction or the prediction of the expected wind speed curve in one point on the geographical map with different height. This can be seen, for example, when the wind measurement device is at a height of 40 m and the wind turbine is installed in the same location yet in a different hub height like 105 m

The horizontal wind speed prediction or the prediction of the expected wind speed curve in one point on the geographical map that has a horizontal difference from the point of measured data. This is witnessed when the wind measurement device is in one location and the wind turbine is installed in another location (top of a mountain where the measurement is very difficult to be obtained) [5].

In both cases the no time shift tool can be used in order to get the predicted wind speed at the height of the used wind turbine in a new location with no available measurements. In this tool it is enough to know the pressure and temperature of a nearby location (not at the same place of the location of interest).

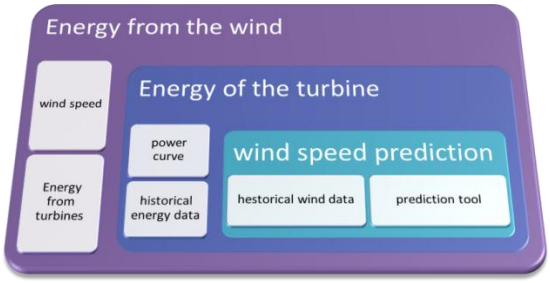


Figure 2. Wind speed prediction and energy module connections

The historical wind data shown in Figure 2 indicates that the atmospheric parameter measurements such as the pressure and temperature that were available for the location in our case. The energy historical data indicates a previous energy output for a previous wind turbine installed in the location of interest, which was not available in our case (as this was the first wind farm to be installed in this location).

Finally a wind prediction tool was built in order to predict the wind speed in locations where no wind speed measurement devices are available which means to predict the wind speed from the available data of a specific site (Like pressure, temperature …etc.). In this case no time shift is introduced, which mean that the prediction is done for the measured data for the same time.

C. Feed Forward Neural Network with Backpropogation

A neural network is a computational structure that resembles a biological neuron. It can be defined as a

"*massively parallel distributed processor made up of storing processing units, which has a natural propensity for storing experimental knowledge and making it available for use*" [6].

A feed-forward neural network consists of layers. Every layer will be connected to the previous once with more than one connection that has a weight to determine the importance of this connection. Every network has at least three layers. These include the input layer, output layer, and the hidden layer(s). The strength of a set of inputs can be determined by the activation function after adding the whole input signals as shown in Figure 3.

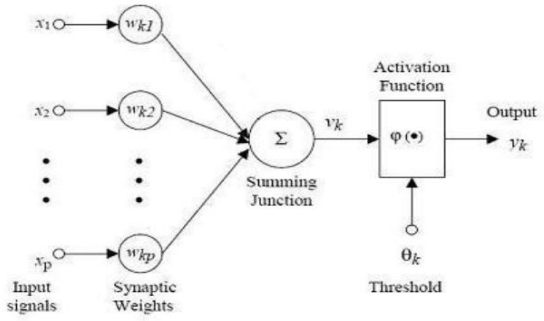


Figure 3. Basic structure of a neuron [6]

The raw data was provided in a form of Excel file. Patterns were generated and a statistical analysis performed to get a good correlation among the input values. Some data was fed as an input in the prediction network for training purposes while other data was specifically employed for network testing purposes.

The following steps were taken to get the wind speed prediction:

1- Data Acquisition & Pre-processing.
2- Data conversion & Normalization.
3- Statistical Analysis.
4- Design of the Neural Network &Training.
5- Testing.

D. Data Acquisition, Pre-processing, and Data Conversion

The different weather parameter values were collected from the used measurement devices in the location of interest [7]. Time series was provided for every ten minutes with the help of the Syrian National Energy Center for Research and Development. The values of three different parameters were utilized to include the pressure, temperature and wind direction as shown in Table I.

TABLE I. LIST OF NETWORK PARAMETERS[8]

station	day	hour	speed 40 s wvt	direct 40 d1 wvt	direct 40 sd1 wvt	speed 40 max	speed 40 std	speed 10 avg	speed 10 max	speed 10 std	temp	pressure
14	1	0	1.656	199.8	8.55	2.01	0.183	1.184	1.57	0.142	3.344	947.24
14	1	10	1.549	185.6	9.74	1.91	0.164	1.136	1.47	0.097	3.581	947.1
…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…
14	366	2230	2.803	278.1	8.74	4.71	0.643	2.579	3.73	0.48	3.624	945.58
14	366	2240	3.069	282.2	7.22	4.29	0.385	2.593	3.55	0.379	3.392	945.68

During the data acquisition stage, the maximum value among each parameter was computed after that normalization was carried out for all the used parameters [9].

During the visit to the Syrian National Energy Center we have learned that not in all cases a wind speed data is available, which means that sometimes it is needed to predict the wind speed from the available information in location of interest, that is why a scenario of wind speed prediction tool with no time shift is introduced and compared with that used for wind speed prediction with time (the ANN is trained with the available wind speed data and get as a results the wind speed for the next year or six months).

E. Statistical Analysis

Since the amount of available data is massive and the characteristic curve of the wind speed continually changes with time, a statistical analysis is needed in order to measure the extent of the relationship between each of the meteorological values and to get rid of the redundant values that might be present in the data set. Therefore, a "*Spearman rank correlation*" was applied. The amount of correlation in a sample (of data) is measured by the sample coefficient of correlation, generally denoted by 'r' or by 'ρ'.

This analysis is very important for the "no time shift scenario". The results of this analysis can determine, which data are a must for the wind speed prediction tool (As input) and which data have a secondary effect on the output of the prediction tool. As a result of this analysis the measurement devices that should be installed in any location can be determined.

F. Spearman's Correlation

Spearman's correlation allows testing the direction and strength of a relationship [10]. For example the relationship between the pressure and the wind speed will be shown (one of the inputs of the prediction tool and the output) to help determine the importance of this parameter on the output of the prediction. This in turn can give a good vision of the expected output of the suggested ANN tool. This approach can also be applied to problems in which data cannot be measured quantitatively but in which a qualitative assessment is possible. In this case, the best individual is

given rank number 1, the next rank 2, etc. (In our case the highest wind speed, which is the rated wind speed for the wind turbine in which the wind turbine generate its rated power in kW, will get the rank number 1).

The correlation coefficient takes values between [1,-1]. A value of /1/ indicates that the relationship between the two different parameters is very strong and has a positive effect (when "X" increases, "Y" value will also increase). The value/-1/ has the same strength meaning of /1/ yet the relation is inverse. A value of /0/ means that no relationship exists between the two different studied parameters.

Steps for achieving a Spearman's ranking:

A- Rank both sets of data from highest to lowest value (make sure to check for tied ranks - readings of the same value and to obtain the same sequence of readings).

B- Subtract the two sets of ranking data to get the difference /d/.

C- Square the values of /d/.

D- Add up the squared values of the differences.

E- Calculate the values using Spearman's Ranking Formula [10]:

$$R = 1 - \frac{6 \times \Sigma D^2}{n(n^2-1)} \quad (1)$$

Table II shows the results obtained from this analysis for one year data (2008). It can be seen that the pressure has an inverse influence on the wind speed and that the temperature has an indirect effect on the wind speed through an inverse relationship with the pressure [11].

TABLE II. SPEARMEN'S RANKING RESULTS FOR 2008

2008				
Correlation	Wind Speed	Direction	Temperature	pressure
Wind Speed	1			
Direction	0.232188687	1		
Temperature	0.257124942	0.214808385	1	
pressure	-0.49489753	-0.29900903	-0.70568132	1

Figure 4 gives a statistical analysis for 6 years of available data. The figure can be used to clarify the results obtained from the prediction tool as it shows that the atmospheric parameters and the character curve of the wind speed changes on a yearly basis. The information obtained from Spearmen's ranking can help determine which data should be selected as training data in order to contain the best possible situation and get better prediction results for this specific location. What is more it can be noted that the pressure has the most direct influence on the wind speed so it should be used as an input for the prediction tool in any case (with or without time shift scenarios). Although the temperature has no big effect on the wind speed as the pressure it still should be included in the wind prediction tool as it has a good effect on the pressure.

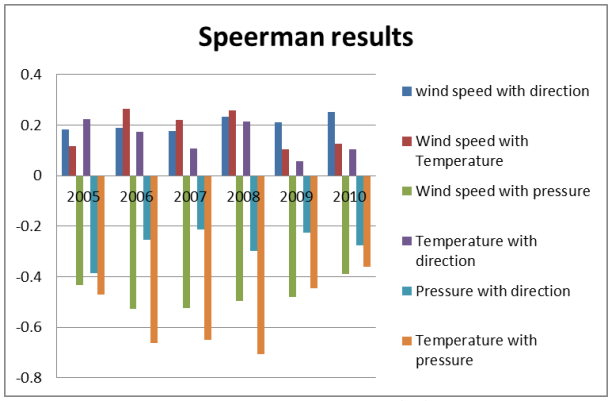


Figure 4. Spearman's analysis

It is important to mention that some ANN tool uses a "helping function". The helping function is normally determined by carrying out a statistical analysis (in our case speerman). This helping function works as pre-determining function, which will help the ANN tool by telling what the expected output should look like avoiding unexpected outputs by the ANN tool. Unfortunately this approach was not used in this work as the purpose of this work was to get the differences of using different internal parameters used in the ANN tool itself (e.g. Training functions, number of neurons, number of the hidden layers, Activation functions, …etc.).

G. *Design of the Neural Network & Training*

Designing the neural network means sizing the network in order to fit our need. Unfortunately, there is currently no mathematical equation for sizing the network or determining which training functions to use [12]. Thus, engineers often rely on trial and error and personal experience to solve these issues. In our case, the sizing of the number of hidden layers, training functions, activation functions, number of neurons in the hidden layers, and determining the best training input pattern was accomplished through trial and error and, as shown in Figure 5, a comparison of the results. Finally, 2 hidden layers with feed forward activity were chosen (as the differences in the results between 2 and 3 hidden layers were very small and neglectable see Table III in comparison to Table IX).

TABLE III. RMS VALUES WHEN USING 3 HIDDEN LAYER PREDICTION TOOL

one year input with three hidden layers and different training data																				
Description	max					min					average					RMSD				
Year		2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
year 2008 training	21.036	21.04	20.37	13.31	9.594	-18.1	-18.1	-17.6	-18	-12.8	3.94646	3.946	-4.51	-0.947	0.057	0.137	0.137	0.1406	0.115	0.049
year 2007 training	13.467	13.47	16.78	9.242	16.71	-15.5	-15.5	-8.71	-7.96	-10.2	-5.5216	-5.52	1.734	0.003	5.963	0.143	0.143	0.0959	0.043	0.135

Using the back propagation algorithm in each training set, the weights were modified in order to reduce the root mean squared error (deviation) (RMSE/D/) between the predicted values and the actual readings as target values. Thus, the modification takes place in the reverse direction

from the output layer until the terminating condition is reached. The steps are:

- Initialize the weights.
- Propagate the inputs forward.
- Back propagate the error.
- Terminating condition.

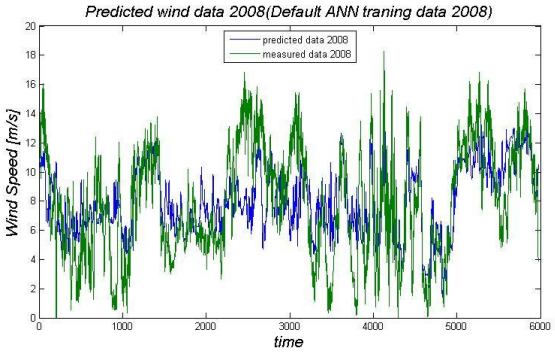


Figure 5. Deviation of predicted and measured wind speed for 2008

Figure 5 shows both the predicted wind speed and the measured wind speed for the year 2008. It is clear that in some points the measured wind speed takes a drastic change is speed (between X=2000 and X=3000 on the time axis in Figure 5) that the prediction tool did not expect, in this case a helping function can be useful to correct the prediction tool results.

H. Testing

Testing is the final stage needed to finalize the proposed wind speed prediction tool. While different methods can be used to evaluate the results obtained from the prediction tool, in this case the Mean Square Error method was used [13].

TABLE IV. RMSE OF THE WIND PREDICTION TOOL WITH DIFFERENT INPUT POSSIBILITIES FOR DIFFERING YEARS.

different inputs with time																				
Description	max					min					average					RMSD				
Years	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
Wind data input	0	17.331	17.479	7.866	11.819	0	-7.8058	-9.7378	-10.206	-7.7274	0	-0.0814	1.14062	-0.0161	1.48332	0	0.0623	0.0824	0.05091	0.0749
all as input	0	11.573	15.195	5.447	11.485	0	-16.792	-16.34	-13.66	-9.8059	0	-5.5984	-2.2118	-3.7902	-0.0325	0	0.13893	0.1214	0.0916	0.0597

TABLE V. RMSE OF THE WIND PREDICTION TOOL WITH DIFFERENT TRAINING DATA FOR DIFFERING YEARS.

two year input for prediction with time tool																				
Description	max					min					average					RMSD				
Year	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
two year 2007-2008	NP	15.68	15.21	8.52	9.099	NP	-17.5	-13.9	-7.88	-10.4	NP	-0.36	-1.6	0.098	0.065	NP	0.098	0.057	0.038	0.057
two year 2006-2007	NP	17.61	15.89	8.731	15.56	NP	-22.7	-8.23	-8.03	-10.6	NP	-0.42	-0	0.257	5.677	NP	0.159	0.0441	0.068	0.127

TABLE VI. RMSE OF THE DIFFERENT YEARS WIND PREDICTION TOOL WITH DIFFERENT TRAINING FUNCTIONS.

different training functions with time																				
Description	max					min					average					RMSD				
Years	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
Bayesian Regulation	0	16.856	14.09	12.35	10.101	0	-10.119	-11.103	-12.967	-10.662	0	-0.6578	-0.4479	-0.7117	-9E-05	0	0.07538	0.0791	0.07472	0.0669
Fletcher-Reev	0	17.202	13.017	8.371	11.139	0	-8.9468	-10.426	-10.182	-8.418	0	-9E-05	-0.3887	-0.7028	0.04619	0	0.07435	0.0786	0.07087	0.067
Marquardt	0	20.416	12.621	11.05	9.8743	0	-18.207	-9.7308	-10.111	-8.6624	0	1.59405	-0.4967	-0.9127	-0.063	0	0.1548	0.0787	0.06833	0.0669
Quasi-netwton	0	16.916	13.239	11.34	10.486	0	-9.0429	-9.3813	-9.9776	-8.5122	0	-1.241	-0.486	-0.8636	-0.0545	0	0.07359	0.0786	0.06932	0.067

The previous tables give the results of the differing sizing possibilities that can be used for the prediction tool. It can be seen that the usage of the pressure, temperature and wind direction as inputs for the prediction tool is more effective than using each parameter alone. Also, the 2007-2008 input data gives better results than the 2006-2007 data because the MSE is better in the first case. After determining which training data to be used as input for the suggested prediction tool, different training functions are tested in order to see which training function results in the lowest RMS values. The next step will be to determine the activation function which is very important to determine which input summation is enough to get a high output values. More information about the different training function and activation function can be found in Matlab help, with a definition of the different properties of every function.

