CTRQ 2013

The Sixth International Conference on Communication Theory, Reliability, and Quality of Service

ISBN: 978-1-61208-263-9

April 21 - 26, 2013

Venice, Italy

CTRQ 2013 Editors

Eugen Borcoci, Politehnica University of Bucharest, Romania

Pascal Lorenz, University of Haute Alsace, France

CTRQ 2013

Foreword

The Sixth International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ 2013), held between April 21st-26th, 2013 in Venice, Italy, continued a series of events focusing on the achievements on communication theory with respect to reliability and quality of service. The conference also brought onto the stage the most recent results in theory and practice on improving network and system reliability, as well as new mechanisms related to quality of service tuned to user profiles.

The processing and transmission speed and increasing memory capacity might be a satisfactory solution on the resources needed to deliver ubiquitous services, under guaranteed reliability and satisfying the desired quality of service. Successful deployment of communication mechanisms guarantees a decent network stability and offers a reasonable control on the quality of service expected by the end users. Recent advances on communication speed, hybrid wired/wireless, network resiliency, delay-tolerant networks and protocols, signal processing and so forth asked for revisiting some aspects of the fundamentals in communication theory. Mainly network and system reliability and quality of service are those that affect the maintenance procedures, on the one hand, and the user satisfaction on service delivery, on the other hand. Reliability assurance and guaranteed quality of services require particular mechanisms that deal with dynamics of system and network changes, as well as with changes in user profiles. The advent of content distribution, IPTV, video-on-demand and other similar services accelerate the demand for reliability and quality of service.

We take here the opportunity to warmly thank all the members of the CTRQ 2013 Technical Program Committee, as well as the numerous reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to CTRQ 2013. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the CTRQ 2013 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that CTRQ 2013 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of communication theory, reliability and quality of service.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the charm of Venice, Italy.

CTRQ Advisory Committee:

Eugen Borcoci, Politehnica University of Bucharest, Romania
Joel Rodrigues, Instituto de Telecomunicações / University of Beira Interior, Portugal
Pascal Lorenz, University of Haute Alsace, France
Raj Jain, Washington University in St. Louis, USA

CTRQ 2013

Committee

CTRQ Advisory Committee

Eugen Borcoci, Politehnica University of Bucharest, Romania
Joel Rodrigues, Instituto de Telecomunicações / University of Beira Interior, Portugal
Pascal Lorenz, University of Haute Alsace, France
Raj Jain, Washington University in St. Louis, USA

CTRQ 2013 Technical Program Committee

Gergely Acs, INRIA, Rhone-Alpes - Montbonnot, France
Artur Andrzejak, Ruprecht-Karls-University of Heidelberg, Germany
Himzo Bajric, Joint Stock Company BH Telecom - Sarajevo, Bosnia and Herzegovina
Jasmina Barakovic Husic, BH Telecom – Sarajevo, Bosnia and Herzegovina
Eugen Borcoci, Politehnica University of Bucharest, Romania
Bruno Checcucci, Perugia University, Italy
Laurent Ciavaglia, Alcatel-lucent, Italy
Javier Del Ser Lorente, TECNALIA Research & Innovation, Spain
Michel Diaz, LAAS, France
Manfred Droste, Universität Leipzig, Germany
Ali El Masri, Troyes University of Technology -Troyes, France
Andras Farago, The University of Texas at Dallas - Richardson, USA
Alexandre Fonte, Polytechnic Institute of Castelo Branco, Portugal & Centre for Informatics and Systems of the University of Coimbra (CISUC) , Portugal
Tulsi Pawan Fowdur, University of Mauritius, Mauritius
Julio César García Alvarez, Universidad Nacional de Colombia Sede Manizales, Colombia
Bogdan Ghita, University of Plymouth, UK
Marc Gilg, Université de Haute Alsace, France
Antti Hakkala, University of Turku, Finland
Bjarne J. Helvik, NTNU, Norway
Robert Ching-Hsien Hsu, Chung Hua University, Taiwan
Mohsen Jahanshahi, Islamic Azad University - Central Tehran Branch, Iran
Brigitte Jaumard, Concordia University, Canada
Sokratis K. Katsikas, University of Piraeus, Greece
Wojciech Kmiecik, Wroclaw University of Technology, Poland
Michal Kucharzak, Wroclaw University of Technology, Poland
Archana Kumar, Delhi Institute of Technology & Management - Haryana, India
Pascal Lorenz, University of Haute Alsace, France
Malamati Louta, University of Western Macedonia - Kozani, Greece
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France
Wail Mardini, Jordan University of Science and Technology, Jordan
Rubens Matos, Federal University of Pernambuco, Brazil

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Mean Waiting Time of an End-user in the Multiple Web Access Environment

Yong-Jin Lee
Department of Technology Education
Korea National University of Education
Cheongwon, Korea
e-mail: lyj@knue.ac.kr

Abstract— **Mean response time for single user and mean waiting time for multiple users are important measures of Quality of Service (QoS) in accessing a web server. This paper presents analytical models to find the mean response time and the mean waiting time for web service using Hyper Text Transfer protocol (HTTP) over Stream Control Transmission Protocol (SCTP). The proposed response and waiting time model assumes the multiple packet losses and a narrowband network, where fast retransmission is not possible due to small window. Our experiments validate the accuracy of the proposed model. It is shown that the differences between the results from the model and those from the experiments are very small on average. We also find that the mean waiting time for HTTP over SCTP is less than that for HTTP over TCP. The model can be used for dimensioning of the network link bandwidth to satisfy the QoS of end users.**

Keywords-mean waiting time; multiple web access; QoS

I. INTRODUCTION

TCP [1] provides a single streamed and strictly ordered delivery of data, which increases the users' perceived latency. SCTP [2,3] was proposed as a new transport layer protocol which has multi-streaming capability to transmit several independent streams of chunks (or messages) in parallel. When a packet loss occurs in a stream, it affects the relevant stream only.

Typically, response time is affected by data size and transmission time according to transmission rate of link as well as by congestion control mechanism. The congestion control mechanism of SCTP is similar with window-based one of TCP. Their common functions are slow-start, congestion avoidance, timeout, and fast retransmission.

Previous related works on analytical models of data transmission delay over TCP are as following: Padhye [4] considered large amount of data transmission on steady state over TCP. Most of TCP connections for HTTP data transmission, however, are short for small amount of data instead of large one in current internet environment. Connection setup or slow-start time dominates the performance of web in this environment. Noticing this phenomenon, Cardwell [5] extended the above steady state model but he did not consider delay of TCP after time-out. Jiong [6] enhanced the Cardwell's model by considering slow-start time after timeout of retransmission. However, since the above models assumed wideband network, they cannot be applied to the narrowband network environment, which this paper considers. That is because the narrowband network environment does not allow fast retransmission of data due to the very small size of window [7]. Furthermore, the previous studies are limited to single user cases, where the response time is a good measure of the end-to-end delay experienced by a user.

Chang et al. [8] studied the performance of File Transfer Protocol (FTP) over SCTP, and Lu [9] analyzed the performance of Session Initiated Protocol (SIP) over SCTP. Fei Ge [10] presents a simple closed-form formula to estimate the HTTP latency over FAST TCP, taking into account the network parameters such as packet size, link capacity, and propagation delay. Eklund et al. [11] developed a model that predicts the transfer times of SCTP messages during slow start. However, mean waiting time model for HTTP over SCTP in multiple users' environment has not yet been presented.

The motivation of this paper is to study the case of multiple users accessing a server, where the waiting and turnaround times depend on the server load. In such a case, the response time may not be a good measure of end-to-end delay.

The results reported in this paper can be used by network engineers to dimension a network in terms of bandwidth requirement and to develop scheme distributing the load among a number of web servers, in order to improve the waiting delay perceived by end users. The objective of this study is to find the theoretical upper bound of the actual waiting and turnaround times of users in a real environment when they download web objects using HTTP over SCTP in the narrowband network, which does not allow fast retransmission.

We achieve our objectives by developing an analytical model to compute the mean waiting and turnaround time of an end user when multiple users simultaneously access the web server. In contrast to previous work [12,13,14], which only considered the response time of an object for single user, we first consider the response time for single user and then find waiting delay for multiple users. The results of this paper will allow us to compute more realistic end-to-end delay experienced by a user in the real environment.

Since the estimated mean waiting time in this paper can be considered as QoS of end-users, it can be used as a benchmark to pre-estimate waiting time by considering size of objects, bandwidth, and round trip time. To validate the proposed mean waiting time model, we experimented in a simple test-bed and compared the results with estimated value. In addition, we compared the values with the mean waiting time of HTTP over TCP.

 ISBN: 978-1-61208-263-9

Sections 2 and 3 describe the estimation model and algorithm of mean response and waiting time for HTTP over SCTP, respectively. Section 4 discusses performance evaluation and analysis. We conclude this paper in section 5.

II. MEAN RESPONSE TIME MODEL FOR SINGLE USER

In this section, we first describe the mean response time model, when single user retrieves a web object in the narrowband network [14].