The results for using the "no time shift scenario" are as follows:

TABLE VII. RMSE OF THE WIND PREDICTION TOOL WITH DIFFERENT ACTIVATION FUCNTIONS.

different activation functions (no time prediction)																				
Description	max					min					avarage					RMSD				
Year	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
purelin Act.	14.46	16.4	14.9	7.6	9.977	-7.77	-8.1	-7.95	-17.6	-8.07	0.18	-1.29	-0.48	-1.41	-0.07	0.069	0.0666	0.079	0.061	0.069
logsig Act.	21.04	10.6	10.5	8	9.518	-18.1	-17	-15.9	-17.2	-11.1	3.95	0.982	-3.31	-5.6	-0.03	0.137	0.1043	0.105	0.124	0.056
MSE Perf.	5.703	13.6	11.9	9.17	11.01	-17.2	-15	-16.5	-16.8	-10.4	-7.2	1.473	-2.81	-2.83	0.011	0.15	0.0964	0.098	0.094	0.056
sum Perf.	13.23	13.2	9.79	9.07	10.76	-13.2	-13	-13.1	-17.4	-11.7	-1.1	-1.11	-3.01	-2.49	-0	0.071	0.0715	0.101	0.08	0.062

TABLE VIII. RMSE OF THE WIND PREDICTION TOOL WITH DIFFERENT TRAINING FUCNTIONS.

different training functions (no time prediction)																				
Description	max					min					average					RMSD				
Year	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
Bayesian Regulation	12.227	16.531	18.456	10.86	10.02	-17.98	-18.296	-17.041	-18.297	-10.928	-8.3043	-9.6229	-1.7939	-5.3751	-0.0492	0.17806	0.20016	0.1247	0.1671	0.0515
Fletcher-Reev	18.0054	17.237	17.458	8.371	10.334	-16.149	-16.123	-11.41	-11.64	-11.409	-1.2824	-0.0492	-1.5887	-1.0298	-0.0376	0.11547	0.12557	0.1026	0.07491	0.056
Marquardt	22.7813	20.416	16.424	13.31	10.017	-17.549	-18.207	-17.974	-17.832	-11.179	3.2116	1.59405	-5.3782	-2.0494	-0.0295	0.1733	0.1548	0.1513	0.13921	0.0505
Quasi-netwton	20.6997	13.824	8.5364	11.16	9.5953	-17.654	-18.182	-13.513	-18.29	-10.257	-5.948	-6.6341	-4.2221	-3.2991	-0.0501	0.15514	0.15158	0.112	0.10729	0.0533

TABLE IX. RMSE OF THE WIND PREDICTION TOOL WITH DIFFERENT TRAINING DATA.

default type of ANN with different training data (2008-2007) (no time prediction)																				
Description	max					min					average					RMSD				
Year	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008	2004	2005	2006	2007	2008
2008 training data	15.531	13.959	12.466	8.525	9.5902	-16.225	-17.764	-15.76	-18.238	-10.675	-1.8815	4.10261	-3.013	-4.0005	-0.0306	0.12767	0.12201	0.1051	0.10668	0.054
2007 training data	22.7516	15.671	17.515	8.371	12.153	-15.23	-15.547	-7.4397	-8.5895	-9.8371	4.3987	-4.0795	1.35332	-0.0386	0.09141	0.16641	0.13326	0.0901	0.04862	0.0761

It is clear from the above tables that the RMS errors of a wind speed prediction tool without time shift is better than the one done with time shift, which lead to the conclusion that the same tool can be used for both type of prediction

(with or without time shift with acceptable errors).

The most important conclusion that can be driven from the data in the previous tables is that if the error to be reduced a new approach is needed in order to overcome the errors generated from the unaccounted character changes of the wind speed. In this new approach is to take only half a year into account. Also this half year was divided among collecting testing (or validation) data and training data. The half year period was divided into days, with one day allocated for training and the next one for testing and so on as shown in Figure 6.

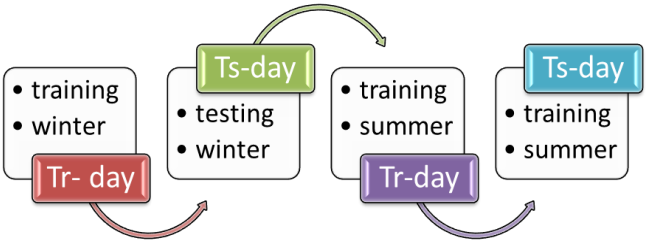


Figure 6. The new wind prediction tool (Tr= training, Ts= testing).

The new approach can be described as a short term prediction tool with a nonconventional way of selecting the training and testing data. The old approach can be described as a conventional long term approach.

Figure 7 compares the results of the old approach with the results of the new one. The first two columns shows the results of the old approach along with the best results from the different input data and training data respectively (shown in the previous RMS results tables) , and the second two columns show the same results but for the new approach. An error of /RMSD=0.0449/ was obtained.

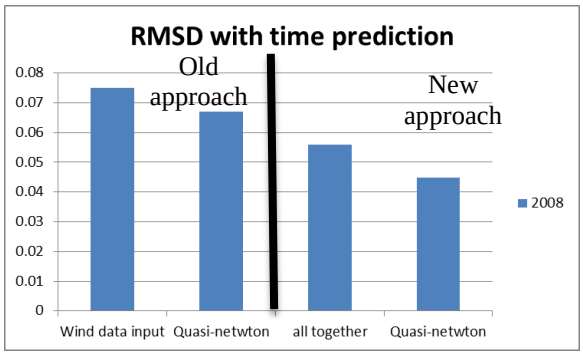


Figure 7. The error reduction due to the usage of the new approach.

Although the results shown in Figure 7 only describe one year (2008), if applied to more than one year as shown in Figure 8 it can be seen that the RMSD for the old approach is better than the new approach. However, since the purpose of this research is for energy calculations and the energy market (in other words, for engineering not meteorological applications -short term prediction is enough in this case-) the approach needs to have a very small value of error. For this reason, the new approach can be considered more effective in this situation as shown in Figure 5 or Figure 9, which show the deviation between the measured and predicted wind speed using the old and new approach respectively.

It should be mentioned that in the European energy market a 4 day wind energy prediction is needed and considered as a short term prediction period. Every grid operator should be able to provide the expected available energy for the next day in order to finish the bidding on the energy soled amount (this process normally takes one day).

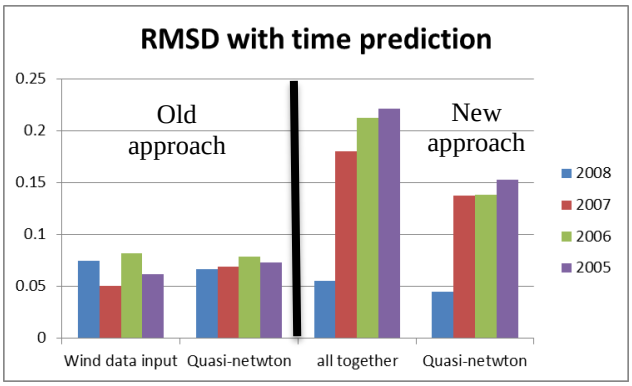


Figure 8. Comparison of the RMSE for the old and new approach for differing years.

Thus the usage of proposed prediction tool has a great influence on the selection of which scenario to work with.

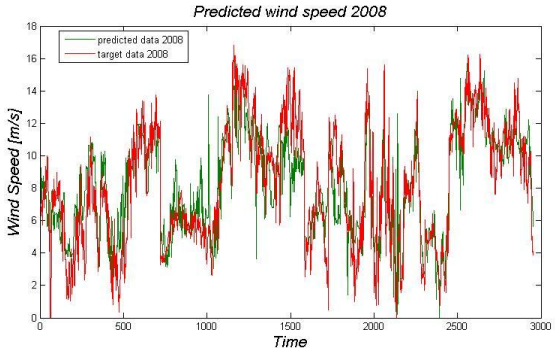


Figure 9. Measured and predicted wind speed for 2008 using the new approach using the preasure, the temperature and the wind speed diraction as an input.

When comparing the different prediction scenario's results (with time shift and without time shift) in Figure 8 and Figure 11, it can be seen that the RMS values of the not time shift scenario are lower than the other one for the whole year range which lead to the conclusion that the used input variables (pressure, temperature) are good enough for getting a reliable output (wind speed). Those results were also compared with the WASP program calculation for the

wind speed with a result of /RMS=0.004/ (as the WASP program uses NWP module in order to calculate the vertical wind speed difference between the measurement mass and the wind park selected location).

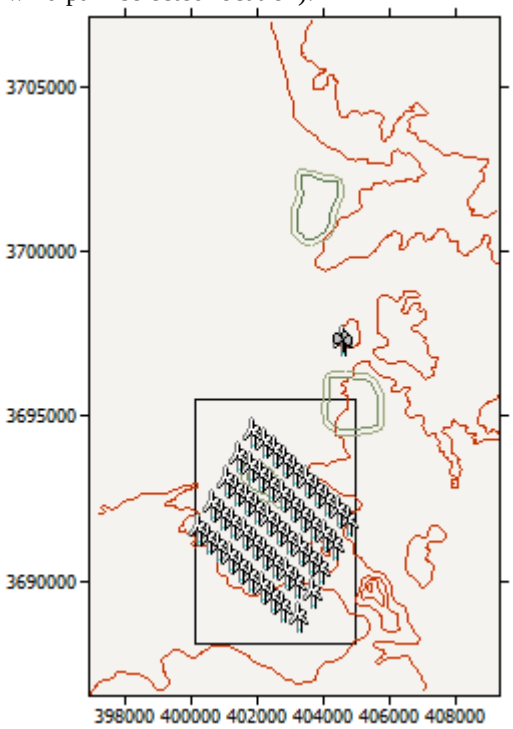


Figure 10. the location of the suggested wind park as designed by WASP program

The difference in the location between the measurement mass and the wind park is clear in Figure 10, which shows the map of wind park location in Syria using the WASP program for calculating the expected energy output from the suggested wind park.

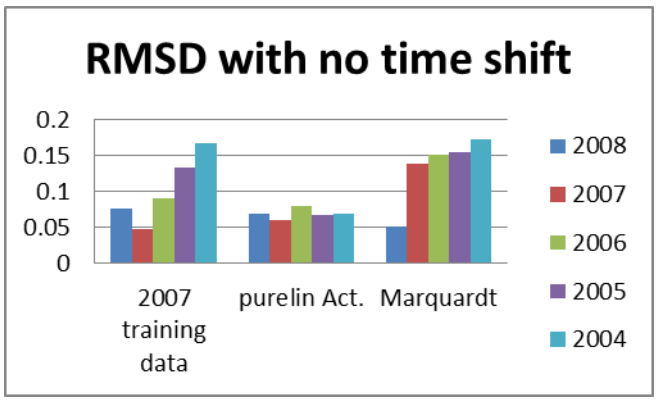


Figure 11. The best errors values that are resulting from using different training fucntion and training data and activation fucntion .

The best training function selection is done by comparing the regression figures of the different training functions, like in Figure 12 where the fit line shows the trend line of training, validation, testing and all. The more the trend line is adjacent to the orange line (Y=T) the better prediction results can be obtained from the ANN prediction tool. The dotted line presents the case where the output results of the ANN tool are the same of with the target data which mean the best case scenario for the prediction tool.

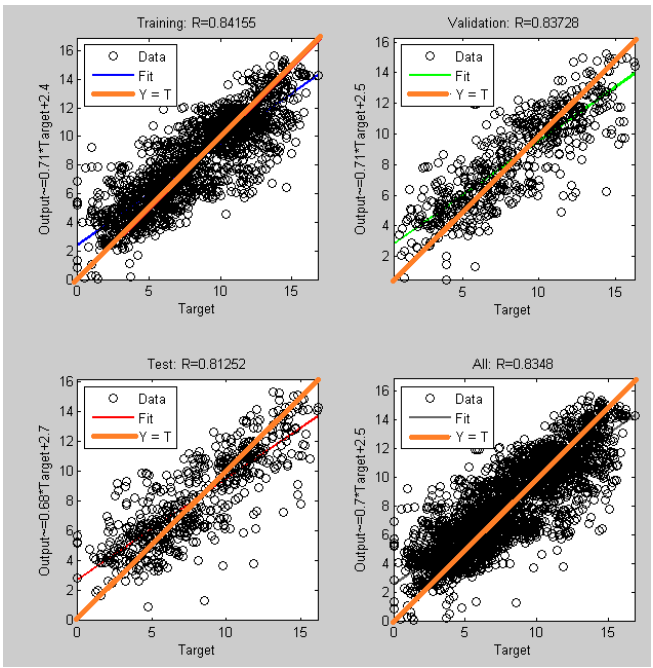


Figure 12. The best errors values that are resulting from using different training fucntion and training data and activation fucntion

Finally it should be mentioned that comparing Figure 9, which shows a comparison between measured wind speed values and the predicted one for the new approach, and Figure 5, which shows the same information for the old approach, for the same time interval (e.g. same moth) will reveal the fact that the new approach give a better output results. And it is also necessary to emphasis that the prediction errors can fall into two main categories:

1- Amplitude errors: This type of error results from the difference between the predicted wind speed value and measured wind speed value for the same point of time (same point on the x-axis).
2- Time shift errors: This type of error results from the difference between two wind speed points with the same value.(different X-axis /time/ values but the same Y-axis /wind speed in m/s/ values).

The second type of errors is more critical compared to the first one. As the results of this prediction will be used after that to determine the available energy that can be produced, which in its turn will be sold in an energy market. Assuming that the first type of errors (Amplitude errors) appears during a real time situation the conserved energy should be able to cover the difference (this is the case in European energy market) yet if the other type of errors accurse in a real time situation it might result in a complete black out as the preserved energy may not be able to cover

difference in energy values (Y-axis /wind speed that results eventually in Energy/ has a big difference in values (Amplitude) at the same time point on the x-axis that is not expected and not taken into consideration during the biding period done earlier) that is not considered.

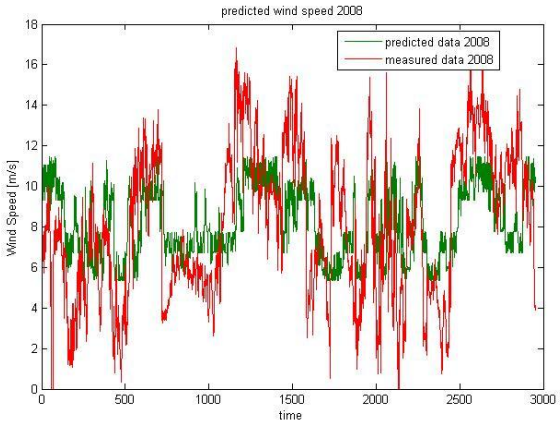


Figure 13. Measured and predicted wind speed for 2008 using the new approach and wind speed as input data

In other words, it is difficult to overcome an unexpected shortage in the energy (Like, in the second error type) than to overcome an expected value of shortage in the energy (Like, in the first error type). Figure 13 is a good example for showing the time shift error, which occur between X=0 and X=500 s. it was expected (predicted) that we will have about Y= 10 m/s wind speed but in realty we got only Y=8 m/s this difference will be tripled in the energy point of view and in its turn will result is some kind of problem to the Grid operator when no enough reserved energy is available to cover this difference.