Fig. 1 shows the congestion control mechanism of SCTP in the narrowband network. In Fig.1, $th(1)$, $th(2)$, and $th(3)$ are the slow start thresholds and initially $th(1)=\infty$. y coordinate is the congestion window($cwnd$) and its initial value is $2\times mtu$. Here, mtu represents the maximum transfer unit of the link. Thus SCTP executes the slow-start period by increasing $cwnd$ exponentially such as 2, 4, 8, ... and detects the packet loss when timeout occurs at ①. SCTP responds to this as following.

$$th(2) = max(cwnd/2,\ 2 \times mtu\ )$$
$$cwnd = 1 \times mtu \qquad (1)$$

That is, the threshold of next stage is reduced to half size of the window in which packet loss occurred and slow-start period is repeated with congestion windows exponentially increased from 1 to 2, 4, 8, etc. When the congestion window exceeds threshold $th(2)$, congestion avoidance period is started. Since this period needs an acknowledgement every packet, it is called linearly increasing period. If a packet loss occurs as Fig. 1, ② in this period, there are two choices according to timeout. First of all, using (1) new threshold ($th(3)$) is obtained. If three duplicate acknowledgements are obtained before timeout, then fast retransmission (Fig. 1, ③) is started. Otherwise slow-start (Fig. 1, ④) is executed. In this paper, we assume the narrowband network which is not able to receive three duplicate acknowledgements during timeout. Thus the slow-start is executed.

In order to simplify the model we assume that sizes of web objects are identical and received packets are transmitted in an upper layer in terms of window unit. Let the size of an object to transfer be θ bits and maximum transfer unit mtu bits, then the number of packets to transfer for an object is $n = \lceil \theta/mtu \rceil$.

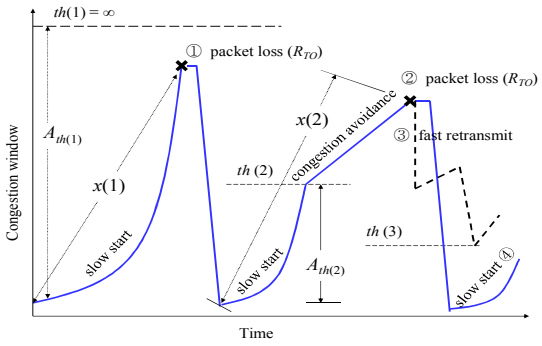


Figure 1. Congestion control of SCTP in the narrowband network

When the probability of a packet loss is p, the expected number of total packet loss is $\alpha = \lceil np \rceil$ in terms of binomial distribution. At this moment, a certain packet loss occurs during either slow-start phase or congestion avoidance phase.

From the above, we can identify the packet loss phase by comparing, for k^{th} packet loss, the possible number of packets ($A_{th(k)}$) to transmit until the threshold ($th(k)$, k=1,2,..,a) at which congestion avoidance starts, with the expected number of packets ($x(k)$: k=1,2,..,a) transmitted before the packet loss. At this time, $x(k)$ is calculated as a function of remained packets $N(k)$ and packet loss rate p.

We can determine that an arbitrary k^{th} packet loss occurs either during slow-start phase or congestion avoidance phase, when either $x(k)<A_{th(k)}$ or $x(k)\geq A_{th(k)}$, respectively. For example, in Fig. 1, the total number of packets transmitted is $x(1)$ until the first loss ① and the possible number of packets to transmit is $A_{th(1)}$ until $th(1)$. And since $x(1)<A_{th(1)}$, it is considered that the packet loss occurs during slow-start phase. Similarly, since the number of packets sent before the loss ② is $x(2)>A_{th(2)}$, it is determined that the packet loss occurs during congestion avoidance.

Mean response time for HTTP over SCTP is given as (2).

$$E(T_{sctp}) = \sum_{k=1}^{\alpha} \left[\beta E(T_{slow}^{k}) + (1-\beta)E(T_{cong}^{k})\right] + R \qquad (2)$$

Since the first packet loss (k=1) of SCTP in (2) occurs always during slow-start phase as shown in Fig. 1, $E(T_{slow}^{1})$ needs to be added. Packet losses after second one occur during either slow-start phase or congestion avoidance phase. $E(T_{slow}^{k})$ and $E(T_{cong}^{k})$ represent mean response time, when the k^{th} packet loss (k=2,3,...a) occurs during slow-start phase and congestion avoidance phase, respectively. Detailed computation procedures of $E(T_{slow}^{k})$ and $E(T_{cong}^{k})$ are presented in [14]. Since an arbitrary packet loss cannot occurs simultaneously during slow-start phase and congestion avoidance phase, β is either 0 or 1 for the given k^{th} packet loss. That is, if k^{th} packet loss occurs during slow-start phase and β=1, then $E(T_{sctp})$ is accumulated by adding $E(T_{slow}^{k})$. Similarly, if k^{th} packet loss occurs during congestion avoidance phase and β=0, then $E(T_{sctp})$ is accumulated by adding $E(T_{cong}^{k})$. Therefore the total mean response time of an object needs to add either $E(T_{slow}^{k})$ or $E(T_{cong}^{k})$ (k=1,2,...,a) as the expected value of lost packet number (a). R, which is the time to transfer the remained data, $N(a+1)$ after the last packet loss occurred, can be calculated without considering additional packet losses since the expected value of packet losses is already equal to a. That is, if $N(a+1)$ is less than the possible amount of data to transfer until the last threshold $th(a+1)$, the transmission is completed during slow-start phase. Therefore R is sum of slow-start time ($ST(N(a+1))$) and transmission time ($N(a+1)\times mtu/\mu$) until then. μ represents the bandwidth of the link. Otherwise the transmission is completed during congestion avoidance phase. Thus R is sum of slow-start time ($ST(A_{th(a+1)})$) and transmission

 ISBN: 978-1-61208-263-9

time ($N(a+1)\times mtu/\mu$) until the threshold adding the extra time ($(N(a+1)-A_{th}(a+1))\times rtt$) in congestion avoidance phase.

III. MEAN WAITING TIME MODEL FOR MULTIPLE USERS

The mean response time of HTTP over SCTP ($E(T_{sctp})$) found in the previous section is total time for a user to connect to a web server and download an object. Mean waiting and turnaround time are defined as the performance measure when multiple users access the web server simultaneously.

We assume the asynchronous TDM (time division multiplexing) based on packet for web service. A web object consists of n packets, thus, packet response time (τ) is equal to $E(T_{sctp})/n$ when every τ is the same. Also, n is given by $\lceil \theta/mtu \rceil$. Now, if we assume that four clients (m=4) request the same file, each user's expected response time ($E(T_{sctp})$) will be the same. For example, we consider the case where n=3 with the asynchronous TDM. When a client requests an object from the server, three packets are included in the object. $E(T_{sctp})$ means total response time that each client expects.

Now, we develop analytical models for the mean waiting and turnaround times for two cases depending on whether the packet response times are same or not.

When the web servers are connected to the external users through only one link, the total waiting time, the mean waiting time(W_{sctp}^{same}), total turnaround time, and mean turnaround time(T_{sctp}^{same}) are given by the following equations:

$$total\ \ waiting\ \ time = \sum_{i=1}^{m}(m-i)\tau + m(n-1)(m-1)\tau \tag{3}$$

$$W_{sctp}^{\ same} = \frac{\sum_{i=1}^{m}(m-i)\tau + m(n-1)(m-1)\tau}{m} \tag{4}$$

$$total\ \ turnaround\ \ \ time = m\tau[m(n-1) + \frac{m+1}{2}] \tag{5}$$

$$T_{sctp}^{same} = \frac{1}{m}[m(n\text{-}1) + \frac{m+1}{2}]\tau = [\frac{2mn\text{-}m+1}{2}]\tau \tag{6}$$

When the web servers are connected to the external users through several links of different bandwidths, the mean waiting and turnaround time are given by (7) and (8) respectively. First, we consider the mean waiting time. To find the waiting time of i^{th} user, we divide the total time into two intervals: the first interval represents the time when all the packets except the last packet of each user has been received; the second interval represents the time when the last packet of each user has been received. Total waiting time of i^{th} user until the first interval is (the number of packets–1)×[(the number of users for group including i^{th} user–1)×τ_i+(total packet response time excluding i^{th} group)]. The waiting time of i^{th} user is the sum of response times of other users prior to him. By generalizing and adding this all, we obtain the following equation for the mean waiting time. Both m_0 and τ_0 are zeros in the equation.

$$W_{sctp}^{\ diff} = \frac{(n-1)\sum_{i=1}^{P} m_i[(m_i-1)\tau_i + \sum_{i=1, j\neq i}^{P} m_j\tau_j]] + \sum_{i=1}^{P}[\sum_{j=1}^{i} m_i(m_{j-1}\tau_{j-1}) + \sum_{j=1}^{m_i}(j-1)\tau_i]}{m} \tag{7}$$

Now, we consider the mean turnaround time. If we use the same procedure as the waiting time, total turnaround time of i^{th} user until the second interval is (the number of packets–1)× [the number of users (m_i)×the sum of packet response time (τ_i)]. The turnaround time of any user in the second interval is the sum of response times of other users prior to him and his own packet response time. Thus, by generalizing and adding this all, we obtain the following equation. Both m_0 and τ_0 are zeros in the equation.

$$T_{sctp}^{\ diff} = \frac{m(n-1)\sum_{i=1}^{P} m_i\tau_i + \sum_{i=1}^{P}[\sum_{j=1}^{i} m_i(m_{j-1}\tau_{j-1}) + \sum_{j=1}^{m_i} j\tau_i]}{m} \tag{8}$$

IV. PERFORMANCE EVALUATION

Based on the model discussed in section 2 and 3, we can construct an algorithm for the whole procedure as in Algorithm 1 (Fig. 2). When the number of packets for an object is n, the complexity of the algorithm is $O(n)$.

We consider a simulation of web server for TCP and SCTP, and an environment to emulate HTTP. Desktop computers are used as client-server to send data. In order to simulate real network, we use a laptop computer with NIST emulator [15] between a client and a server, and adjust various network conditions such as packet loss (p), bandwidth (μ), and RTT (rtt).