I. Energy Module

After getting acceptable results as an output from the previously built prediction tool, those results are used as an input for the energy module. Figure 14 shows the suggested energy model, which has the following components:

1. A signal Builder: contains the predicted wind speed data.
2. Lookup Table: contains the power curve data of the used power turbine [14].
3. Integrator: is used to get the output energy from the wind turbine [15].
4. Scope: is used to show the results in Figure 14.
5. Display: is used to show the accumulated value of the energy.

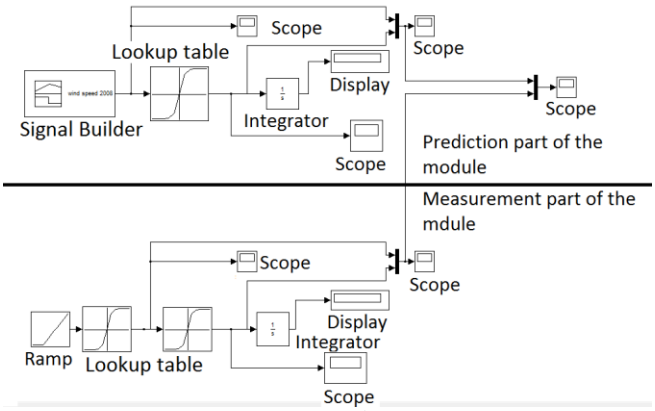


Figure 14. Block diagram of the energy module.

Putting the previous component together give us the energy which can be produced by the used turbine. As the energy is the area under the power curve of an electrical generation unit. The integration of the power function over a certain time period can result with the energy generated in the same time period. The energy calculated in this module results from the integration of the combined wind speed data (one input is the predicted values and the other input is the measured values) and the power curve of the used wind turbine (in order to make sure that only the useful wind speed values will result in the corresponding power).
Vestas Wind System /V90/ was selected as the working wind turbine with a 90 m rotor diameter, 105 m height, and 2 MW power [8]. Figure 15 shows the power curve of this wind turbine.

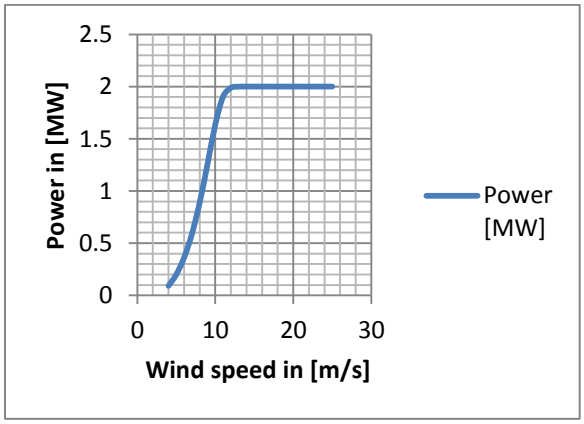


Figure 15. Power curve of V90 wind turbine.

Using all of above information, the energy can be obtained as shown in Figure 16 where the error is tripled due to the relation between the energy and the cubic wind speed. For this reason, the error of the predicted wind speed should be at its minimum with no time deviation errors [16][17].

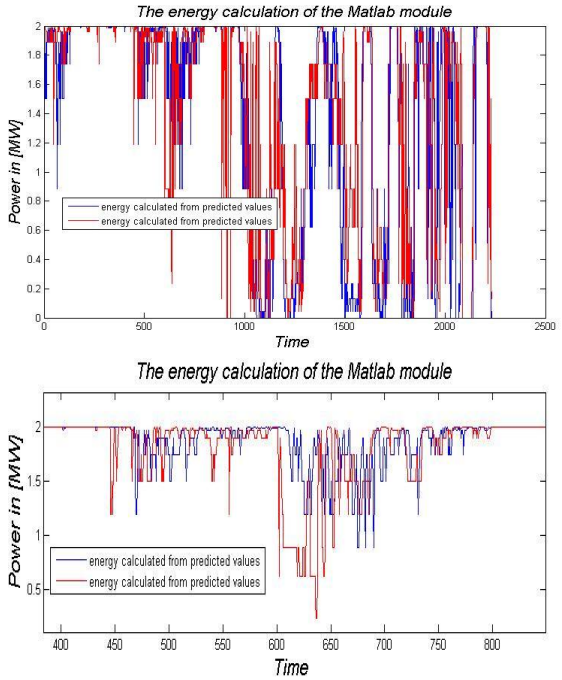


Figure 16. The difference between the energy calculated from the predicted and measured wind speed

It is important to mention here, that integrating the wind turbine curve in the energy module helped to avoid using the yield energy equation from the wind speed (the energy equation [17]) where other parameters like the density of the air is not included.

III. CONCLUSION

As a conclusion of this work it is observed that the prediction errors can be reduced by the usage of same characteristic properties of the predicted wind speed and that in its turns will lead to less errors in the Energy module (the error of the energy is cubical to the wind speed errors due to the cubic relationship between the Energy and the wind speed). More understanding of the data lead to better results in the prediction tool that is why spearman's analysis is an important method of determining the strength of connection between the different data used as input to get the output of the prediction tool. Finally, the short term prediction helps of reducing the errors in the prediction tool and also the work load on the computing device.

From all the information presented in this work it can be concluded that the time shift wind speed prediction tool is used in order to estimate the wind speed changes in the future in the location where the measurement is available. After that the no time shift wind speed prediction tool can be used to transfer these results from the previous tool to the location of interest and the height of the wind turbine's hub. Finally those results can be used as input for the energy module which in its turn will results in the estimated energy that will be generated from the chosen wind turbine in the future (the same time interval as the one predicted by the time shift wind speed prediction tool).

REFERENCES

[1] *A. Alkhatib, S. Heier, and M. Kurt"The development and Implementation of a short Term Prediction Tool Using Artificial Neural Networks" Published for ComputationWorld 2011, IARIA, Rome, Italy ,2011*

[2] *J.T. Houghton, L.G. Meira Filho, B.A. Callander, N. Harris, A. Kattenberg, and K. Maskell " Climate Change 1995, The Science of Climate Change, Contribution of WGI to the Second Assessment Report of the Intergovernmental Panel on Climate Change" Published for the Intergovernmental Panel on Climate Change, Cambridge University Press, 1995, pp. 27-73.*

[3] *Dr. K. Rohrig and R. Jursa "Online-Monitoring and prediction of wind power in German transmission system operation centers" Königstor 59, D-34119, Kassel: IWES, pp. 1, 2002.*

[4] *Dr. M. Lange and Dr. U. Focken" State of the art in wind power prediction in Germany and international developments"Marie-Curie-str.1, D-26129 Oldernburg: Oldenburg uni.,pp. 2-3, 2009.*

[5] *T. Burton, D. Sharpe, N. Jenkins, and E. Bossanyi "Wind energy handbook" London: John Wiley & Sons, Ltd., 2001.*

[6] *M. Kumar" SHORT-TERM LOAD FORECASTING USING ANN TECHNIQUE" Rourkela-769008: National Institute of Technology,pp. 9, 2009.*

[7] *Ministry of Electricity, "Request for qualification (RFQ)" Damascus: Syrian Arab Republic, 2009.*

[8] *Decon "Pre-feasibility study AL hijana. Damascus" Syrian Energy Center, Damascus, Syria, 2005.*

[9] *S. Mathew" Wind Energy Fundamentals, Resource Analysis and Economics" Springer, 2007.*

[10] *S.Chand. Managerial statistcs. AMIT ARORA, 2009.*

[11] *K.Sreelakshmi and P. Ramakanthkumar" Neural Networks for short term wind speed prediction" World Academy of Science ,Engineering and technology, pp.724, 42 2008.*

[12] *J. R.Rabunal and J.Dorado "Artificial neural network in real-life application" London: Idea Group Publishing, 2006.*

[13] *Wikipedia®. (2010). http://en.wikipedia.org/wiki/Root_mean_square_deviation.*

Retrieved 06,12, 2012, from www.wikipedia.com: www.wikipedia.com

[14] K.Sreelakshmi, P. Ramakanthkumar" Neural Networks for short term wind speed prediction" World Academy of Science ,Engineering and technology, pp.724, 42 2008.

[15] A. Kumar Mishra and L. Ramesh "Application of Neural networks in wind power (Generation) Prediction" IEEE, pp. 3, 2008.

[16] M. Hayashi and B. Kermanshahi "Application Artificial neural network for wind speed and determination of wind power generation output" Nakamachi, Koganei-shi,Tokyo 184-8588, Japan: Tokyo Uni.,pp.2-3, 2009.

[17] M.C. Mabel and E.Fernandez" Estimation of Energy Yield from Wind Farms Using Artificial Neural Networks" IEEE, pp. 3, 2009.

[18] A. Alkhatib "Developing a Wind speed prediction tool using artificial neural networks and designing Wind Park in Syria using WASP software" Kassel university, pp.6, 2010.

Performance and Design Guidelines for PPETP, a Peer-to-Peer Overlay Multicast Protocol for Multimedia Streaming

Riccardo Bernardini, Roberto Cesco Fabbro, Roberto Rinaldo
DIEGM – Università di Udine, Via delle Scienze 208, Udine, Italy
{riccardo.bernardini,roberto.cesco,roberto.rinaldo}@uniud.it

***Abstract*—One major issue in multimedia streaming over the Internet is the large bandwidth that is required to serve good quality content to a large audience. In this paper we describe PPETP, a peer-to-peer protocol for efficient multimedia streaming to large user communities. The performance of the protocol (such as the robustness of the protocol with respect to packet losses and churn) are quantitatively analyzed and guidelines for designing peer-to-peer streaming systems based on the described protocol are given.**

***Keywords*-Data transmission; multimedia streaming; overlay multicast; peer-to-peer network; push networks**

I. Introduction

A problem that is currently attracting attention in the research community is the problem of streaming live content to a large number of nodes. The main issue to be solved is due to the amount of upload bandwidth required to the server that, unless multicast is used, is equal to the bandwidth required by a single viewer (some Mb/s for DVD quality) multiplied by the number of viewers (that can be very large, for example, it is reported that in 2009 the average number of viewers per F1 race was approximately $6 \cdot 10^8$). Multicast could be a possible solution, but it has drawbacks too. For example, multicast across different Autonomous System (AS) has several issues, both technical and administrative ones.

An approach that recently attracted interest in the research community is the use of peer-to-peer (P2P) solutions as described in [2] to [15]. With the P2P approach each viewer re-sends the received data to other users, implementing what could be roughly defined as an overlay multicast protocol where each user is also a router. Ideally, if each user retransmitted the video to another user, the server would just need to "feed" a handful of nodes and the network would take care of itself.

Unfortunately, the application of the P2P paradigm to multimedia streaming has some difficulties. For example, depending on the media type and quality, residential users could have enough download bandwidth to receive the stream, but not enough upload bandwidth to retransmit it. This problem is known as the *asymmetric bandwidth* problem.

Another important issue with P2P networks of residential nodes is due to the *churn* of the network, that is, the "turbulence" induced by users joining and leaving the network at random. In particular, if a user suddenly leaves the network, other users could be left without data for a long time.

Moreover, P2P networks have several security issues [16]. Here we simply cite the *stream poisoning attack* where a node sends incorrect packets that cause an incorrect decoding and are propagated to the whole network by the P2P mechanism.

This article is the extension of [1] and describes the *Peer-to-Peer Epi-Transport Protocol* (PPETP), a peer-to-peer protocol developed at the University of Udine and hosted as part of the project *Corallo* on *SourceForge*. While the description given in [1] was more of a qualitative nature, describing the main feature of PPETP, without going into quantitative details, this paper aims to give a more analytical description of the feature of PPETP, with the objective of giving guidelines for designing networks based on PPETP. For the sake of completeness, in this paper we briefly summarize some results published elsewhere, taking care of marking explicitly the parts taken from other works.

This paper is organized as follows. Section III gives a qualitative overview of PPETP and introduces some jargon; Section IV introduces the concept of reduction procedure, a key concept in PPETP; Section V analyzes some features of PPETP that derives from the use of reduction functions; Section VI analyzes the packet loss probability experienced by the nodes in a PPETP network; Section VII gives some quantitative results about the robustness of PPETP against churn; Section VIII gives some guidelines for designing networks based on PPETP; Section IX presents the conclusions.

II. State of the art

The first P2P streaming networks had a tree structure, inspired by IP multicast. For example, ZIGZAG [17], built a multicast tree for media streaming at the application layer. This structure is, however, quite weak, mainly because, differently from IP-layer multicast, the P2P tree is built upon peers that may join and leave at any time. This is a serious issue, since a departing peer disconnects all its descendants from the source.

Multiple tree-based overlay architectures, such as SplitStream [10], CoopNet [18] and ChunkySpread [19], are proposed to mitigate the issues in single-tree architectures. Compared to architectures based on a single tree, architectures based on multiple trees are more resilient to peer departures

and failures. In addition, they can more efficiently use the uploading link capacity of each peer, since each peer works as an interior node in at least one tree. However, they achieve these benefits at the cost of more complicated architectures and media encoding methods.

Recently, many proposed P2P streaming networks, such as CoolStreaming [20], AnySee [21], PRIME [22], and DagStream [23] use mesh networks. In an *unstructured* mesh network (e.g., PRIME, CoolStreaming) a peer connects with a large number of randomly selected peers, with the purpose of providing more neighbors and more diverse paths. In a *structured* mesh network peers are typically grouped into clusters, in order to reduce the propagation delay of packets. However, such a locality-aware network may suffer from the shared bottleneck problem, where the media quality of all peers in a cluster strongly depends on the available bandwidth at a shared bottleneck. For these reasons, a locality-aware approach constructs a mesh with some special structure in order to achieve good network connectivity. Another approach used in some P2P streaming systems is the use of rateless codes. Examples of this approach are *rStream* [24] and *ToroVerde* [25].

Most of the currently available P2P streaming systems are mesh-based and employ ideas similar to the ones used in P2P file sharing systems such as BitTorrent. In this type of P2P streaming systems, the content is split into sections (called *chunks*, so this type of systems is sometimes referred to as *chunked* P2P systems). In a typical chunked system a node queries and requests them content chunks. A serious issue in chunk-based P2P systems is that they have very long start-up times due to the fact that in order to use a chunk-based system with live material, it is necessary to do some buffering.

III. Overview of PPETP

The goal of this section is to give a brief overview of the structure of PPETP and to introduce some PPETP jargon that will be used in the following. For the sake of brevity, many details will be omitted. A more detailed description can be found in the Internet Draft [26].

PPETP can be considered as a multicast overlay protocol based on a P2P approach that sends data and commands over a non necessarily reliable protocol (e.g., UDP). The type of multicast done over a PPETP network can be both Any Source Multicast (ASM) or Source Specific Multicast (SSM); in this paper we will consider, for the sake of concreteness, the SSM case only, the adaptations for the ASM being obvious. In the SSM case the origin of the content will be called *origin server*.

Since each node streams autonomously to other nodes, a PPETP network can be considered a *push* network. If node A receives data from node B, we will say that A is a *lower peer* of B and that B is an *upper peer* of A (therefore, data flows from top to bottom). PPETP does not mandate any specific network topology, the only constraint being that each node has a minimum number of upper peers.

Fig. 1 shows an example of a possible PPETP network for multimedia streaming with three upper peers per node. Each

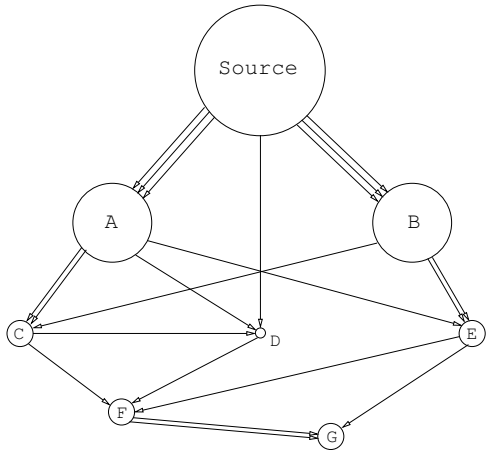


Figure 1. Example of a PPETP network for multimedia streaming.

arrow represents a stream, each circle represents a node and the node available upload bandwidth is represented by the circle size. For example in Fig. 1, node A (an upper peer of C, D and E) sends to C *two* different streams. Note also that the source "feeds" directly nodes A and B by sending them three different streams. Other examples of possible topologies for a PPETP network are shown in Fig. 2. Note that not only tree-structured networks are possible with PPETP.

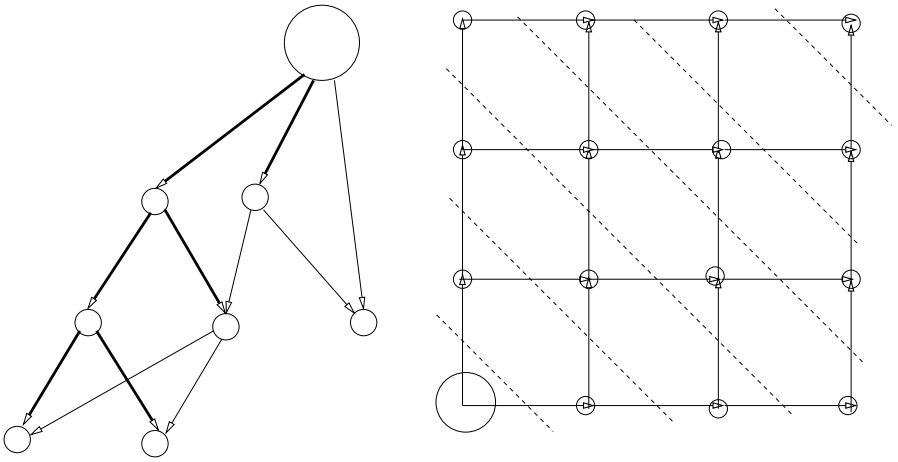
Remark III.1 (What PPETP is not)
A P2P streaming system is a complex piece of software that must take care of several things: transferring data, finding new peers, tracking content and so on. We would like to emphasize here that PPETP is designed to take care only of the efficient data distribution; other important aspects of the P2P streaming application (e.g., building the network) are demanded to extra-PPETP means. This is similar to what happens with TCP: the standard specifies how data is carried from a host to another, but does not specify, for example, how one host finds the other, this being handled by protocols such as DNS.

IV. Data reduction procedures

A key characteristic of PPETP is that, in order to solve the asymmetric bandwidth problem, every node does not send to its lower peers the whole content stream, but a *reduced stream* that requires less bandwidth. The reduced stream is obtained by processing each packet in the content stream with a suitable *reduction function*.

Informally, a reduction function is a function that maps the set of bit-strings (i.e., the set of packets) into itself, with the property that the result is shorter (actually, R times shorter) and that one can recover the original bit-string when at least R reduced versions if the original bit-string are known.

We will represent mathematically packets as elements of $\mathfrak{B} \stackrel{\text{def}}{=} \{0,1\}^*$, the set of all finite bit-strings. With this position, reduction functions are represented by functions mapping packets into packets, that is, $\mathfrak{B}$ in $\mathfrak{B}$. In the following will be convenient to have a notation that allows to represent compactly a vector of reduction functions.



(a) (b)

Figure 2. Examples of network topologies compatible with PPETP (a) parallel trees and (b) onion skin. The dashed lines mark the stratum boundaries

Notation 1. *Let* $\mathfrak{B} \stackrel{def}{=} \{0,1\}^*$ *be the set of all finite bit-string, let* J *be a finite set and let*

$$\mathfrak{R} = \{f_a : \mathfrak{B} \to \mathfrak{B}, a \in J\} \tag{1}$$

a set of functions mapping bit-strings in bit-strings and indexed by J*. For every n-ple of indexes* $\mathbf{a} = (a_1, a_2, \cdots, a_n) \in J^n$ *we will denote with* $g_{\mathbf{a}} : \mathfrak{B} \to \mathfrak{B}^n$ *the function*

$$g_{\mathbf{a}}(x) \stackrel{def}{=} [f_{a_1}(x), \ldots, f_{a_R}(x)] \tag{2}$$

The key property of reduction functions is what we call R-reconstruction property, that is, the possibility of recovering a packet when at least R different reduced versions of it are known. This idea is made precise in Definition 1.

Definition 1. *Set* $\mathfrak{R}$ *in (1) is said* to satisfy the R-reconstruction property *if for every* $\mathbf{a} \in J^R$ *the corresponding function* $g_{\mathbf{a}}$ *(defined as in (2)) is injective.*

We will say that the R-reconstruction property is satisfied tightly *by* $\mathfrak{R}$ *if for every* $\mathbf{a} \in J^{R-1}$ *the corresponding function* $g_{\mathbf{a}}$ *is* ***not*** *injective.*

If a set $\mathfrak{R}$ *satisfies tightly the R-reconstruction property we will also say that* $\mathfrak{R}$ is a set of reduction functions.

In the following the elements of J used to index the functions in $\mathfrak{R}$ will be called *reduction parameters*.

Remark IV.1

As anticipated, Definition 1 is a formal way to say that if (1) is a set of reduction functions, then it must be possible to recover $x \in \mathfrak{B}$ from the knowledge of any R-pla of reduced versions $f_{a_1}(x)$, …, $f_{a_R}(x)$. The condition of tight reconstruction helps in avoiding pathological cases that satisfy the R-reconstruction property but operates no reduction at all (e.g., when all the functions in $\mathfrak{R}$ are the identity function).

Since the idea of a set of reduction functions can seem a bit abstract and it can not be clear if a set of reduction functions exists at all, it is worth to give an example based on Reed-Solomon codes and used in PPETP with the name of *Vandermonde reduction profile* [26].

Example IV.1

Let $d > 0$ be an integer and let $\mathbb{F}_{2^d}$ denote the Galois field with 2^d elements. Galois field $\mathbb{F}_{2^d}$ will be used both as the reduction parameters set J and for computation.

The function $f_c : \mathfrak{B} \to \mathfrak{B}$ associated with reduction parameter $c \in \mathbb{F}_{2^d}$ is computed as follows. Let x be the argument of f_c, let

$$\mathbf{r}_c \stackrel{\text{def}}{=} \begin{bmatrix} 1 & c & c^2 & \cdots & c^{R-1} \end{bmatrix} \tag{3}$$

be the R-dimensional row vector in $\mathbb{F}_{2^d}^R$ whose components are powers of c and let $\mathbf{C}_x$ be the R-row matrix with entries in $\mathbb{F}_{2^d}$ by considering every d-ple of bits of x as an element of $\mathbb{F}_{2^d}$ (if the number of bits of the packet is not an integer multiple of dR, the packet is first suitably padded, see [26] for details). The value of $f_c(x)$ is

$$f_c(x) = \mathbf{r}_c \mathbf{C}_x \tag{4}$$

(Note that in (4) we did a notational abuse, since the value of $f_c(x)$ should be a bit-string, while the right hand side of (4) is a vector of elements of $\mathbb{F}_{2^d}$.)

In order to see that the set of functions f_c is actually a set of reduction functions with reduction factor R, observe that from the knowledge of $f_{c_1}(x), \ldots, f_{c_R}(x)$ one can recover $\mathbf{C}$ by solving the linear system

$$\begin{bmatrix} f_{c_1}(x) \\ f_{c_2}(x) \\ \vdots \\ f_{c_R}(x) \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{c_1} \\ \mathbf{r}_{c_2} \\ \vdots \\ \mathbf{r}_{c_R} \end{bmatrix} \mathbf{C}_x = \underbrace{\begin{bmatrix} 1 & c_1 & \cdots & c_1^{R-1} \\ 1 & c_2 & \cdots & c_2^{R-1} \\ \vdots & \vdots & & \vdots \\ 1 & c_R & \cdots & c_R^{R-1} \end{bmatrix}}_{\mathbf{R}} \mathbf{C}_x \tag{5}$$

Since matrix $\mathbf{R}$ in (5) is a Vandermonde matrix, it is invertible (and (5) has a solution) as soon as all the c_k are different.

Remark IV.2

Although the approach in Example IV.1 is well known, many other sets of reduction functions can be constructed; see, for example, [27].

By exploiting the idea of reduction functions, nodes of a PPETP network propagate the streamed data as follows

- At start-up
  1) Each node chooses one or more reduction parameters a_1, a_2, …. Although the parameter(s) can

be imposed by an external network coordinator, if the parameter space is large enough, the nodes can choose them at random, simplifying the network management. See Section V-B.

2) Contact $N_{up} \geq R$ upper peers. Each upper peer will communicate to the node its reduction parameter before starting streaming.

- For every content packet
  1) Wait for at least R different reduced packets
  2) After receiving at least R reduced packets, recover the content packet
  3) Move the content packet to the application level
  4) Reduce the content packet using the chosen reduction parameters $a_1, a_2, \ldots$.
  5) Send the computed reduced packets to your lower peers

Note that if, because of packet losses, the node receives less than R reduced versions of the content packet, the node can still help in propagating the information by forwarding to its lower peers the reduced data received from the upper peers. We call this (almost obvious) strategy *fragment propagation* and it will be shown in the following that, despite of its simplicity, it is important for system performance.

Remark IV.3

The strategy of fragment propagation can help in solving a problem that is intrinsic to the P2P network of residential nodes. If a residential node has the upload bandwidth smaller than the content bandwidth, it introduces a "bandwidth debt" since it cannot compensate the consumed download bandwidth with an equal upload bandwidth. Such a debt must be covered by other nodes such as super-nodes or by the origin server. Since the bandwidth debt can be expected to grow linearly with the number of residential nodes, this problem can challenge the scalability of the P2P network.

The use of reduction functions and fragment propagation can help in counteracting this problem. A residential node can act as a "repeater" (maybe in exchange of some improved service) by simply joining the network and contacting only one upper peer, but accepting $N_{low} > 1$ lower peers. Automatically, because of the fragment propagation policy, it will forward to its lower peers the packets received from the upper peer. The overall effect is a "bandwidth gain" equal to $N_{low} - 1 > 0$ reduced streams that compensates the debt of other nodes.

A. Data puncturing

In the case of high quality content (that requires a large bandwidth) and a network with low upload bandwidth nodes, it could happen that the required reduction factor R is too large (the drawbacks of a too large reduction factor will become clear in the following). In this case PPETP can reduce the upload bandwidth by *puncturing* the stream of fragments. Puncturing can be both *probabilistic* or *deterministic*. In the former case, the packets to be sent are chosen randomly with a given probability, in the latter case the packets are chosen according to a pattern that is periodically repeated (e.g., send only the even packets). For example, Fig. 3a shows a node with five upper peers, where two peers (nodes C and D) apply a 1:2 puncturing to the data stream, node C sending only even packets and D only odd ones. It is clear that the scheme of Fig. 3a is approximately equivalent to the scheme of Fig. 3b, where the two puncturing nodes are "merged" in a "virtual" no puncturing node.

V. Properties of Data Reduction

In this section we discuss few interesting properties due to the use of data reduction schemes. It is interesting to observe that the properties discussed in this and the following sections do not depend on the actual functions f_a, but only on their property of being reduction functions. Therefore, the properties discussed here hold not only for the Vandermonde scheme of Example IV.1, but also for any other reduction scheme that enjoys the R-reduction property.

A. Solution to the asymmetric bandwidth problem

This property is almost obvious, but it is included here for the sake of completeness. Since the bit-string associated with the size of the reduced packet is R smaller than the size of the content packet, the bandwidth required by the reduced stream is R times smaller. By choosing R large enough, even the nodes with small upload bandwidth can contribute to propagating the content.

B. Distributed assignment of the reduction function

A first interesting property is that if the set of reduction parameters J is large enough, each node can choose its parameter at random, since the probability of having two nodes with the same reduction parameter is negligible. This simplifies the assignation of the reduction parameters to the nodes, since a central authority is not required.

Remark V.1

Note that there is no computational overhead in choosing $|J|$ large, since J represents the "pool" from which reduction functions are chosen, that can be much larger than the number of generated reduction packets. For example, in the PPETP specs [26] $|J| = 2^{32}$ although R can be expected to be at most ≈ 20.

In order to be more quantitative, let $S = |J|$ be the cardinality of the reduction parameter set J, let N_{up} be the number of upper peers of a given node and let a_k be the reduction parameter of the k-th upper peer, $k = 1, \ldots, N_{up}$. The node is able to recover the content stream if and only if there are least R different values of a_k. Fig. 4 shows the probability P_{fail} that this does not happen as a function of S, the reduction factor R and the ratio $\rho = N_{up}/R$ (interpretable as a redundancy factor). It is clear from Fig. 4 that one can achieve negligible P_{fail} by using J of reasonable size and small redundancy factors. The curves in Fig. 4 have been obtained by means of the numerical procedure described in Appendix A where it is also shown that P_{fail} goes to zero with N_{up} as

$$[(R-1)/S]^{N_{up}} = [(R-1)/S]^{\rho R} \quad (6)$$

By means of standard analysis techniques, it is possible to show that (6), as a function of R, has a single minimum at $R \approx 1 + S/e$. Since S is typically very large (for example, $S = 2^{16}$ if $\mathbb{F}_{2^{16}}$ is used in the reduction scheme of Example IV.1),

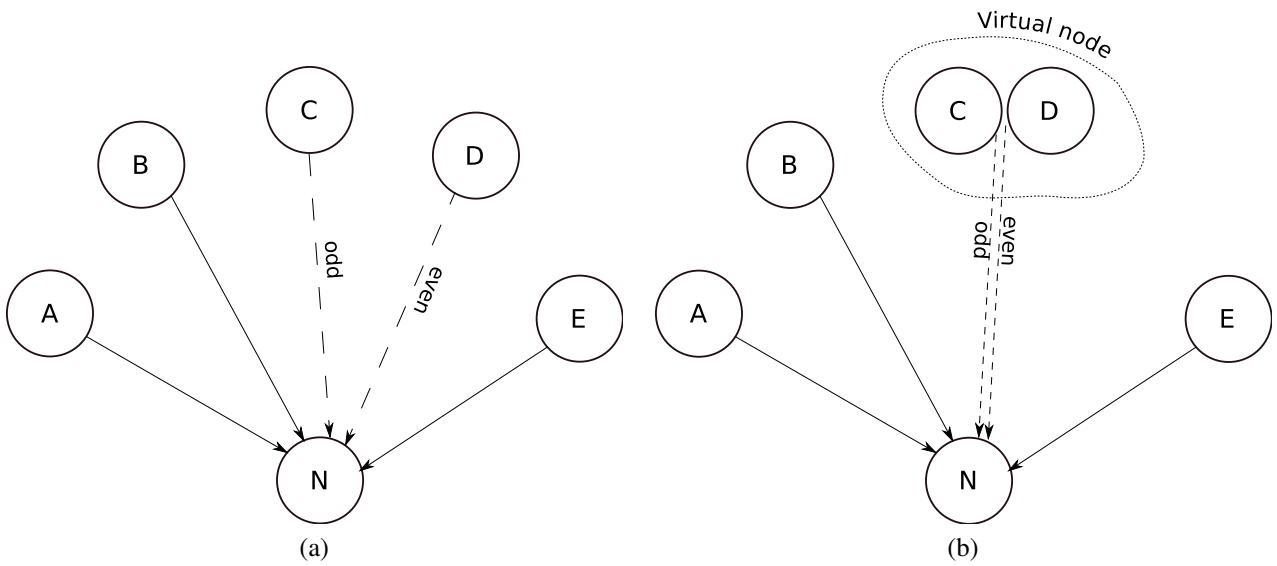


Figure 3. (a) Node N has five upper peers, with two upper peers (C and D) applying a puncturing 1:2. (b) Network equivalent to the network in (a) with a "virtual" upper peer obtained by merging nodes C and D.