Algorithm 1. mean waiting and turnaround time for multiple users

```
Begin
Compute the total number of packets in object
    (n = ⌈θ/mtu⌉)
Compute the expected number of packet loss (α = ⌈np⌉)
Set N(1) = n and th(1) = ∞
Set E(T_sctp) = 0
for all k such that k=1,2…, α do
      Find E(T_slow^k) and E(T_cong^k)
end for
Find the mean response time, E(T_sctp)= E(T_sctp) + R
Find the packet response time, τ = E(T_sctp) / n
If τ is same for all bandwidth type i,
      Find mean waiting ( W_sctp^diff ) and turnaround time using
      (4) and (6) respectively.
else
      Find mean waiting and turnaround time using (7) and
      (8), respectively.
endif
End
```

Figure 2. mean waiting and turnaround time for multiple users

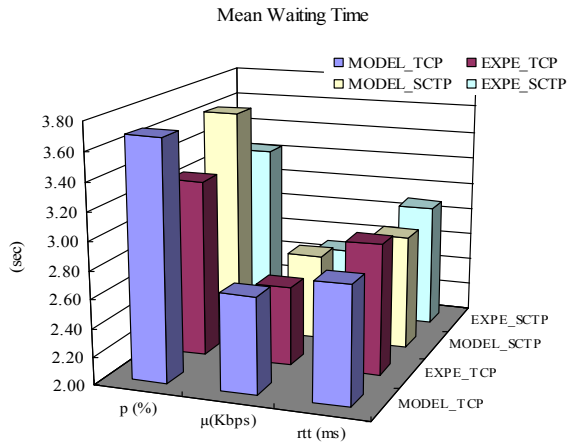


Figure 3. Mean waiting times for p, μ, rtt

Except the number of initial windows, HTTP over TCP model is basically same as HTTP over SCTP. That is, except that mean response time ($E(T^1_{slow})$) for the case of first packet loss occurred in slow-start phase of Algorithm 1 is computed differently, the procedures are same. Mean object size (θ) is 13.5KB and maximum transmission unit (mtu) is 536B. A HTML file contains five web objects.

Our experiments were performed as follows: Firstly, we changed p from 0.4 to 2% after fixing rtt=256ms and μ=40Kbps. Secondly, we changed μ from 400Kbps to 3Mbps after fixing p=1% and rtt=256ms. Finally, we changed rtt from 55ms to 256ms after fixing p=1% and μ=40Kbps.

Fig. 3 depicts the summary of mean waiting times (sec) for each p, μ, rtt. In the figure, MODEL_SCTP and EXPE_SCTP represent W^{diff}_{sctp} and T_{sctp}, respectively. MODEL_TCP and EXPE_TCP also represent W^{diff}_{tcp} and T_{tcp}, respectively. Fig. 3 shows that both models for HTTP over SCTP and HTTP over TCP overestimate mean waiting times for p and μ, respectively, but, models underestimate them for rtt.

Now, we define the mean difference ratio between models and experiments by (9).

$$DIFF_{mean} = \sum_{i=1}^{n} \left[\frac{W^{diff}_{sctp} - T_{sctp}}{W^{diff}_{sctp}} + \frac{W^{diff}_{tcp} - T_{tcp}}{W^{diff}_{tcp}} \right] / n \times 100 \qquad (9)$$

The computed $DIFF_{mean}$ is 4.17%. This value is small; however, more experiments and model adjustments are necessary to describe the real environment exactly. Additionally, we find that the mean waiting time of HTTP over SCTP is less than HTTP over TCP on both the model and experiment.

V. CONCLUSIONS

This paper presents an analytical model to estimate mean waiting time of web service using HTTP over SCTP in the narrowband network when multiple users access web server simultaneously. We first describe the mean response time model for single user, which is one of QoS offered to web users and one of essential parameters to evaluate web performance. We then extend the mean response time model to the mean waiting and turnaround time models for multiple users. Simple test-bed simulation results show that the mean difference ratio, between the analytical model and experiment, is small. Further extension of this work includes model with higher accuracy for the real environment.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A4A01003651)

REFERENCES

[1] V. Paxon, M. Allman, and W. Stevens, "TCPs congestion control," RFC 2581, 1999. http://www.ietf.org/rfc/rfc2581.txt.

[2] R. Stewart, "Stream control transmission protocol (SCTP), RFC 4960, 2007. http://www.ietf.org/rfc/rfc4960.txt.

[3] L. Budzisz, J. Garcia, A. Brunstrom, and R. Ferrus, "A Taxonomy and Survey of SCTP research," ACM Computing Surveys, vol. 44, no. 4, 2012, pp. 18:1-18:36.

[4] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Reno performance: A simple model and its empirical validation," ACM Transactions on Networking, vol. 8, no. 2, 2000, pp. 133-145.

[5] N. Cardwell, S. Savage, and Y. Anderson, "Modeling TCP latency," Proceeding of the 2000 IEEE Infocom Conference, 2000, pp. 1742-1751.

[6] Z. Jiong, Z. Shu-Jing, and Qi-Gang, "An adapted full model for TCP latency," Proceedings of. the 2002 IEEE TENCON Conference, Vol. 2, 2002, pp.801-804.

[7] D. Oliveria and R. Braun, "A dynamic adative acknowledgement strategy for TCP over multihop wireless networks," Proceedings of the IEEE INFOCOM Conference, 2005, pp.1863-1874.

[8] Lin-Huang Chang, Ming-Yi Liao and De-Yu Wang, "Analysis of FTP over SCTP in Congested Network," 2007 International Conference on Advanced Information Technologies (AIT), 2007, pp. 82-89.

[9] Chia-Wen Lu and Quincy Wur, "Perfomance study on SNMP and SIP over SCTP in wireless sensor networks," 14th International conference on advanced communication technology (ICACT), 2012, pp. 844-847.

[10] Fei Ge, Liansheng Tan, Jinsheng Sun, and Moshe Zukerman, "Latency of fast TCP for HTTP transactions," IEEE Communications Letters, vol. 15, no. 11, 2011, pp. 1259-1261.

[11] J. Eklund, K. Grinnemo, A.Brunstrom, G. Cheimonidis, and Y. Ismailov, "Impact of Slow Start on SCTP Handover Performance," Proceedings of the 20th international conference on computer communications and networks, 2011, pp.1-7.

[12] Y. Lee, M. Atiquzzaman, and S. Sivagurunathan, "Mean response time estimation for HTTP over SCTP in wireless environment," Proceedimgs of the 2006 IEEE International Conference on Communications, 2006, pp.164-169.

[13] Y. Lee and M. Atiquzzaman, "Mean waiting delay for web object transfer in wireless environment," Proceedings of the 2009 IEEE International Conference on Communications, 2009, pp.1-5.

[14] Y. Lee, "Mean response delay estimation for HTTP over SCTP in wireless Internet," Journal of the Korea Contents Association, vol. 8, no. 6, 2008, pp. 43-53.

[15] Mark Carson and Darin Santay, "NIST Net – A Linux-based Network Emulation Tool," ACM SIGCOMM Computer Communication Review, vol. 33, no. 3, 2003, pp. 111-126. http://snad.ncsl.nist.gov/itg/nistnet

 ISBN: 978-1-61208-263-9

Management Driven Multicast Protocol with End-to-End QoS

Radu Iorga
Telecommunication Dept.
University POLITEHNICA of Bucharest
Bucharest, Romania
radu.iorga@elcom.pub.ro

Eugen Borcoci
Telecommunication Dept.
University POLITEHNICA of Bucharest
Bucharest, Romania
eugen.borcoci@elcom.pub.ro

Radu Dinel Miruta
Telecommunication Dept.
University POLITEHNICA of Bucharest
Bucharest, Romania
radu.miruta@elcom.pub.ro

***Abstract*—Multicast technologies supporting various media services are increasingly seen in the current Internet and are expected to be used also in future Internet deployments. While traditional IP level multicast and overlay multicast are well known solutions, multi-domain multicast with quality of services (QoS) guarantees is still a research topic. This paper proposes a management driven hybrid multicast system and protocol, QoS enabled and spanning multiple IP domains. Starting from a previously defined architecture, the management system is developed and then, the protocol design, implementation and some performance evaluation are presented.**

Keywords-hybrid multicast; overlay, quality of services; virtualization; multiple domains; service level specification.

I. INTRODUCTION

Increasing demand for multimedia content distribution, while satisfying different levels of quality of services (and quality of experience for users), reinforced the interest for multicast technologies,. While IP level multicast is highly efficient it has not been largely deployed in multi-domain environments. On the other hand, the overlay (application layer) multicast is easier to be deployed, but is less efficient and does not exploit the IP native multicast capabilities where they exist [1]. For multi-domain multicast , one has also to consider that each domain is managed by an independent Network Provider (NP), or operator. Therefore, a *management driven multicast hybrid solution* seems to be attractive. This is true especially if QoS and flow distribution process supervision are wanted to be performed by the management, in order to fulfill requirements of *Service Level Agreements (SLA)* negotiated and agreed between the NPs and the multicast services users/clients.

The work [2] proposed an architecture of such a hybrid multicast framework which is QoS capable, where IP level intra-domain multicast is combined with inter-domain overlay multicast. This paper has started from the architecture in [2] and here the multicast management system is further developed. Then, the main contribution of this work is the *Management Driven Multicast Protocol (MDMP),* based on a powerful algorithm performing jointly a constrained QoS routing, resource reservation, and multicast tree mapping onto multi-domain network topologies, under control of a distributed management system. The specification, design and implementation of the protocol and also some performance evaluation have been accomplished and presented in the paper. The protocol offers a solution for real-time multimedia applications like IPTV, VoD in a multi-domain multi-provider scenario.