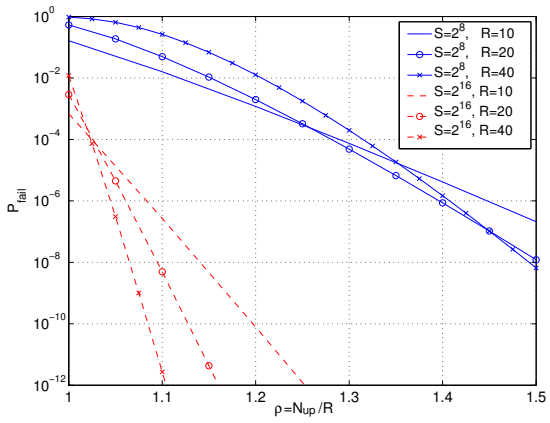


Figure 4. Probability P_{fail} of having less than R different reduction parameters out of N_{up} vs. redundancy ratio $\rho = N_{\text{up}}/R$ for different values of $S = |J|$ and R.

it follows that (6) decreases monotonically with R in every case of practical interest. Moreover, since the minimum of (6) is very low when S is large, it follows that, fixed ρ, one can make P_{fail} as small as desired by taking R large enough.

C. Robustness with respect to packet loss

The scheme is inherently robust with respect to packet losses, typically due to peer departures and network congestion. Actually, a node that contacts $N_{\text{up}} > R$ upper peers will be able to recover the transmitted data as long as not more than $N_{\text{up}} - R$ packets are lost. The effect of packet losses is discussed in more detail in Section VI.

D. Robustness with respect to churn

An important problem in P2P streaming network is that a node can leave at anytime, leaving its lower peers without data. The fact that in a network made of residential users one can expect a high *churn* is one of the reasons that support the use of mesh-based chunky solutions. However, in PPETP the same redundancy that protects against packet losses protects also against the effect of churn. This is discussed in more detail in Section VII.

E. Robustness to stream poisoning

As said above, a possible attack is the injection of "garbage packets." The use of reduction functions offers a simple way to counteract such a threat. Actually, it suffices to contact $N_{\text{up}} > R$ upper peers, use R reduced versions to recover the original data and check that the result is coherent with the remaining reduced packets. More precisely, let u_k denote the packet received by the k-th upper peer, let a_k denote the corresponding reduction parameter and let $\mathbf{a} = [a_1, \ldots, a_R]$ and suppose that at most $N_{\text{up}} - R - 1$ packets u_k can be corrupted.

In order to recover x safely, the node chooses R reduced packets u_{k_1}, ..., u_{k_R}, recovers the content packet as $x = g_{\mathbf{a}}^{-1}(u_{k_1}, \ldots, u_{k_R})$ and checks the result by verifying that the following equalities hold

$$u_{k_\ell} = f_{\alpha_{k_\ell}}(x), \qquad \ell = R+1, \ldots, N_{\text{up}} \tag{7}$$

The following cases may happen

1) All the equalities (7) are verified. In this case x is correctly recovered and every peer sent us a correct packet.
2) Some of the equalities (7) are verified, but not all. In this case x is still correctly recovered, but the peers corresponding to the non verified equalities sent us a corrupted packet.
3) All the equalities (7) are not satisfied. Since we supposed that at most $N_{\text{up}} - R - 1$ packets u_k can be corrupted, this can happen only if we used a corrupted packet in recovering x. In this case we can choose a different set

of u_k and try to recover x again. If only one corrupted packet is present, the expected number of trials before x is recovered is $N_{\text{up}}/(N_{\text{up}}-R)$. If the recovering of x has already been attempted too many times, one can declare the packet lost and let the application conceal the loss.

The procedure described above can be considered as a generalization of the use of error correcting codes. However, in this case data are not corrupted by a noisy channel, but by a malicious attacker that could, in principle, send carefully crafted data that cause the node to recover garbage data that nevertheless passes the test above. Therefore, the question is: can an attacker craft a corrupted packet $\hat{u}_k$ that produces a wrong packet $\hat{x} \neq x$ that satisfies the test above? What about a coordinated attack by A peers? We are going to show that if $N_{\text{up}} \geq R+A$, the system is immune from a coordinated attack by A peers.

To be more precise, let x be the original content packet and suppose a given node receives data from $N_{\text{up}} > R$ peers. Let $u_k = f_{a_k}(x)$ be the reduced packet that the node *should* receive from peer k and let $\hat{u}_k$ be the actual received packet. Since we are supposing that no more than A peers will try a coordinated attack, we know that there are at most A values of k such that $\hat{u}_k \neq u_k$.

The packet recovered by the node is $\hat{x} = g_{\mathbf{a}}(\hat{u}_1,\ldots,\hat{u}_R)$ and the node accepts it if all the following equalities hold

$$f_{a_\ell}(\hat{x}) = \hat{u}_\ell, \qquad \ell = R+1,\ldots,N_{\text{up}}. \tag{8}$$

We can say that *the attack succeeds if $\hat{x} \neq x$ and the node accepts $\hat{x}$*. The following theorem shows that a coordinated attack by A peers fails if $N_{\text{up}} \geq A+R$.

Theorem 1. *Let $x \in \mathfrak{B}$, let $a_1,\ldots,a_{R+A} \in J$, and let $u_k = f_{a_k}(x)$, $k = 1,\ldots,R+A$. Let $\hat{u}_k \in \mathfrak{B}$, $k = 1,\ldots,R+A$ be such that $\hat{u}_k \neq u_k$ for at most A values of k. Let $\mathbf{a} = [a_1,\ldots,a_R]$ and define*

$$\hat{x} \stackrel{def}{=} g_{\mathbf{a}}^{-1}(\hat{u}_1,\ldots,\hat{u}_R) \tag{9}$$

The following equalities hold

$$f_{a_\ell}(\hat{x}) = \hat{u}_\ell, \qquad \ell = R+1,\ldots,R+A \tag{10}$$

if and only if $x = \hat{x}$.

Proof: As a first step, we show that $\hat{x}$ satisfies $f_{a_k}(\hat{x}) = \hat{u}_k$ for all the $k = 1,\ldots,R+A$. Indeed, such an equality is satisfied for $k \leq R$ because of definition (9) and it is satisfied for $k > R$ because of (10). By hypothesis, there are at least R integers $n_1, n_2, \ldots, n_R \in \{1,\ldots,R+A\}$ such that $u_{n_k} = \hat{u}_{n_k}$. Let $\mathbf{a} = [a_{n_1},\ldots,a_{n_R}]$ and observe that

$$\begin{aligned} g_{\mathbf{a}}(\hat{x}) &= [f_{a_{n_1}}(\hat{x}),\ldots,f_{a_{n_R}}(\hat{x})] \\ &= [\hat{u}_{n_1},\ldots,\hat{u}_{n_R}] = [u_{n_1},\ldots,u_{n_R}] \\ &= [f_{a_{n_1}}(x),\ldots,f_{a_{n_R}}(x)] = g_{\mathbf{a}}(x) \end{aligned} \tag{11}$$

By Definition 1, (11) holds only if $x = \hat{x}$. ■

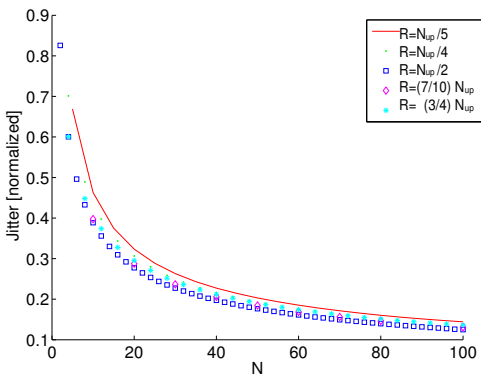


Figure 5. Jitter as a function of N (peer delays distributed as Gaussian with unit variance)

F. Jitter reduction

A nice side effect of the use of network coding in PPETP is, as reported in [28], the reduction of the jitter observed by the node. Intuitively, this happens because the time when a content packet is recovered is the time necessary for the arrival of the R fastest packets out of N_{up}. Fig. 5, taken from [28], shows the theoretical prediction of the jitter (i.e., the standard deviation of the reconstruction time), as a function of R and N_{up}, when the delays are Gaussian with variance σ^2. The values on the vertical axis are measured in units of σ. Note that the jitter decays as $1/\sqrt{N_{\text{up}}}$ [29]. This behavior was also verified experimentally [28].

G. Computational cost

It is convenient to analyze briefly the cost of the computation due to the reconstruction and reduction with the Vandermonde profile, in order to get an estimate of how that cost depends on the design parameters (R, d and N_{up}).

Let Q be the size (in bits) of a content packet. If we work with the Galois field $\mathbb{F}_{2^d}$, the matrix corresponding to the packet will have Q/d entries organized in R rows and $Q/(dR)$ columns (for the sake of notational simplicity, we are supposing that Q is an integer multiple of dR). Let $C_+(d)$ and $C_\times(d)$ be the "cost" associated with, respectively, a sum and a product in $\mathbb{F}_{2^d}$. As the unit of measure of the cost, we will take the time required to do a 32-bit XOR that corresponds to a sum in $\mathbb{F}_{2^{32}}$ and that is implemented with a single instruction on modern microprocessors.

The reconstruction step requires a product between a $R \times R$ matrix (the inverse of the Vandermonde matrix) and the $R \times Q/(dR)$ matrix obtained by stacking the row vectors corresponding to the reduced packets. Such a matrix product requires

$$R \cdot R \cdot Q/(dR) = RQ/d \qquad \text{products} \tag{12a}$$

$$R \cdot (R-1) \cdot Q/(dR) \approx RQ/d \qquad \text{sums} \tag{12b}$$

The reduction step requires the product of the $1 \times R$ reduction vector by the $R \times Q/(dR)$ matrix corresponding to the content

packet. This requires

$$1 \cdot R \cdot Q/(dR) = Q/d \qquad \text{products} \qquad (13a)$$

$$1 \cdot (R-1) \cdot Q/(dR) \approx Q/d \qquad \text{sums} \qquad (13b)$$

Therefore, the overall computational cost per a Q-bit packet is

$$\frac{C_+(d)+C_\times(d)}{d}(1+R)Q = \overline{C}(1+R)Q \qquad (14)$$

where $\overline{C} = (C_+(d)+C_\times(d))/d$ can be interpreted as a "computational cost per bit" due to the operations on the Galois field. If B is the content bit-rate in bit/s, we need to process a packet every Q/B seconds, so that we have a computational load equivalent to

$$\frac{\overline{C}(1+R)Q}{Q/B} = \overline{C}(1+R)B \qquad \text{32-bit XOR/s} \qquad (15)$$

Note that the computational cost grows linearly with the reduction factor.

1) Cost of the operations in $\mathbb{F}_{2^d}$: It is clear that the term $\overline{C} = (C_+(d)+C_\times(d))/d$ depends on the algorithm used for implementing the Galois operations and on the specific architecture. Nevertheless, in order to have a grasp on the dependence of this term on d, we carried out few experiments.

We considered the following possible implementations for a product in $\mathbb{F}_{2^d}$

Long product
: The product in $\mathbb{F}_{2^d}$ is done with an algorithm similar to the integer product algorithm.

Logarithm table
: If 2^d is not too large (say, up to $d = 16$), one can exploit the possibility of defining a logarithm in $\mathbb{F}_{2^d}$ and do the product using a "logarithm table."

Pythagorean table
: If d is quite small (say, up to $d = 8$), one can do the product using a Pythagorean table that stores the product for every possible pair of values.

Kronecker via Pythagorean table
: If d is very small, (e.g., $d = 4$) one can use the Pythagorean table approach to compute more than one product at once. For example, if $d = 4$, $\mathbf{a} = [a_1, a_2] \in \mathbb{F}_{2^4}^2$ and $\mathbf{b} = [b_1, b_2] \in \mathbb{F}_{2^4}^2$, one can compute the Kronecker product $\mathbf{a} \otimes \mathbf{b} = [a_1 b_1, a_2 b_1, a_1 b_2, a_2 b_2]$ by concatenating a_1, a_2, b_1 and b_2 and using the resulting 16-bit index to access a look-up table with the entries of $\mathbf{a} \otimes \mathbf{b}$. The cost of this approach is comparable with the cost of the Pythagorean table approach, but it allows to compute four products at once. An example of this approach can be seen in Fig. 10.a (in assembler) and in Fig. 11 (in C) in Appendix A.

We implemented (in Assembler, in order to avoid the influence of compiler optimizations) the product algorithms described above for $d = 4, 8, 16$ and 32. The source code (with the syntax of the GNU assembler [30] of the implemented procedures is reported in Appendix A, Fig. 10. The time required by those procedure has been measured by means of the `RDTSC` (ReaD Time Stamp Counter), an instruction of x86 processors that allows to obtain the value of the Time Stamp Counter, a 64-bit register increased at each clock cycle. [31]. Note that the programs in Fig. 10 do not include, for the sake of space, side-code such as parameter handling code. The complete set of sources is available, upon request, from the author.

Table I
APPROXIMATE RELATIVE COMPUTATIONAL COST $C_\times(d)$ OF THE PRODUCTS AND COST PER BIT $\overline{C}$ IN DIFFERENT GALOIS FIELD. THE UNITARY RELATIVE COST IS A 32-BIT XOR.