The multicast system discussed in this paper is currently under development in the European FP7 ICT research project, "Media Ecosystem Deployment Through Ubiquitous Content-Aware Network Environments", ALICANTE,[2][3].

The paper is organized as follows. Section II presents samples of related work. Section III introduces the MDMP, showing a high-level view and its main rationale. Section IV is focused on the most important design and implementation issues, including the protocol itself but also the algorithm used for multicast tree construction. Section V contains conclusions and outline of future work.

II. RELATED WORK

Several multicast solutions and protocols are already specified in IETF RFC documents and part of them are implemented and produced by equipment manufacturers. A comprehensive overview of multicast solutions is presented in [4].

However, in this study we are focus on adapting the chosen solution to the general multi-domain architecture defined in [2], [3]. There, *virtual content aware networks* with *guaranteed QoS* and *unicast/multicast enabled* - have to be constructed on top of multi-domain IP infrastructure, under management of several virtual network managers and network managers - the latter being aware of actual network topology and resources. Basically three solutions can be analyzed: IP level, overlay level and application level multicast (ALM). The latter is excluded from our scope given that it is performed in the end-host machines, while we need the network support. Additionally it is shown in [4] that IP multicast and overlay multicast (based on special nodes in the network) have better trade-offs between multicast tree cost and end-to-end delay than ALM.

The PAM protocol ("Adaptive Hybrid Multicast with Partial Network Support") defined in [5] combines the advantages of IP multicast with the ones of *ALM* and reiterates the idea of connecting native IP multicast islands with the use of IP-in-IP tunnels. Just as in the case of AMT

 ISBN: 978-1-61208-263-9

[6] this approach considers the IP multicast tree already constructed and does not offer any QoS support on the tunnels used between *m-routers*. The HOME (Host group based Overly Multicast Environment) approach defined in [7] takes the idea of ALM to move the multicasting towards the end nodes but relocates the terminations of the tree into the Designated Routers (DR) leaving the communication between DR and end-nodes as native IP multicast. The ALM is than created with the DR playing the roles of end nodes.

In [8], several QoS multicast solutions are analyzed. However, QoS constrained routing is not sufficient in our case, given that QoS guaranteed multicast connectivity services are needed. Therefore, some resource reservation is necessary. A QoS extension for OSPF (*QOSPF*) has been proposed in [9] and completed in [10] with multicast extensions. The solution is not appropriate, given the necessity of resource reservation.

In the unidirectional core-based trees the existence of a central node, which might not be on a QoS path between source and receiver, raises even more challenges than in *Shortest Path Tree (SPT)* cases. The best thing that can be done in this case is to try to assure local QoS and not end-to-end [8]. The major core-based protocol, *PIM-SM* [11] has no QoS extensions.. However, the fact that it depends on the underlying unicast routing protocol, makes it good candidate to support QoS constraints. PIM-SM is the main multicast protocol in [12] which proposes a hybrid multicast architecture where IP multicast is used only in the last domain where the receivers are. More, the solution offers QoS guarantees only if the unicast routing protocol used by PIM-SM can offer any. A hybrid multicast system is also proposed in [13] to create an E-Cast tree composed of unicast QoS pipes for inter-domain and use of PIM-SM inside the domains as opposed to MDMP which creates two combined multicast trees with QoS.

The work [14] proposes a QoS framework for a multi-domain multicast service. The QoS control is performed by choosing the best available inter-domain multicast route in order to respect end-to-end connection requirements. A Multicast Inter-Domain Entity is introduced in each network domain to search and test multiple paths from domains already in the multicast group (tree-domains) towards a new joining-domain. The difference of our solution is that we take benefit from existence of Virtual Network Manager and also Network Manager in each domain and the multicast tree computation is performed there so no other entity is needed.

The paper [15] presents a system called Multi-servIce Resource Allocation (MIRA), where a multicast-aware resource reservation protocol for class-based networks that consider routing asymmetries is proposed. MIRA agents are distributed in the network. The difference of our solution is that we do not need MIRA agents inside the network and the trees are not per session based but the multicast tree life is that of a virtual network to which it is associated.

III. MANAGEMENT DRIVEN MULTICAST PROTOCOL

This section will present additional motivations to define the *MDMP* framework. Then the principal architectural features of the MDMP are described. Its name comes from the fact that there is a central manager that computes the tree, based on input data, and programs each router along the tree with the computed information.

An important MDMP usage is in *virtualization based on overlays,* which is seen today as a major method to make the Internet (and also Future Internet) more flexible, [16],[17], [18] by slicing it in parallel isolated planes. Towards this aim, *Network Infrastructure Providers (NIP)* can offer their sliced resources to some *Virtual Network Provider (VNP)* which constructs *Virtual Networks (VNet)* by merging several network infrastructure resources. Each NIP, while cooperating to the VNet, still manages independently its resources and maintains knowledge on their availability. A multicast capable VNet needs a multi-domain tree, where each domain has to construct its own part of the tree.

General *virtual networks mapping* [16],[17],[18] is needed, *abstracting the subset of network resources* (links, nodes). An overlay multicast VNet seems to be the first approach. However it is better to benefit in some domains by native IP multicast capabilities, therefore a *hybrid solution* is more appropriate and so is adopted in this work.

The VNets are *created on demand,* by some *generalised "VNet Users" or Requestors" using a VNP* management framework. The VNP management will perform the multicast VNet planning, advertising, offering, negotiation, provisioning, and commands their operation (installation, modification, manipulation, monitoring, and termination), while cooperating with NIPs. A VNP might be seen as containing one or several VNet Managers (VNetMgr) which at their turn can be associated in one to one relationship with each Net Manager (NetMgr) of the NIP. This approach satisfies the requirement of having distributed management and control by avoiding a single central manager. Creation of *multicast VNets* fulfilling QoS requirements needs vertical and horizontal *negotiations* and SLA concluded between entities while preserving each domain's resource management independence.

VNets optimized mapping [19], is necessary onto multi-domain substrate. This is supported by the MDMP which combines several functions: constrained routing, admission control for resource reservation, and final tree mapping combined with inter-domain QoS-enabled routing and resource reservation [19]. An appropriate metric should be defined for tree construction to be used in the selection of tree paths and then a reservation action is performed at the level of VNetMgr and respectively NetMgr. An additive metric incorporating QoS needs has been defined. Scalability of the solutions is also necessary.

The result of the above considerations is the architectural solution shown in Fig. 1, where MDMP constructs the multicast VNets. The actors involved are: *Multicast VNet requester – (*e.g., a high level Service

Provider); *Multicast VNet Manager (MVNetMgr)* –the block actually computing and controlling the tree installation. For each tree there is a MVNet Manager, which is the initiator of the process and this is seen as a central management node with enough information (topology, capacities, etc.) and political rights to take all the construction of the tree in its responsibility; *Network Manger* that manages a single domain resources and in particular has the responsibility to map its part of the multicast tree on its real topology.. Also at the request of MVNet Manager, it commands the routers to install the tree. Assuming that VMnetMgr knows the topology of the network, a modified Dijkstra algorithm is used to compute parts of the tree. The existence of a source-based tree means that *Source Specific Multicast (SSM),* [20] should be used as addressing scheme. This solves any issues with group address management.

The major steps of *MDM* algorithm are: *1.Get request and topology; 2. Compute SPT on the graph after removing the link that do not meet the QoS constraint; 3. Enforce the decision of the algorithm (config. multicast routing tables and do the actual reservation of resources in the routers).*

IV. DESIGN AND IMPLEMENTATION

The MDMP system has been implemented as part of Alicante work and installed on a pilot testbeds (Fig. 2). It consists in three fully meshed domains, with all nodes as Linux routers with IP multicast and QoS support enabled. In this example, the managers are located in the same physical machine as the routers (see C12 in domain1) but the communication between different entities is made over TCP connections allowing physical separation.

MDMP can deal with any topology provided that there is some entity able to find and disclose it (discovery mechanism is out of this paper's scope). For the ease of portability, an xml format has been defined to represent the topology:

<ip>141.85.43.124</ip>
<intf>eth1</intf>
<nip>141.85.43.130</nip>
<bwd>3</bwd>

The implementation is made in C under Linux, hence the name of the interfaces have the Linux format. The topology should be read like this: Node .124 has a neighbor node .130 reachable through interface *eth1* with available bandwidth of *3 Mb/s*.

Only bandwidth is used in this implementation as being the easiest case, but in a form of any additive metric [19] (cost of a link is 1/bandwidth). This representation of topology can be used for both inter-domain and intra-domain, even though in inter-domain case the *<ip>* represents the MVNetMgr IP while in intra-domain the *<ip>* represents the router IP.

An open-source library called *libxml2* [23] is used to read and parse the xml file describing the topology. Each node is given a unique ID. As this is a multithreading environment *mutex-es* are used to ensure the uniqueness of the IDs. The extracted data is kept in simple linked lists: one that keeps , all the nodes and each node has a linked list with its neighbors (reduced structures are presented):

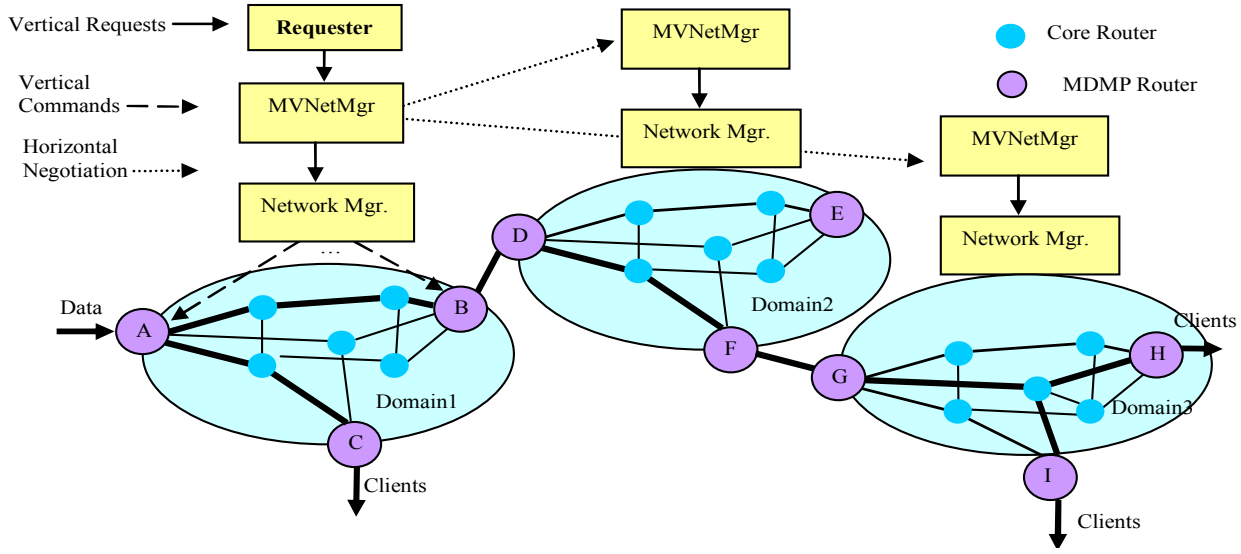


Figure 1. A MDMP typical architecture

```
/*Linked list holding ALL nodes in topology*/
struct mcast_node {
        int   node_id; //The ID of the node.
        char  node_ip[IP_MAX_SIZE]; // Node's IP
        struct neighbor   *neigh_head; // Head of neighbors
                                       linked list
           ... //some other internal variables
           struct mcast_node *next; //next node.};
  /* intf struct: A linked list holding ALL neighbors*/
struct neighbor {
```

```
char intf_name[INTF_NAME_SIZE]; //
char    intf_id; // intf id : eth0->0
int     bwd; // Bandwidth to neighbor
char    neigh_ip[IP_MAX_SIZE];
 ... //some other internal variables
struct neighbor *next; // next neigh};
```

This is just a proof of concept on a small size testbed, hence the performance is not very stringent. However, in a real life implementation, advanced data structures can be used to speed up things. For instance, now if we need to lookup a node, a *O(n)* complexity will be achieved, but if, in an advanced implementation, a hash table is used and the hashing function is correctly chosen, a complexity of *O(1)* can be obtained.

A. Inter-domain MDMP

Message Sequence Chart (MSC) is presented in Fig. 3 showing the MDMP signaling to build the multicast tree spanning multiple domains. The trigger to MDMP is a request for a QoS assured multicast tree and it must contain the source IP of the tree (Src_IP), the desired QoS (only bandwidth in our implementation) and the receivers (routers M13, M22, M23 and C32 in Fig. 2). We consider a request for 3 MB/s in this example. Several requests can be handled because we have a separate thread for each new tree requested and semaphores to protect the data.

The MVNetMgr receiving this requests will be called the *Initiator* and it will execute some preliminary checks to determine if the tree can be accepted or some negotiations are needed (*Neg_req()* and *Neg_rsp()*). The next step is to build the Overlay Tree which must start with determination of the domains with receivers: in our implementation, where the domain is represented by the MVNetMgr IP, this kind of mapping is stored in a database. It is outside the MDMP scope how this mapping is obtained.

In Fig. 2 the mapping will mean that for M13 the IP of MVNetMgr1 will be found, for M22 and M23 the IP of MVNetMgr2 will be found and so on (action 1 at MVNetMgr1). The second action is to pick a Group IP address. Note that we have also the source of traffic so the pair (Src_IP, Grp_IP) is unique and will be referred to as (S,G). At this point we have a graph with domains as nodes, we have the receiving domains and we have the 3Mb/s constraint. First of all, the non-compliant links are

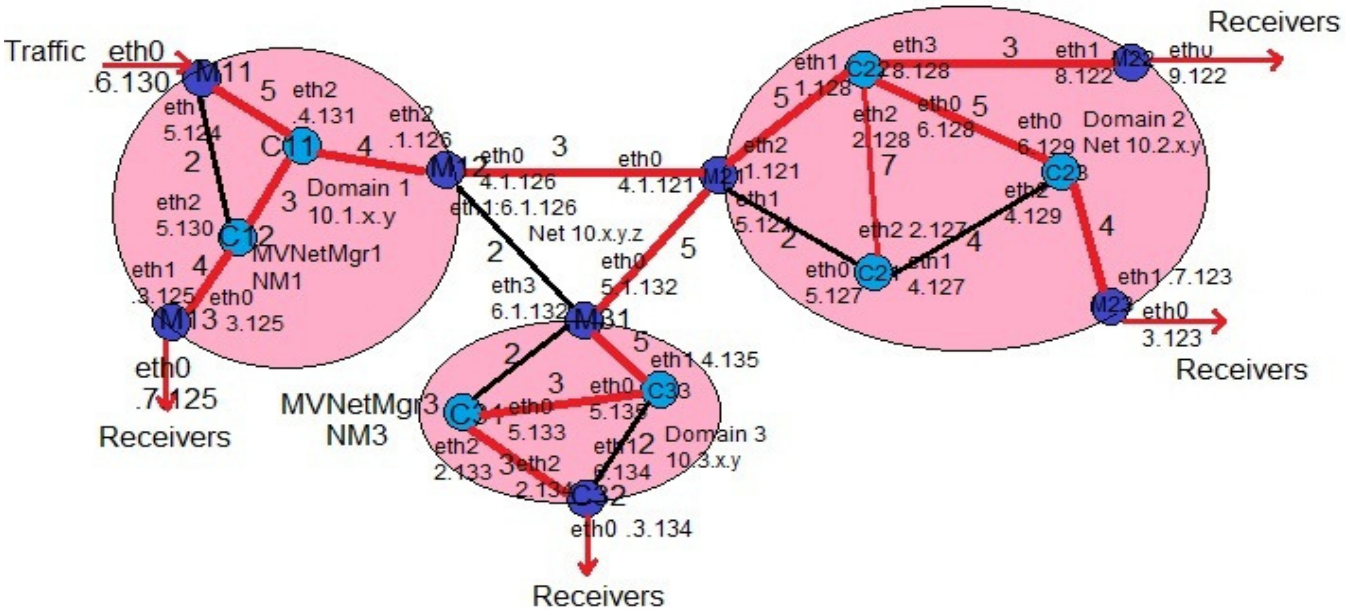


Figure 2. The testbed topology

eliminated (i.e., link between M12 and M32) and based on the new graph and using Dijkstra algorithm the Shortest Path Tree (SPT) is computed (action 3). Details of the algorithm itself are presented in section C. But SPT is covering all nodes in a graph so we have a special API that traverses the graph and prunes the nodes that have no receivers or are not on the path to any receiver. At this point we have the Overlay Tree, so the Initiator will now try to contact each domain, including own domain, and request the IP multicast tree. As there might be many domains we developed a multithreading environment using Linux pthreads. For each domain belonging to the Overlay Tree we create a pthread to handle the negotiation. After all the signaling is done all threads are terminated to save CPU resources.As shown in Fig. 3, a *Req_tree()* is sent to MVNetMgr2 and MVNetMgr3. The request sent to other MVNetMgr are similar to the one received from the SP. However the receivers are different: the request sent from MVNetMgr1 to MVNetMgr2 has as source the IP of MVNetMgr1 and one of the receivers is now MVNetMgr3.

 ISBN: 978-1-61208-263-9

Using the mapping database, MVNetMgr1 will determine that the source of the IP multicast tree is M21 and that M21 is also a receiver as it is the route towards MVNetMgr3 which is a receiver. But this is normal as different interfaces of M21 are used as source and receiver. Apart from the Initiator all the other MVNetMgr have fewer actions to perform: get the mappings from database and just relay the request to own NM. The *Req_tree()* message from MVNetMgr2 to NM2 is very similar to the one from MVNetMgr 1 to MVNetMgr2. The main difference is that now the MVNetMgr IP is replaced with the needed router IP. The order of actions is presented sequential in the MSC to make it easier to understand, but in reality it is hard to know the exact order due to the multithreading behavior of our solution. All the above communication is realized over a TCP socket in order to make it reliable. The socket is created using the IPs involved and predefined TCP ports:

#define ROUTER_PORT 39393 // to enforce on the routers
#define INTRANRM_PORT 19191 // to request from own domain
#define CANMGR_PORT 29292 // to negotiate with other domains