Field	$C_\times(d)$	Cost per bit $\overline{C}$	Memory	Notes
$\mathbb{F}_{2^4}$	0.5	1.5/4 = 0.375	128 K	Kronecker
$\mathbb{F}_{2^8}$	1	2/8 = 0.250	64 K	Pythagorean table
$\mathbb{F}_{2^{16}}$	3.5	4.5/16 = 0.281	256 K	Logarithm table
$\mathbb{F}_{2^{32}}$	20	21/32 = 0.656	0	Long product

The results of such measurements can be seen in Table I that shows the relative complexity of the product in $\mathbb{F}_{2^d}$ and the corresponding "cost per bit" $\overline{C}$ for some values of d, where the computational cost is measured, as anticipated, relatively to the computational cost of a 32-bit XOR. It is worth observing that the overall cost per bit does not change much with the size of the Galois field, with the most expensive field being $\mathbb{F}_{2^{32}}$.

Remark V.2
It is worth observing that the algorithms chosen for the experiments and the representation used for the elements of $\mathbb{F}_{2^d}$ are not the only possible. The choice of the "best" representation of elements of $\mathbb{F}_{2^d}$ and of the "best" algorithms can be done *only* once the architecture has been chosen since, especially with procedures as short as the ones presented here, details such as the internal structure of the processor, memory alignment, cache, and maybe others can play a non-negligible role. Therefore, the complexity figures given in this section should be taken only as *planning figures*.

H. Information obfuscation

The reduction procedure described in Example IV.1 has some similarity with the secret sharing technique of Shamir [32]. This suggests that one could use it to add some protection to the transmitted content. In order to simplify the discussion, we need a new definition.

Definition 2. *We will say that a reduction scheme with reduction factor R* achieves *k*-secrecy *if, given*

- *A positive integer K*
- *Any k-ple of K-dimensional row vectors (representing reduced packets) $\mathbf{u}_{c_1}, \ldots, \mathbf{u}_{c_k} \in \mathbb{F}_{2^d}^K$*
- *Any content packet $\mathbf{C}$ with R rows and K columns,*

it is possible to find $R-k$ reduced packets $\mathbf{u}_{c_{k+1}}$ $\mathbf{u}_{c_R} \in \mathbb{F}_{2^d}^K$ so that the content packet recovered from the whole set of reduced packets $\mathbf{u}_{c_1}, \ldots, \mathbf{u}_{c_R}$ is $\mathbf{C}$.

Definition 2 formalizes the idea that, if k-secrecy is achieved, an opponent that gets to know no more than k reduced packets, cannot deduce anything about the original content packet since any content packet can give rise to ("is

compatible with") the sequence of eavesdropped packets $\mathbf{u}_{c_1}$, ..., $\mathbf{u}_{c_k}$. Note that the reduction scheme in Example IV.1 does not achieve even 1-secrecy since, given any reduced packet, there are many content packets that are not compatible with it.

We want to show how the scheme in Example IV.1 can be modified to obtain 1-secrecy. Let K be the number of columns of $\mathbf{C}$ and let $\eta^t \in \mathbb{F}_{2^d}^K$ a random row vector whose entries are iid and uniformly distributed over $\mathbb{F}_{2^d}$. Use η to extend $\mathbf{C}$ to obtain

$$\widehat{\mathbf{C}} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{C} \\ \eta \end{bmatrix} \tag{16}$$

Now reduce matrix (16) using, of course, a reduction vector with $R+1$ columns, that is,

$$\hat{\mathbf{u}} = [1, c, \ldots, c^R]\widehat{\mathbf{C}} = \mathbf{r}_c\mathbf{C} + c^R\eta \tag{17}$$

Suppose now that an eavesdropper gets to know the reduced version $\hat{\mathbf{u}}$; we claim that the eavesdropper cannot deduce anything about $\mathbf{C}$. The reason is that for every choice of $\mathbf{C}$ one can find η such that (17) is satisfied, indeed

$$\eta = c^{-R}(\hat{\mathbf{u}} - \mathbf{r}_c\mathbf{C}) \tag{18}$$

where c^{-R} makes sense since c is a non-null element of $\mathbb{F}_{2^d}$. We have achieved 1-secrecy. It is easy to prove that k-secrecy can be achieved by extending $\mathbf{C}$ with a k-row random matrix.

Remark V.3

Although this technique seems to be specific for the Vandermonde reduction procedure, it can be extended to a more general case, as shown in [27], where it is also shown that a slightly stronger form of k-secrecy is achieved, that is, that the mutual information [33] between the content packet $\mathbf{C}$ and the reduced value $\hat{\mathbf{u}}$ is zero.

The advantage of this technique with respect to usual cryptography is that it does not require any key distribution, the drawbacks are an increased bandwidth (the reduced packets have the same dimension, but now a node must receive at least $R+k$ reduced packets instead of R) and the fact that an adversary that can get all the needed reduced packets can recover the content. If those drawbacks are compensated by the simplification due to the fact that no key distribution is necessary, depends on the applicative context.

VI. Packet loss probability

The current version of PPETP runs over UDP that, as well known, is an unreliable protocol. This means that a fragment sent to a lower peer could not reach its destination. It is clear that the probability that a given peer reconstructs a packet is a complex function of the packet loss probability and network structure. It is also clear that it is important to have an estimate, as precise as possible, of the overall packet loss probability experienced by a node. In this section we present some preliminary results about this topic.

A. Network model

A PPETP network can be represented by a Direct Acyclic Graph (DAG) where edges link each node to its lower peers, and where the server(s) is (are), clearly, the node(s) that do not have any upper peer. For the sake of notational simplicity, we will suppose that every link is an erasure channel that drops packets with probability P_ℓ.

We associate with each node n of the network the random variable W_n defined by the following experiment. We let the server(s) send to the network a single content packet, and we let $W_n \in \{0, 1, \ldots, N_{\text{up}}\}$ be the number of fragments received by node n. From the knowledge of the statistical properties of W_n, it is possible to determine several values of interest. For example, the packet loss probability P_{eq} seen by the application can be computed as $P_{\text{eq}} = P[W_n < R]$.

As explained in paragraph IV, a node sends reduced packets to its lower peers if it receives at least T reduced packets, where $T = 1$ if fragment propagation is employed and $T = R$ otherwise. If node n received at least T reduced packets (i.e., if $W_n \geq T$) we will say that the node is *active* or in *firing state*. We will define the random variable F_n to be equal to 1 if node n is in firing state and 0 otherwise.

1) Network topology: A difficulty in studying the behaviour of the abstract P2P system considered here is that the statistical properties of W_n depend on the network topology, a characteristic that it is not easily captured by a small set of parameters. In order to simplify the study, it is convenient to put some constraint on the topology.

A useful constraint that nevertheless is general enough to describe practical networks is the hypothesis of *limited spread*. Let n be a node of the network, consider the lengths of the paths joining n with the server (since the network is a DAG there is a finite number of paths joining n with the server) and define $d(n)$ and $D(n) \geq d(n)$ as the minimum and the maximum of these lengths. Value $D(n)$ will be called the *depth* of node n, and difference $D(n) - d(n)$ will be called the *spread* of n. The network will be said to have Δ-*limited spread* if $D(n) - d(n) \leq \Delta$ for every node n. The hypothesis of limited spread is quite natural and it is expected that this type of networks will be the natural outcome of the tentative of maximizing locality.

In this paper, we consider a special case of limited spread networks, namely, *stratified* networks; we use the term *stratified* to avoid confusion with the term *layered* possibly used in other contexts. In a stratified network, the nodes can be partitioned into sets (*strata*) $\mathfrak{L}_K$, $K \in \mathbb{N}$, such that all the upper peers of a node in $\mathfrak{L}_K$ belong to $\mathfrak{L}_{K-1}$. It is easy to verify that a network is stratified if and only if it has 0-limited spread and that the stratum index coincides with the node depth. Fig. 2 shows few examples of stratified networks, namely a tree network, a network made of parallel trees and an "onion skin" network. (Onion skin networks are interesting because the ratio of non-streaming nodes goes to zero when the network size goes to infinity.)

B. Notation

In this section we will consider Markov chains with a finite alphabet. We will use $\rightarrow$ to denote a one-step reachability relation, that is, we will write $a \rightarrow b$ if the chain can transition from a to b in one step. We will use $a \rightarrow^n b$ if there is a path of length n from a to b and $a \rightarrow^* b$ if there is a path of *any* length from a to b. If the Markov chain is homogeneous, we will use the shorthand $P(a \rightarrow b_1 \rightarrow b_2 \rightarrow \cdots b_N)$ to denote $P[s_{n+N} = b_N, \ldots, s_{n+1} = b_1 | s_n = a]$. Note that this notation factorizes, that is, $P(a \rightarrow b \rightarrow c) = P(a \rightarrow b)P(b \rightarrow c)$.

1) Notation for stratified networks: We will denote with L_K the number of nodes in stratum K. The n-th node in stratum K, $n = 0, \ldots, L_K - 1$, will be named as (K,n). The set of upper peers of (K,n) will be represented by the vector $\mathbf{u}_{K,n} \in \{0,1\}^{L_{K-1}}$ whose m-th component is 1 if $(K-1,m)$ is an upper peer of (K,n) and zero otherwise.

We will collect all the random variables $W_{K,n}$ and $F_{K,n}$, relative to nodes of stratum K, in two vectors $\mathbf{W}_K$ and $\mathbf{F}_K$ defined as

$$[\mathbf{W}_K]_n = W_{K,n} \qquad ; \qquad [\mathbf{F}_K]_n = F_{K,n} \tag{19}$$

Note that $\mathbf{F}_K \in \{0,1\}^{L_K}$. It will prove useful to have a special notation for some states in $\{0,1\}^{L_K}$. More precisely, we will define the *empty state* as $\phi = [0,0,\ldots,0]$ (no node in active state), the *full state* as $\Omega = [1,1,\ldots,1]$ (every node in active state) and, for every $k \in \{0,\ldots,L_K - 1\}$, the *k-th singleton state*, $\mathbf{e}_k$ as $[\mathbf{e}_k]_n = \delta_{k,n}$ (only the k-th node is active).

C. Equivalent loss probability

Consider a node (K,n) in stratum K and consider the following experiment: the origin server sends a content packet over the network and we check if node (K,n) recovers the content packet or not. Our goal is to obtain a bound to the equivalent loss probability P_{eq}, that is, the probability that the node does not recover the packet. It will be more convenient, from a notation point of view, to bound the probability of the complementary event $1 - P_{\text{eq}}$.

Property 1. *The following bound holds*

$$1 - P_{eq} \geq \eta \lambda^{\sum_{n=1}^{K} L_n} \tag{20}$$

where

$$\eta = P[\mathscr{B}(N_{up}, P_T) \geq R] \tag{21a}$$

$$\lambda = P[\mathscr{B}(N_{up}, P_T) \geq T] \tag{21b}$$

and $\mathscr{B}(N_{up}, P_T)$ in (21) is a binomial random variable with N_{up} trials and success probability P_T.

The proof of Property 1 is given in Appendix B.

Remark VI.1

Note that if fragment propagation is employed, $T = 1$ and (21b) can be written as

$$\lambda = 1 - P_\ell^{N_{\text{up}}} \tag{22}$$

Bound (20) decays exponentially with the number of peers in the network ($\sum_{n=1}^{K-1} L_n$ is the number of peers in the strata above stratum K), so it would seem not a very good bound. However, note that since the empty state ϕ is absorbing (that is, when a stratum reaches ϕ every successive strata will remain in ϕ), a well-known results on Markov chains implies that the probability of the empty state goes to 1 when the number of strata goes to infinity, so that the probability of reconstruction must converge to zero. Therefore, any lower bound of such a probability must converge to zero, too.

It is worth considering a simple numerical example, in order to understand better bound (20) and the difference between using or not fragment propagation. Suppose, for the sake of this example, that $P_\ell = 0.1$, that every node has $N_{\text{up}} = 10$ upper peers, that the reduction factor is $R = 6$ and that every stratum has $L = 100$ nodes.

If fragment propagation is employed, according to (22)

$$\lambda = 1 - P_\ell^{N_{\text{up}}} = 1 - 10^{-10} \tag{23}$$

Suppose we want to find $M = \sum_{n=1}^{K-1} L_n$ such that term λ^M becomes equal to 0.999. It is

$$M = \frac{\log_{10} 0.999}{\log_{10}(1 - 10^{-10})} = \frac{-4 \cdot 10^{-4}}{-4 \cdot 10^{-11}} = 10^7 \tag{24}$$

that corresponds to 10^5 strata if every stratum has 100 nodes. That is, although the bound (20) goes to zero when the network size goes to infinity, the decay is slow enough to be negligible for all but very large networks.

If no fragment propagation is employed, the value of λ is

$$\lambda = P[\mathscr{B}(10, 0.9) \geq 5] \approx 1 - 1.410^{-4} \tag{25}$$

Note that without fragment propagation, the term λ^M becomes smaller than 0.999 already with $M = 7$. Although (20) is only a lower bound, it suggests that convergence to the empty state can be very fast if fragment propagation is not used.

It is also worth observing that while the value of λ without fragment propagation depends on the redundancy $\rho = N_{\text{up}}/R$ (we had to use $\rho = 2$ in (25) in order to get a fairly large value for λ), the value of λ without fragment propagation *depends only on N_{up}* and we can obtain values of λ very close to 1 even with ρ small.

D. Repeated fragments

Note that in the proof of Property 1 it was implicitly assumed that the fragments received by the node were all different and one could wonder how much this hypothesis is true in a real case. This section is devoted to the discussion of this hypothesis.

First observe that, according to the results of Section V-B, we can safely assume that the reduction parameters chosen by the upper peers of a given node are different one another as soon as the number $|J|$ of reduction parameters is large enough. Moreover, if the network is not too large we can assume that the reduction parameters chosen by all the the *ancestors* of a given node are different.

Therefore, if the number of reduction parameters is sufficiently large with respect to the network size, if a node receives the same fragment twice, both fragments must have been originated by a single node. This happens, for example,

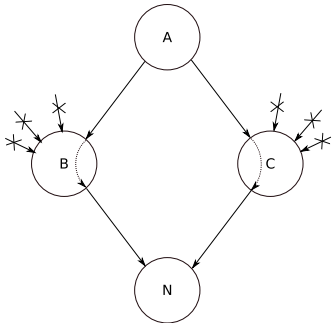


Figure 6. Example of a duplicated fragment event: both nodes B and C receive only the fragment from A and forward it to node N.

in the case of the *diamond* shown in Fig. 6 where node A sends a fragment to nodes B and C, both upper peers of N. If both B and C receives less than R fragments, it is possible that they both send to N the fragment received from A. Note that if N_{up} and R are suitably chosen, the probability that this happens is very small since it is necessary (but not sufficient) that both B and C cannot recover the corresponding content packet.

Since the problem of having an estimate of the event of duplicated fragment is still open, we decided to validate the effect of the hypothesis of no duplicated fragment by carrying out some simulations reported in Section VI-E.

E. Experimental results

We carried out some simulations in order to verify the theoretical results above and to asset the importance of the hypothesis of no duplicated fragment in a real case. For every choice of parameters P_ℓ, R, N_{up} we generated 20 random networks with 15 nodes per stratum, each node with N_{up} upper peers. Over each network we sent 1000 content packets and measured the probability (averaged over all the networks) that a node of a given stratum is able to recover the content packet. We carried out the simulations both with and without the fragment propagation policy. For each fragment we tracked its reduction parameter, therefore taking into account in the simulation the event of duplicated fragments. When a node is not able to recover the content packet, it selects one of the received fragments at random and forwards that to the lower peers.