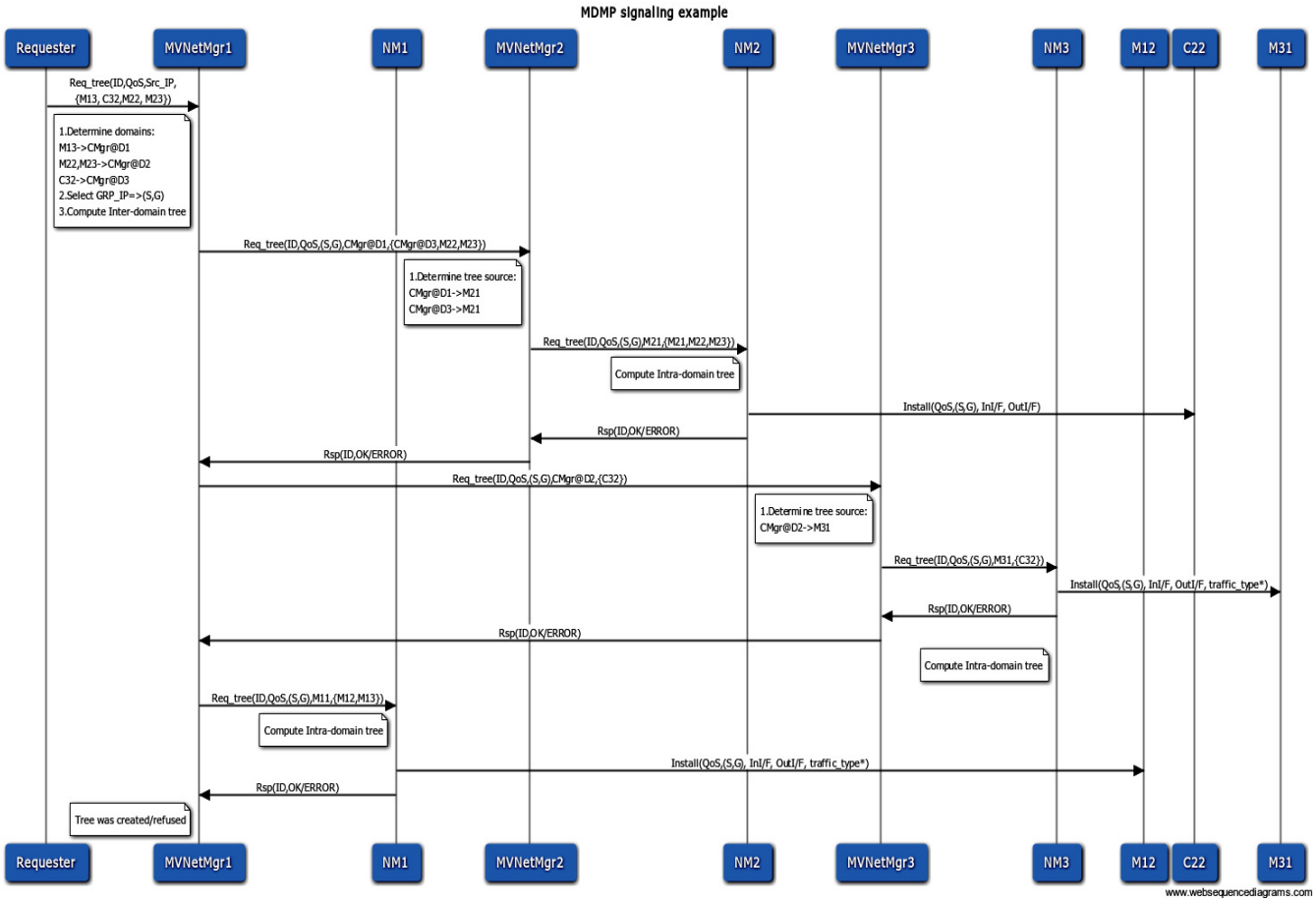


Figure 3. Control messages for inter and intra domain

B. *Intra-domain MDMP*

Switching focus to NMx will duplicate most of the actions presented in inter-domain scenario at intra-domain level. From the protocol point of view, a request is received and the same algorithm from section C is used. We reiterate the discussion about the difference between SPT and the multicast tree in order to emphasize in Fig. 2 a case where a node (C21) is part of the SPT (thin red line) but, as it has no receivers nor is on the path to any receiver, is pruned from the actual multicast tree (thick red lines). The NM will have to enforce the result on the routers. We present the three most relevant moments to install the tree: right after the tree is computed; at a later time triggered by an Invocation message from the requester; at a later time specified from the beginning by the requester. We've implemented the first case as it offers the most testing possibilities.

For each router that needs instructions a separate thread is opened. In an intra-domain topology this might became a problem as there might be many routers and the CPU might get over-loaded. However, the processing needed in each thread is not very big, so the CPU overloading issue might not be a top priority issue. Details on router configuration are presented in SectionIV-D.

 ISBN: 978-1-61208-263-9

C. MDMP algorithm

Because the different approaches between unicast and multicast situations, for this last case we propose an enhanced version of the algorithm, more detailed presented in [19], but focused for this time on multicast. The new improvements here assume, apart from specific adaptations of multicast the concept of prioritizing requests..In order to alleviate the impact of the unsolved requests, a better situation will be if these requests will be the least important ones. All requests form the received set are grouped based on the source node and *group priority* is defined (lower value means higher priority). In the case of several groups with the same priority, the algorithm will permute the processing order obtaining the best cost. Based on our simulations results just after a few permutations the total cost is not decreasing significantly, so it is not need to run the n! permutations, where n is the number of different group with the same priority. To offer a maximum flexibility solution w.r.t. Requester interests, one should admit that the Requester can specify a priority for each individual request. The group priority has precedence over the individual request priority.

The used algorithm for unicast is a little bit more extensive that the current one for multicast in terms of requested bandwidth. For the multicast case, we should have the same bandwidth values for different requests even if they are part of the same group. Even if the requested bandwidth is the same for each request, the order of solving requests inside each group becomes important: honoring one request before the other can exhaust the available bandwidth for a segment. Other specific adaptation for the multicast case, compared with the unicast situation presented in [19] is the non using of the blind search (when constraints are applied on the STP found by the modified Djikstra algorithm, the path can take other way, through other nodes, different from the ones indicated by Djikstra). Leaving the STP found by the modificd Djikstra algorithm, takes us away from the multicast context. In order to obtain more accurate results we simulated a two-level hierarchical network with 31 ASes, each AS containing a maximum of 8 routers, totaling a number of 186 nodes using a dedicated tool for topology generation from Scilab [21].

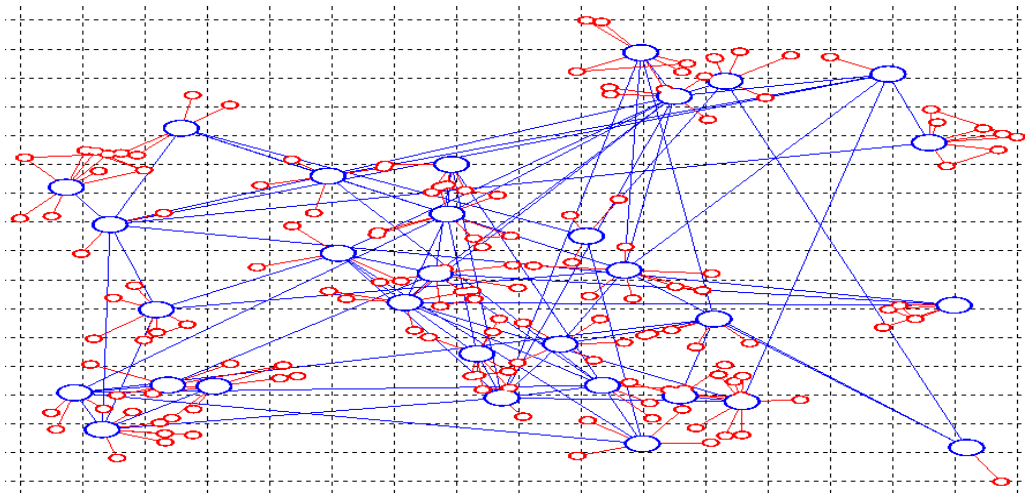


Figure 4. Two levels hierarchical network topology

The obtained topology is the one from Fig. 4 where the circles with larger diameter and their associated segments represents the inter-domain topology and the smaller diameter circles with their associated links are the intra-domains. Each segment for both inter and intra-domain areas has an associated bandwidth generated in respect to a Gaussian distribution centered in 100.

Because the NARVAL_T_HierWaxmanConD function from NARVAL [22] was created to generate a network graph with much more layers, we modified it for only two.

Each segment for both inter and intra-domain areas has an associated bandwidth generated in respect to a Gaussian distribution centered in 100. Because the NARVAL_T_HierWaxmanConD function from NARVAL [22] was created to generate a network graph with much more layers, we modified it for only two. The network backbone of size n (31 in our case) and thereafter the second layer added were created based on the Waxman model. The topology was generated using the following parameters values (more details about the significance of each parameter can be found at the NARVAL help):

a=0.4;//first parameter of the Waxman model
b=0.5;//second parameter of the Waxman model
n=31;//network backbone size
l=1000;//network squared area side
nl=8;//maximal quantity of nodes per subnetwork
db=25;//original diameter of nodes

 ISBN: 978-1-61208-263-9

dd=15;//diameter difference between successive network layers
cv=[2 5];//color of each network layer
[g,d,v]=NARVAL_T_2layers(a,b,n,l,nl,db,dd,cv);//application of NARVAL_T_HierWaxmanCon
And centered using the following script :
Lxmin=100; Lxmax=900; Lymin=100; Lymax=900;
nodex=g.node_x; nodey=g.node_y;
xmin=min(nodex); xmax=max(nodex);
ymin=min(nodey); ymax=max(nodey);
nodex=(Lxmax-Lxmin)/(xmax-xmin)(nodex-xmin)+Lxmin;*
nodey=(Lymax-Lymin)/(ymax-ymin)(nodey-ymin)+Lymin;*
g.node_x=nodex; g.node_y=nodey;
ind=1;//window index
f=NARVAL_G_ShowGraph(g,ind);//graph visualization
After this step we successfully extracted the adjacency matrix containing at this moment zeros (no link) and ones (presence of a link) and we created other scripts in order to insert in this matrix the generated bandwidth values attached at each link.