According to the theoretical results described in Section VI-C we make the following predictions

- In the case *with* fragment propagation, with $P_\ell^{N_{\text{up}}}$ small, we expect P_{eq} approximately equal to the loss probability experienced when protecting packets sent over a channel with erasure probability P_ℓ using a (N_{up}, R) code, practically independent on the stratum number.
- In the case *without* fragment propagation we expect a P_{eq} that converges very rapidly to 1.

Fig. 7 shows, on a logarithmic scale, the probability of packet recovery $1-P_{\text{eq}}$ as a function of the stratum number for the cases $P_\ell = 0.2$, $N_{\text{up}} = 13$, redundancy factor R equal to 7 (first row), 8 (second row) and 9 (third row), with and without fragment propagation (left and right hand column, respectively). Observe that the probability remains approximately constant for all the cases with fragment propagation, while it decreases rapidly when fragment propagation is not employed. Note the reduced stratum range for figures Fig. 7b2 and Fig. 7c2; if we used the same range of the other figures, the curve would have looked like a vertical line. Note also that although in the case of Fig. 7a2 the probability does not decay as fast as in the other two figures of the same column, the decay is perceptible, while it is practically invisible in the three figures of the first column, relative to the fragment propagation case.

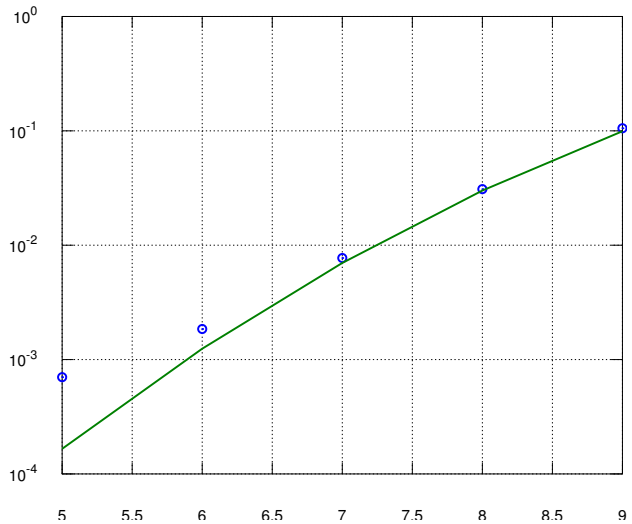


Figure 8. Comparison between the measured P_{eq} and the theoretical prediction for $P_\ell = 0.3$, $N_{\text{up}} = 13$ and $R \in \{6,7,8,9\}$.

Fig. 8 compares the measured equivalent packet loss probability P_{eq} (averaged over all the strata) in the case of fragment propagation with the probability of not receiving at least $R \in \{6,7,8,9\}$ packets out of $N_{\text{up}} = 13$ with a loss probability equal to $P_\ell = 0.3$. The match between theory and experiment is very good, the relatively large disagreement for $R = 6$ is due to the fact that the number of iterations (1000×20) is relatively small with respect to the expected value of P_{eq} ($\approx 10^{-3}$).

VII. Robustness against churn

An important problem in P2P streaming network is that a node can leave at any time, leaving its lower peers without data. The "turbulence" in the network induced by the random leaving of peers is called *churn*. Protecting a P2P streaming system from the effect of churn is a major goal in P2P system design. The effect of churn on PPETP was originally analyzed in [34]. In this section after recalling, for the sake of completeness, some results from [34], we simplify the results of [34] by giving some bounds that can make the design of a PPETP network easier.

Consider a network where each node is supposed to have N_{up} upper peers and let $H(t) \in \{0,\ldots,N_{\text{up}}\}$ denote the actual number of upper peers of a given node at time t. Note that $H(t)$

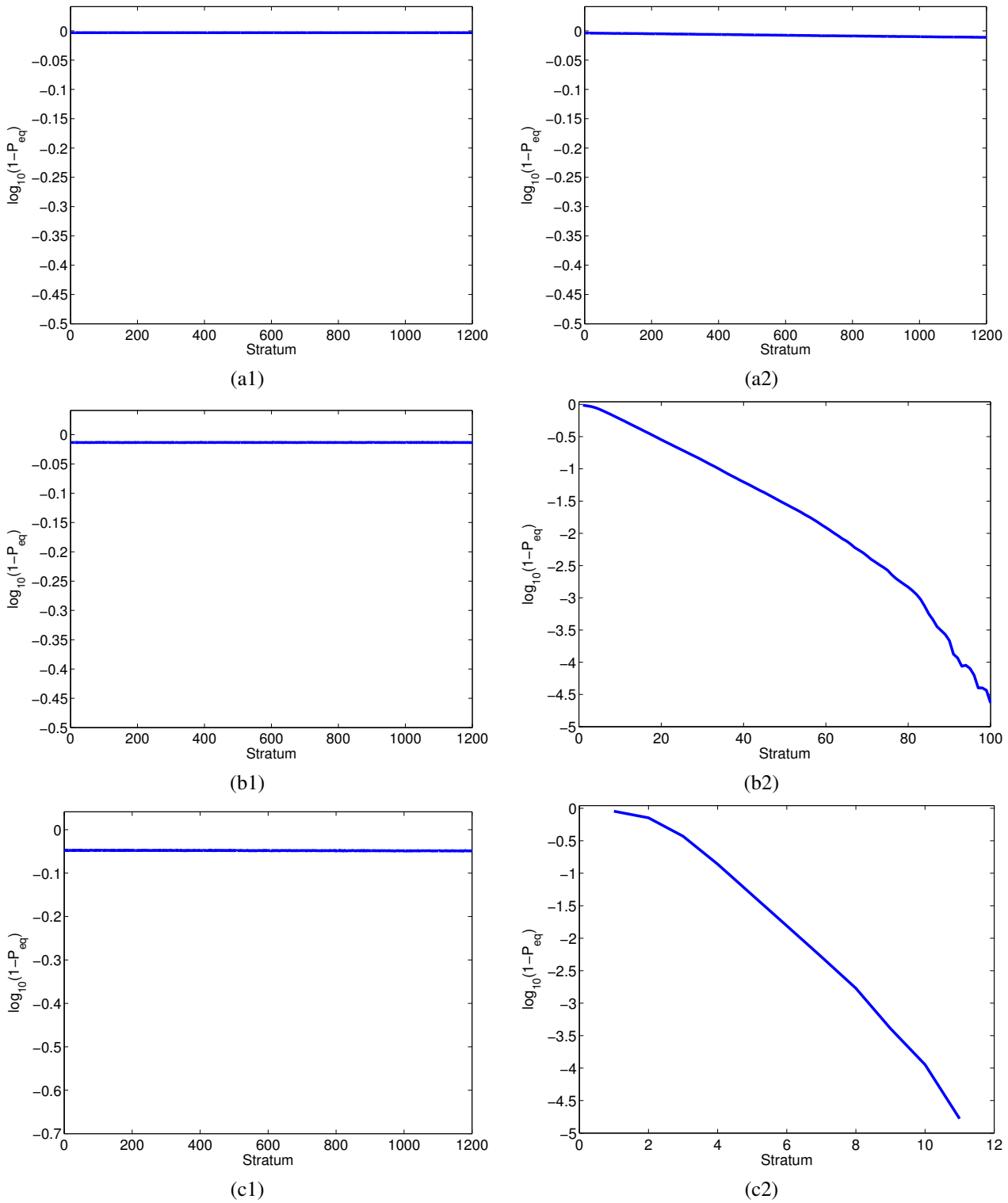


Figure 7. Probability of content packet recovery as function of the stratum number. All the plots are relative to $P_\ell = 0.2$ and $N_{\text{up}} = 13$. (a1) $R = 7$, with fragment propagation; (a2) $R = 7$, without fragment propagation; (a1) $R = 8$, with fragment propagation; (b2) $R = 8$, without fragment propagation; (a1) $R = 9$, with fragment propagation; (c2) $R = 9$, without fragment propagation. Plots (b2) and (c2) have a reduced range on the x axis since otherwise the plot would have looked as a vertical line.

can be smaller than N_{up}, for example, after some upper peer leaves. Note that if $H(t) < R$ the node cannot receive enough reduced packets to recover the content. We will call this event an *underflow* event and we will denote its probability as P_{under}.

In [34] probability P_{under} is computed as a function of R and N_{up} supposing that

1) The search of a new peer requires a time that can be described by an exponential random variable, with appropriate parameter λ. The average $1/\lambda$ is typically in the order of a few seconds or fraction of seconds.
2) The time a peer remains connected can be described by an appropriate distribution. with average $1/\mu$. Typical values of $1/\mu$ are in the order of at least several minutes. Note that we do not suppose the distribution exponential since some results in [7] show that distributions other than exponential model can be more appropriate.

According to [34], P_{under} can be written as

$$P_{\text{under}} = P[H(t) < R] = \frac{\sum_{n=0}^{R-1} \frac{\gamma^n}{n!}}{\sum_{n=0}^{N_{\text{up}}} \frac{\gamma^n}{n!}} \tag{26}$$

where $\gamma = \lambda/\mu$. Note that in a typical case one can expect γ to be quite large, at least of the order of few hundreds, even thousands.

The formula above, albeit exact, can be inconvenient to use for design purposes. In the following we are going to give an upper bound to probability (26) that holds for large values of γ and depends on γ, R and N_{up} in a more intuitive way. The upper bound we are going to present is not in [34] and it is published here for the first time. We will need the following lemma that allows us to upper bound the sums in (26) with a single term as soon as γ is large enough.

Lemma 1. *For every $M \in \mathbb{N}$ and $\gamma > 2(M-1)$ the following inequalities hold*

$$\frac{\gamma^M}{M!} < \sum_{n=0}^{M} \frac{\gamma^n}{n!} < 2\frac{\gamma^M}{M!} \tag{27}$$

Proof: The following equality is well-known

$$\sum_{n=0}^{M} \frac{\gamma^n}{n!} = e^{\gamma} \frac{\Gamma(M+1,\gamma)}{M!} \tag{28}$$

where $\Gamma(M+1,\gamma)$ is the incomplete gamma function. In [35] it is shown that for $a > 1$, $B > 1$ and $x > (a-1)B/(B-1)$ the following inequalities hold

$$x^{a-1}e^{-x} < \Gamma(a,x) < Bx^{a-1}e^{-x} \tag{29}$$

Using inequalities (29) in (28) with $a = M+1$, $B = 2$ and $x = \gamma > (a-1)B/(B-1) = 2M$, it follows

$$\frac{e^{\gamma}(\gamma^M e^{-\gamma})}{M!} < e^{\gamma}\frac{\Gamma(M+1,\gamma)}{M!} < \frac{e^{\gamma}(2\gamma^M e^{-\gamma})}{M!} \tag{30}$$

From (30) the thesis follows. ■

We will give the upper bound in the specific case of $N_{\text{up}} \leq 2R$. The more general case is not much more difficult, but it gives rise to more complex expressions that partially spoil the simplification introduced by the upper bound. Note that the hypothesis $N_{\text{up}} \leq 2R$ is quite reasonable since it seems very unlikely to have the necessity of a number of upper peers that is more than twice the minimum.

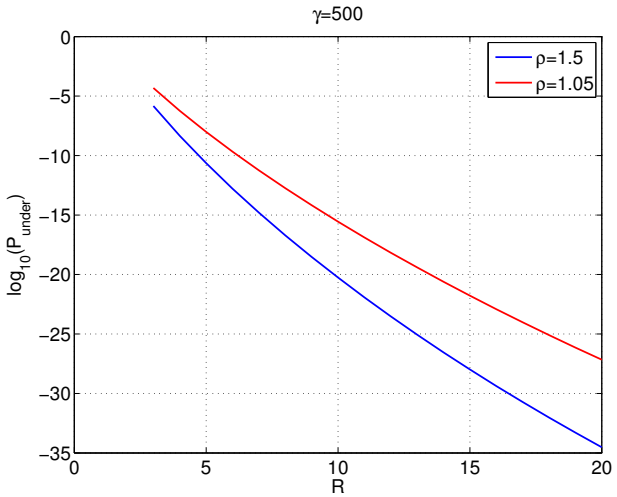


Figure 9. Upper bound to the underflow probability P_{under} for $\gamma = 500$, and $\rho = 1.05$ (the upper curve) and $\rho = 1.5$ (the lower curve).

Property 2. *If $N_{up} \leq 2R$ and $\gamma > 2(R-1)$, then the underflow probability P_{under} can be upper bounded as*

$$P_{under} \leq 2\left(\frac{N_{up}}{\gamma}\right)^{N_{up}-R+1} \tag{31}$$

Proof: From Lemma 1 with $M = R-1$, it follows that when $\gamma > 2(R-1)$,

$$P_{\text{under}} = \frac{\sum_{n=0}^{R-1}(\gamma^n/n!)}{\sum_{n=0}^{N_{\text{up}}}(\gamma^n/n!)} \leq \frac{\sum_{n=0}^{R-1}(\gamma^n/n!)}{\gamma^{N_{\text{up}}}/N_{\text{up}}!} \leq \frac{2\gamma^{R-1}/(R-1)!}{\gamma^{N_{\text{up}}}/N_{\text{up}}!} \tag{32}$$

By observing that

$$\begin{aligned} \frac{\gamma^{R-1}/(R-1)!}{\gamma^{N_{\text{up}}}/N_{\text{up}}!} &= \gamma^{R-N_{\text{up}}-1}R\cdots N_{\text{up}} \\ &\leq \gamma^{R-N_{\text{up}}-1}N_{\text{up}}^{N_{\text{up}}-R+1} = \left(\frac{N_{\text{up}}}{\gamma}\right)^{N_{\text{up}}-R+1} \end{aligned} \tag{33}$$

the thesis follows. ■

Note that since in a practical case γ will be of the order of many hundreds, while N_{up} is expected to be at most around ten, from Property 2 one can deduce that one can make P_{under} very small with a small number of excess peers $N_{\text{up}} - R$. Fig. 9 show two bounds for $\gamma = 500$, and $\rho = 1.05$ (the upper curve) and $\rho = 1.5$ (the lower curve).

VIII. Design Guidelines

The procedure for designing a PPETP network depends, of course, on the specific application and the corresponding figures of merit of interest. In this section we give some guidelines that can be useful in designing a PPETP network in

what can be expected to be a fairly common setup. Of course all the characteristics considered so far (e.g., robustness to packet loss, jitter, computational complexity, efficiency, ...) are interrelated one another and the trade-off between them will depend on the specific application.

We will suppose to have an estimate of the values P_ℓ (the loss probability over a single link), $1/\lambda$ (the average time a peer remains connected, see Section VII), $1/\mu$ (the average time required to find a new peer, see Section VII), the content bandwidth B and the minimum upload bandwidth $U_{\min}$ available at the nodes. We also have some quality of service constraints, such as a maximum underflow probability P_{under} (see Section VII) and a maximum packet loss probability at the application level P_{eq} (see Section VI). Finally, we desire to keep $\rho = N_{\text{up}}/R$ as small as possible, since the overall required bandwidth grows linearly with ρ. The parameters that we need to determine are the Galois field $\mathbb{F}_{2^d}$, the reduction factor R and the redundancy ρ (or, equivalently, the number of upper nodes $N_{\text{up}} = \rho R$).

An obvious constraint on R is given by the fact that, if data puncturing is not employed, it must be $R \geq \lceil B/U_{\min} \rceil$, where $\lceil x \rceil$ denotes the smallest integer not smaller than x.

Observe that the minimum value admissible for ρ is fixed by R and P_ℓ since ρ must be such that the probability of receiving at least R fragments out of ρR is not smaller than $1 - P_{\text{eq}}$. Note that for small values of P_{eq}, ρ cannot be smaller than $1/P_\ell$ and that it gets closer to that optimal value as R grows. Therefore, in order to minimize ρ, it is convenient to choose a large value for R.

Using a large R has other advantages, too. For example, both the probability P_{under} of the underflow event (see Section VII) and the the probability P_{fail} of having less than R different reduction parameters (see Section V-B) decrease with R. Moreover, for a fixed value of ρ, also the jitter (see Section V-F) and the decay of recovery probability $1 - P_{\text{eq}}$ (see Section VI) improve with R since $N_{\text{up}} = \rho R$. The only drawback of a large value of R is, according to (15), an increased computational complexity.

Summarizing, we can give the following guidelines for the choice of ρ and R

- Choose, tentatively, $d = 32$ since the increase in computational complexity is not huge (see Table I) and it helps in keeping P_{fail} small.
- Choose R as large as possible, at least large enough to satisfy the constraint $R \geq \lceil B/U_{\min} \rceil$.
- Choose ρ so the the probability of receiving at least R fragments out of ρR is not smaller than $1 - P_{\text{eq}}$.
- Verify that, with the given choices of ρ and R, the constraints on P_{fail} and P_{under} are satisfied. If they are not, choose a larger R. Note that even if we increase R, we can keep the same ρ since it will satisfy the P_{eq} constraint even if R is increased. Alternatively, if R cannot be increased because of the computational complexity, one can increase ρ.
- Verify that the computational complexity for the chosen d and R is acceptable. If it is not, one can lower it by using a different value for d (e.g., $d = 16$) or reiterating the design procedure with a smaller R. If R was already equal to the minimum value $\lceil B/U_{\min} \rceil$, one can try to employ data puncturing for the nodes with smallest bandwidth.

```
# Input  : cl and dl
# Output : al and ah

movb %cl, %dh
shll #1, %edx
movw tbl_4(%edx), %ax
```

(a)

```
# Input  : cl and dl
# Output : al

movb %cl, %dh
movb tbl_8(%edx), %al
```

(b)

```
# Input  : cx, dx
# Output : ax

# compute log(%ecx)
movl %ecx, %eax
orw  %ax, %ax
jz   done
shl  #1, %eax
movw log_16(%eax), %cx

# compute log(%edx)
movl %edx, %eax
orw  %ax, %ax
jz   done
shl  #1, %eax
movw log_16(%eax), %dx

# Compute the sum of
# the logs mod 2**16-1
addw %dx, %cx
jnc no_mod_needed
incw %cx

no_mod_needed:
   shll #1, %ecx
   movw exp_16(%ecx), %ax
done:
```

(c)

```
# Input  : ecx, edx
# Output : eax

carry_mask=0x8299
# Init the result
movl #0, %eax

orl  %edx, %edx
jz   done

main_loop:
   # If the LSB of
   # edx is 1, xor
   # ecx with the result
   test #0x0001, %edx
   jz  skip_xor
   xorl %ecx, %eax

skip_xor:
   # Shift left cx
   shl #1, %ecx
   jnc no_reduction
   xor carry_mask, %ecx

no_reduction:
   # Shift right dx
   shr #1, %edx
   jnz  main_loop
done:
```

(d)

Figure 10. The product algorithms used in the velocity tests. (a) Algorithm for $\mathbb{F}_{2^4}$. (b) Algorithm for $\mathbb{F}_{2^8}$. (c) Algorithm for $\mathbb{F}_{2^{16}}$. (d) Algorithm for $\mathbb{F}_{2^{32}}$.

IX. Conclusions and Future Work

This article has described PPETP, an overlay multicast protocol that allows for efficient data propagation even when some nodes have limited resources. A quantitative analysis of some figures of merit of PPETP and some design guidelines have been presented.

A. Acknowledgments

PPETP is partially funded by Italian Ministry PRIN project *Arachne*.

Appendix

A. Computation of P_{fail}

Our goal is to compute $P_{\text{fail}}(N_{\text{up}}, R, S)$, that is the probability that after N_{up} drawing from an alphabet with S elements we have less than R different values. This experiment can be represented by the finite state system shown in Fig. 12. Each state is labeled with the number of different symbols extracted

```
uint16_t table[65536];  /* Filled at init time */

void F16_mult(byte b,
              byte in_1,   byte in_2,
              byte *out_1, byte *out_2)
 {
   /* Make the 16-bit index to access the table
    * as follows
    *
    *     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    *     |  r_2  |  r_1  |       b       |
    *     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    *    MSB                             LSB
    */
   uint16 index    = in_2 << 12 + in_1 << 8 + b;
   uint16 out_pair = table[index];

   *v_1 = out_pair & 0xff;  /* LS bits */
   *v_2 = out_pair >> 8;    /* MS bits */
 }
```

Figure 11. Pseudo-C code to compute with a single table access products `out_1 = in_1*b` and `out_2 = in_2*b`, where `in_1` and `in_2` represent elements of $\mathbb{F}_{16}$ and `b`, `out_1` and `out_2` vectors in $\mathbb{F}_{16}^2$.

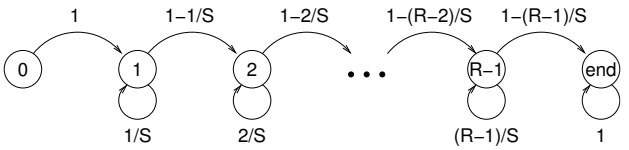


Figure 12. Markov chain used to compute the curves of Fig. 4.

so far and the system starts from state 0. The system goes in the state marked with "end" when R or more different symbols have been extracted. It is easy to see that $P_{\text{fail}}(N_{\text{up}},R,S) = 1-P[s(N_{\text{up}}) = \text{end}]$, where $s(n)$ is the state after n extractions. This probability can be easily obtained from $\mathbf{P}^{N_{\text{up}}}$, where $\mathbf{P}$ is the transition matrix of Fig. 12. By writing explicitly $\mathbf{P}$ it is easy to check that the $\lim_{n\to\infty} P[s(n) = \text{end}] = 1$ and that the largest eigenvalue of $\mathbf{P}$ less than 1 is $(R-1)/S$. Therefore, $P_{\text{fail}}(n,R,S)$ converges to zero as $[(R-1)/S]^n$.

B. Proof of Property 1

In order to proof Property 1 we need the following lemma.

Lemma 2. *For every $K \geq 1$, the probability that stratum K is in full state is bounded as follows*

$$P[\mathbf{F}_K = \Omega] \geq \lambda^{\Sigma_{n=1}^{K} L_n} \tag{34}$$

where λ is as in (21).

Proof: We proceed by induction. For $K = 1$ the event $\mathbf{F}_1 = \Omega$ holds if every node of the first stratum receives at least T fragments. Since the upper peers of the nodes of the first stratum are origin servers, the number of fragments received by a node is a binomial variable with N_{up} tentatives and success probability P_T, that is, the probability that a given node of the first stratum is in firing state is λ. Since all the links from stratum 0 to stratum 1 are independent one another,

$$P[\mathbf{F}_1 = \Omega] = \lambda^{L_1} \tag{35}$$

that is (34) with the equality sign.

Suppose now that bound (34) holds for $K-1$ and prove it for $K > 1$. It is

$$\begin{aligned} P[\mathbf{F}_K = \Omega] &= \sum_{\mathbf{u}\in\{0,1\}^{L_{K-1}}} P[\mathbf{F}_K = \Omega|\mathbf{F}_{K-1} = \mathbf{u}]P[\mathbf{F}_{K-1} = \mathbf{u}] \\ &\geq P[\mathbf{F}_K = \Omega|\mathbf{F}_{K-1} = \Omega]P[\mathbf{F}_{K-1} = \Omega] \\ &\geq P[\mathbf{F}_K = \Omega|\mathbf{F}_{K-1} = \Omega]\lambda^{\Sigma_{n=1}^{K-1} L_n} \end{aligned} \tag{36}$$

where the last inequality follows from the inductive hypothesis.

In order to compute $P[\mathbf{F}_K = \Omega|\mathbf{F}_{K-1} = \Omega]$ observe that if all the nodes in stratum $K-1$ are in firing state, a reasoning similar to the one used to derive (35) holds and one obtains

$$P[\mathbf{F}_K = \Omega|\mathbf{F}_{K-1} = \Omega] = \lambda^{L_K} \tag{37}$$

Using (37) in (36) gives the thesis. ■

References

[1] R. Bernardini, R. C. Fabbro, and R. Rinaldo, "Ppetp: A peer-to-peer overlay multicast protocol for multimedia streaming," in *Proc. of CONTENT 2011*, (Rome), Sept. 2011. Best paper award.

[2] E. Adar and B. A. Huberman, "Free riding on Gnutella," *First Monday*, vol. 5, October 2000.

[3] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker, "Making gnutella-like p2p systems scalable," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '03, (New York, NY, USA), pp. 407–418, ACM, 2003.

[4] V. Fodor and G. Dán, "Resilience in live peer-to-peer streaming," *IEEE Communications Magazine*, vol. 45, pp. 116–123, June 2007.

[5] V. Padmanabhan, H. Wang, P. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking," in *Proc. of NOSSDAV 2002*, (Miami, Florida, USA), ACM, May 2002.

[6] M. Piatek, T. Isdal, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "Do incentives build robustness in BitTorrent?," in *Proceedings of 4th USENIX Symposium on Networked Systems Design & Implementation (NSDI 2007)*, (Cambridge, MA), USENIX, April 2007.

[7] D. Stutzbach and R. Rejaie, "Understanding churn in peer-to-peer networks," in *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, (Rio de Janeriro, Brazil), pp. 189–202, SIGCOMM, 2006.

[8] M. Wang and B. Li, "R2: Random push with random network coding in live peer-to-peer streaming," *IEEE Journal on Selected Areas in Communications*, vol. 25, pp. 1655–1666, December 2007.

[9] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content-addressable network," in *IN PROC. ACM SIGCOMM 2001*, pp. 161–172, 2001.

[10] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: High-bandwidth multicast in cooperative environments," in *19th ACM Symposium on Operating Systems Principles, 2003*, 2003.

[11] S. Marti and H. Garcia-molina, "Taxonomy of trust: Categorizing p2p reputation systems," *Computer Networks*, vol. 50, pp. 472–484, 2006.

[12] S. Iyer, A. Rowstron, and P. Druschel, "Squirrel: A decentralized peer-to-peer web cache," in *12th ACM Symposium on Principles of Distributed Computing (PODC 2002)*, pp. 1–10, July 2002.

[13] Y. Yue, C. Lin, and Z. Tan, "Analyzing the performance and fairness of bittorrent-like networks using a general fluid model," *Computer Communications*, vol. 29, no. 18, pp. 3946 – 3956, 2006.

[14] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *SIGCOMM Comput. Commun. Rev.*, vol. 31, pp. 149–160, August 2001.

[15] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems," in *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pp. 329–350, Nov. 2001.

[16] X. Hei, Y. Liu, and K. W. Ross, "IPTV over P2P streaming networks: The mesh-pull approach," *IEEE Communications Magazine*, vol. 46, pp. 86–92, Feb. 2008.

[17] D. A. Tran, K. A. Hua, and T. Do, "Zigzag: An efficient peer-to-peer scheme for media streaming," in *Proceedings of IEEE INFOCOM*, (San Francisco, CA), 2003.

[18] V. Padmanabhan, H. Wang, and P. Chou, "Supporting heterogeneity and congestion control in peer-to-peer multicast streaming," in *Proceedings of IPTPS*, (San Diego, CA), Feb. 2004.

[19] X. Liao, H. Jin, Y. Liu, L. Ni, and D. Deng, "Chunkyspread: Heterogeneous unstructured tree-based peer to peer multicast," in *Proceedings of IEEE International Conference on Computer Communications*, IEEE Computer Society, Apr. 2006.

[20] X. Zhang, J. Liuy, B. Liz, and P. Yum, "CoolStreaming/DONet: a data-driven overlay network for efficient live media streaming," in *Proceedings of IEEE International Conference on Computer Communications*, IEEE Computer Society, Mar. 2005.

[21] L. X., J. H., L. Y., N. L., and D. D., "AnySee: Peer-to-Peer live streaming," in *Proc. INFOCOM 2006*, pp. 1–10, 2006.

[22] N. Magharei and R. Rejaie, "Prime: peer-to-peer receiver-driven mesh-based streaming," in *Proceedings of IEEE International Conference on Computer Communications*, (Alaska), May 2007.

[23] J. Liang and K. Nahrstedt, "Dagstream: locality aware and failure resilient peer-to-peer streaming," in *Proceedings of MMCN*, Jan. 2006.

[24] C. Wu and B. Li, "rStream: resilient and optimal peer-to-peer streaming with rateless codes," *T-par*, pp. 77–92, Jan. 2008.

[25] A. Magnetto, R. Gaeta, M. Grangetto, and M. Sereno, "P2p streaming with lt codes: a prototype experimentation," in *Proc. ACM Multimedia 2010*, pp. 7–12, Oct. 2010.

[26] R. Bernardini, R. C. Fabbro, and R. Rinaldo, "Peer-to-peer epi-transport protocol." `http://tools.ietf.org/html/draft-bernardini-ppetp`, Jan. 2011. Internet Draft, work in progress.

[27] R. Bernardini, R. C. Fabbro, and R. Rinaldo, "Group based reduction schemes for streaming applications," *ISRN Communications and Networking*, vol. 2011, 2011. Article ID 898254, doi:10.5402/2011/898254.

[28] R. Bernardini, R. C. Fabbro, and R. Rinaldo, "Peer-to-peer streaming based on network coding improves packet jitter," in *Proc. of ACM Multimedia 2010*, (Florence, Italy), Oct. 2010.

[29] H. A. David, *Order Statistics 2nd edition*. Wiley-Interscience, 1981.

[30] http://www.gnu.org/software/binutils.

[31] Intel Corporation, *Intel® 64 and IA-32 Architectures Software Developer's Manual*. No. 253669-033US, December 2009.

[32] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, pp. 612–613, November 1979.

[33] T. M. Cover and J. A. Thomas, *Information theory*. New York: Wiley, 1991.

[34] R. Bernardini, R. Rinaldo, and A. Vitali, "A reliable chunkless peer-to-peer architecture for multimedia streaming," in *Proc. Data Compr. Conf.*, (Snowbird, Utah), pp. 242–251, Brandeis University, IEEE Computer Society, Mar. 2008.

[35] P. Natalini and B. Palumbo, "Inequalities for the incomplete gamma function," *Math. Inequal. Appl. 3*, no. 1, pp. 69–77, 2000.

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