Simulation results
We created a set of 10 requests divided into 5 different groups with different priorities as in Fig. 5, this representing the Traffic Distribution Matrix.

1	1 75 31 1 2	6	7 100 31 1 2
2	1 115 31 1 1	7	7 142 31 1 1
3	1 19 31 1 3	8	14 112 25 3 1
4	5 78 27 2 1	9	14 18 25 3 2
5	5 121 27 2 2	10	21 175 55 4 1

Figure 5. Set of requests

Each line of the matrix specifies an individual request as *{source node, destination node, requested capacity, group priority, individual request priority}*.
Because of the multicast context each request from a group must have the same requested bandwidth value.
Using the adjacency matrix of the above topology and this set of requests as the input data of our algorithm it produces the following results (it was introduced only 2 permutations in case of groups with the same priorities):

==========
Input file multicast.in:
Request 1->115, carry 31: 1 14 15 4 22 16 115
Request 1->75, carry 31: 1 14 18 8 75
Request 1->19, carry 31 unsatisfied. Node unreachable
Request 7->142, carry 31: 7 17 15 24 23 142
Request 7->100, carry 31: 7 17 14 13 100
…..
Cost: 19.360294. Satisfied requests: 9 / 10

Request 7->142, carry 31: 7 17 15 4 23 142
Request 7->100, carry 31: 7 17 14 13 100
Request 1->115, carry 31: 1 14 6 23 4 22 16 115
Request 1->75, carry 31: 1 14 18 8 75
Request 1->19, carry 31 unsatisfied on 0->19. Node unreachable
……
Cost: 20.120386 . Satisfied requests: 9 / 10

Best cost: 19.360294
Satisfied Requests: 9 / 10
==========
Total time: 0.004000

The total processing time displayed here includes the time for printing functions which is quite significantly and it was obtained using a personal PC equipped with Intel Core 2 CPU, T5600@ 1.83 GHz, 2,00 GB RAM and a 32-bit OS. *Removing these printing functions, for the current input data, the processing time is too small for this six zeros granularity (0.000000)*
As it can be observed for both permutations only 9/10 requests were solved with a better cost associated for the first order. The request remained unsolved in this case is due to the situation of an unreachable node. Even if the network graph is constructed as a connex one, because of some optimization techniques used during the implementation (remove from graph all segments which do not respect the condition: avail_bwdb >= min_req_bwd value from the group) some nodes could become unreachable.
All requests were solved in respect with their group priorities and in respect with their individual priorities inside the group also.
It can be observed that we have 2 groups with the same priority (group 1 and 7) and because of this the algorithm performs 2 permutations, thus obtaining the best cost.

D. MDMP routers

There are two types of routers involved in MDMP: the core routers (any router with IP multicast and QoS capabilities), and the special edge routers, [2]. An ingress router receives unicast traffic and will send it as multicast; egress router receives either unicast or multicast, and will send unicast. An ingress router should determine the (S,G) tree the given packet belongs to. The incoming packet has as IP destination the IP of the egress router. So other type of information needs to be used, several of which have been proposed in Alicante [2]: the (SIP, DIP, proto, SrcPort ,DstPort) tuple configured by control plane; some special information inserted in the packet such as Content Aware Transport Information (CATI); NAL unit inside SVC header; deep packet inspection. The egress router has to change the IP destination of the packet to the IP of the directly connected edge router of the neighboring domain. This can be done by UDP tunneling the packet or by rewriting the destination address.
In Fig. 3 an *Install* message containing information about (S,G) and QoS is sent. For the edge routers the sequence of action is more complicated and has been described

 ISBN: 978-1-61208-263-9

above. But for the core routers, a simple system call to *smcroute* is made to install the routes into the kernel:

```
sprintf(cmd,"smcroute -a eth%d %s %s %s",
        r_cfg.in_intf, r_cfg.src_ip, r_cfg.grp_ip, oif);
system(cmd);
```

In our implementation we have used Linux and the traffic control tool *(TC)* for QoS, based on Hierarchical Token Bucket (HTB),.To apply the QoS requirements using *TC*'s HTB there are two steps: create the class of traffic based on the requested QoS (with the option of borrowing bandwidth if unused by other flows) and create the filter (based on (S,G) or the tuple above or any field in the packet) for the traffic to be classified inside the class. All traffic matching the filter should be guaranteed the traffic class. Using the same programming method as for adding the multicast routes, we create the strings needed for classes and filters based on received *Install* command parameters, and we make a system call to instruct the Linux kernel of our needs. Our implementation just tries to demonstrate that MDMP works. Of course that if the routers support more sophisticated QoS rules they can be applied with ease, as the protocol poses no restrictions.

V. CONCLUSION AND FUTURE WORK

The paper addresses the problem of building multicast QoS enabled trees spanning multiple domains. Some implementation details have been shown to prove that the concept. Scalability and performance aspects have been discussed. The MDMP trees are built under management actions that are supposed to be rare, so the response time of the protocol is not a constraint. These challenges towards a more scalable and performing software are in the front list of our future work plans related to MDMP. Further research is needed to solve in a timely manner any possible updates (prunes or grafts) to already installed trees.

ACKNOWLEDGMENT

This work was supported partially by the EC in the context of the ALICANTE project (FP7-ICT-248652) and partially by the project POSDRU/88/1.5/S/61178

REFERENCES

[1] H. Asaeda et.al., "Architecture for IP Multicast Deployment: Challenges and Practice", IEICE Transactions on Communications, Vol. E89-B, No. 4, 2006, pp. 1044-1051.

[2] E. Borcoci, G. Carneiro and R. Iorga**,** "Hybrid Multicast Management in a Content Aware Multidomain Network**",** AFIN 2011, The Third International Conference on Advances in Future Internet, pp. 90-95 http://www.thinkmind.org/index.php?view=article&articleid=afin_2011_5_20_70107

[3] ALICANTE, Deliverable D2.1, ALICANTE Overall System and Components Definition and Specifications, http://www.ict-alicante.eu/, Sept. 2011.

[4] Li Lao, J.-H. Cui, M Gerla, and D Maggiorini, "A Comparative Study of Multicast Protocols: Top, Bottom, or In the Middle?", Technical Report TR040054, 2005,Computer Science Department, UCLA,http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.59.4904 (last accessed March 4, 2013)

[5] H. Luo, K. Harfoush; "Adaptive Hybrid Multicast with Partial Network Support" - http://www4.ncsu.edu/~kaharfou/Papers/hybrid_multicast.pdf

[6] D. Thaler, M. Talwar, A. Aggarwal, L. Vicisano and T. Pusateri; "Automatic IP Multicast Without Explicit Tunnels (AMT)" – internet draft 2010 - http://tools.ietf.org/html/draft-ietf-mboned-auto-multicast-10

[7] D. Kim, Ki-Sung and Yu, A Scalable Hybrid Overlay Multicast Adopting Host Group Model for Subnet-Dense Receivers International Journal of Computer Science and Network Security(IJCSNS 2007).

[8] B Wang, J.C. Hou, "Multicast Routing and Its QoS Extension :Problems,Algorithms and Protocols", IEEE Network January/February 2000, pp. 22-36

[9] R. Guerin et al., "QoS routing mechanism and OSPF extensions" Internet draft – 1998 - http://tools.ietf.org/html/rfc2676.html

[10] C. Tseng, C. Chen, "Multicast Extensions to QOSPF", Internet and Multimedia Systems and Applications, EuroIMSA 2005, Grindelwald, Switzerland, pp. 370-376, February 21-23, 2005 IASTED/ACTA Press, 2005.

[11] B. Fenner, M. Handley, H. Holbrook and I. Kouvelas, RFC 4601 - Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification (Revised)

[12] R. Iorga, E. Boroci, and S. Obreja, "QoS Enabled IP multicast in a Multi-domain Multimedia Distribution System", Telecomunicatii 2008

[13] E. Borcoci, A. Pinto, A Mehaoua, L. Fang and N. Wang "Resource Management and Signalling Architecture of a Hybrid Multicast Service for Multimedia Distribution", Springer, Lecture Notes in Computer Science Volume 5274, pp. 39-51, 2008

[14] E. Guainella, C. Sansone, B. Angelini and N. Angelini, "The DAIDALOS approach to IP multicast, Inter-domain QoS control", http://www.eurasip.org/Proceedings/Ext/IST05/papers/524.pdf (last accessed March 4, 2013)

[15] A. Neto, E. Cerqueira, A. Rissato and E. Monteiro, "A Resource Reservation Protocol Supporting QoS-aware Multicast Trees for Next Generation Networks, Proc. of Computers and Communications, ISCC 2007. 12th IEEE Symposium on, pp. 707 – 714, 2007

[16] N.M. Mosharaf Kabir Chowdhury and R. Boutaba, A survey of network virtualization, Computer Networks, Volume 54, Issue 5, 8, pp. 862-876, April 2010, ISSN 1389-1286, 10.1016/j.comnet.2009.10.017.

[17] T. Anderson et al, "Overcoming the Internet Impasse through Virtualization", Computer, vol. 38, no. 4, pp. 34–41, Apr. 2005.

[18] N. Wang; et al; , "A two-dimensional architecture for end-to-end resource management in virtual network environments," Network, IEEE , vol.26, no.5, pp. 8-14, September 2012.

[19] R. Miruta, E. Borcoci and E. Pallis - Planning and Provisioning of Virtual Content Aware Networks over IP Infrastructures – 2012 – TEMU, IEEE Conference Publications pp. 118-123 http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6294701&contentType=Conference+Publications&sortType%3Dasc_p_Sequence%26filter%3DAND%28p_IS_Number%3A6294693%29%26rowsPerPage%3D50

[20] H. Holbrook, B. Cain (August 2006). RFC 4607: Source-Specific Multicast for IP - http://tools.ietf.org/html/rfc4607

[21] http://www.scilab.org/

[22] http://atoms.scilab.org/toolboxes/NARVAL/1.0

[23] http://www.xmlsoft.org/

 ISBN: 978-1-61208-263-9

On How to Provision Quality of Service (QoS) for Large Dataset Transfers

Zhenzhen Yan, Malathi Veeraraghavan
Dept. of Electrical and Computer Engineering
University of Virginia
Charlottesville, VA, USA
Email: {zy4d,mvee}@virginia.edu

Chris Tracy, Chin Guok
Energy Sciences Network (ESnet)
LBNL
Berkeley, CA, USA
Email: {ctracy,chin}@es.net

***Abstract*—There is recent interest in using traffic-engineered, QoS-controlled paths for large-sized, high-rate dataset transfers in the scientific community. We refer to TCP flows created by such transfers as α flows. Research-and-education network providers are interested in intra-domain traffic engineering systems for identifying α flows at ingress routers within their networks, and redirecting them to traffic-engineered paths. This is primarily because of the adverse effects these α flows have on delay-sensitive multimedia flows. The focus of this work is to determine what QoS mechanisms are suitable to achieve the dual goals of preventing α flows from adversely affecting delay-sensitive flows, while simultaneously allowing them to enjoy high throughput. The interaction between policing schemes on the ingress interfaces and scheduling schemes on the egress interfaces was studied through a set of experiments on a high-speed router testbed. Our conclusions are that a scheduling-only mechanism, with no policing, is well suited to achieve these dual goals if the level of fairness offered by today's IP-routed service is sufficient for simultaneous α flows.**

***Keywords*—policing; scheduling; high-speed networks; traffic-engineering; virtual-circuit networks**

I. INTRODUCTION

For large-sized scientific dataset transfers, scientists typically invest in high-end computing systems that can source and sink data to/from their disk systems at high speeds. These transfers are referred to as α flows as they dominate other flows [1]. They also cause increased burstiness, which in turn impacts delay-sensitive real-time audio/video flows. In prior work [2], we proposed an overall architecture for an intra-domain traffic engineering system called Hybrid Network Traffic Engineering System (HNTES) that performs two tasks: (i) analyzes NetFlow reports offline to identify α flows, and (ii) configures the ingress routers for future α-flow redirection to traffic-engineered QoS-controlled paths. The prior paper [2] then focused on the first aspect, and analyzed NetFlow data obtained from live ESnet routers for the period May to Nov. 2011. The analysis showed that since α flows require high-end computing systems to source/sink data at high speeds, these systems are typically assigned static global public IP addresses, and repeated flows are observed between the same pairs of hosts. Therefore source and destination address prefixes of observed α flows can be used to configure firewall filter rules at ingress routers for future α-flow redirection. The effectiveness of such an offline α-flow identification scheme was evaluated with the collected NetFlow data and found to be 94%, i.e., a majority of bytes sent in bursts by α flows would have been successfully isolated had such a traffic engineering system been deployed [2].

The work presented here focuses on the second aspect of the HNTES by addressing the question of how to achieve α-flow redirection and isolation to traffic-engineered paths. Specifically, service providers such as ESnet [3] are interested in actively selecting traffic-engineered paths for α-flows, and using Quality-of-Service (QoS) mechanisms for policing and scheduling these flows. With virtual-circuit technologies, such as MultiProtocol Label Switching (MPLS), ESnet and other research and education network providers, such as Internet2, GEANT2, and JGN-X, offer a dynamic circuit service. An On-Demand Secure Circuits and Advance Reservation System (OSCARS) Inter-Domain Controller (IDC) [4] is used for circuit scheduling and provisioning. To support inter-domain (virtual) circuits, an IDC Protocol (IDCP) [5] is being standardized. The vircuit circuit (VC) setup phase offers an opportunity for path selection, and hence HNTES identifies the ingress/egress routers corresponding to the source and destination addresses of α flows, and requests intradomain circuits between these routers.

The basic interface to the IDC requires an application to specify the circuit rate, duration, start time, and the endpoints in its advance-reservation request. The specified rate is used both for (i) path computation in the call-admission/circuit-scheduling phase and (ii) policing traffic in the data plane. If the application requests a high rate for the circuit, the request could be rejected by the OSCARS IDC due to a lack of resources. On the other hand, if the request is for a relatively low rate (such as 1 Gbps), then the policing mechanism could become a limiting factor to the throughput of α flows, preventing TCP from increasing its sending rate.

The purpose of this paper is to evaluate the effects of different scheduling and policing mechanisms to achieve two goals: (i) reduction in delay and jitter of real-time sensitive flows that share the same interfaces as α flows, and (ii) support for high-throughput α-flow transfers.

Our *key findings* are as follows: **(i)** With the current widely deployed best-effort IP-routed service, which uses first-come-first-serve (FCFS) scheduling on egress interfaces of routers,

 ISBN: 978-1-61208-263-9

the presence of an α flow can increase the delay and jitter experienced by audio/video flows. **(ii)** This influence can be eliminated by configuring two virtual queues at the contending interface and redirecting identified α flows to one queue (an α queue), while all other flows are directed to a second queue (a β queue). **(iii)** If α flows use the dynamic circuit service offered by providers such as ESnet and Internet2, the currently configured policing mechanism will direct in-profile packets to a higher priority queue, and out-of-profile packets to a lower priority queue, which in turn, may have adverse effects on throughput. The reason of this degraded α-flow throughput is that the separation of in-profile and out-of-profile packets to different queues can cause out-of-sequence arrivals at the TCP receiver, which triggers TCP's fast retransmit/fast recovery congestion algorithm. **(iv)** An alternative approach to dealing with out-of-profile packets is to probabilistically drop a few packets using Weighted Random Early Detection (WRED), and to buffer the remaining out-of-profile packets in the same queue as the in-profile packets. This prevents the out-of-sequence problem and results in a smaller drop in α-flow throughput when compared to the separate-queues approach. Nevertheless, even with this WRED approach α-flow throughput is reduced when compared to the no-policing, scheduling-only solution. The WRED approach has a fairness advantage when multiple α flows are directed to the same α queue. However, preliminary NetFlow analysis indicates that the likelihood of two simultaneous α flows sharing a single link is fairly low if the α-flow threshold is relatively high (and it needs to be high in order to have adverse effects on other flows requiring its isolation). In summary, it may not be worth sacrificing α-flow throughput with policing if multiple simultaneous α flows occur rarely.

Section II provides background and reviews related work. Section III describes the experiments we conducted on a high-speed testbed to evaluate different combinations of QoS mechanisms and parameter values to achieve our dual goals of reduced delay/jitter for real-time flows and high throughput for α flows. Our conclusions are presented in Section IV.

II. Background and related work

The first three topics, historical perspective, a hybrid network traffic engineering system, and QoS support in state-of-the-art routers, provide the reader with relevant background information. The last topic, QoS mechanisms applied to TCP flows, covers related work.

Historical perspective: In the nineties, when Asynchronous Mode Transfer (ATM) [6] and Integrated Services (IntServ) [7] technologies were developed, virtual circuit (VC) services were considered for delay-sensitive multimedia flows. However, these solutions were not scalable to large numbers of flows because of the challenges in implementing QoS mechanisms such as policing and scheduling on a per-flow basis. Instead, a solution of overprovisioning connectionless IP networks to prevent buildups in router buffers was sufficient to meet delay requirements of real-time audio/video flows. While this solution works well most of the time, there are occasional periods when a single large dataset transfer is able to ramp up to a very high rate and adversely affect other traffic [8]. Such transfers, which are referred to as α flows, occur when the amount of data being moved is large, and the end-to-end sustained rate is high.

In the last ten years, there has been an emergent interest in using VCs but for α flow transfers not multimedia flows. As noted in Section I, service providers are interested in routing these α flows to traffic-engineered, QoS-controlled paths. The scalability issue is less of a problem here since the number of α flows is much smaller than of that of real-time audio-video flows. Based on the threshold chosen for α flows, this number could be as small as 1. It is interesting to observe this "flip" in the type of applications being considered for virtual-circuit services, i.e., from real-time multimedia flows to file-transfer flows.

Hybrid Network Traffic Engineering System (HNTES): Ideally if end-user applications such as GridFTP [9] alerted the provider networks en route between the source and destination before starting a high-rate, large-sized dataset transfer, these networks could perform path-selection and direct the resulting TCP flow(s) to traffic-engineered, QoS-controlled paths. However, most end-user applications do not have this capability, and furthermore inter-domain signaling to establish such paths requires significant standardization efforts. Meanwhile, providers have recognized that intra-domain traffic-engineering is sufficient if α flows can be automatically identified at the ingress routers. Deployment of such a traffic-engineering system lies within the control of individual provider networks, making it a more attractive solution. Therefore, the first step in our work was to determine whether such automatic α flow identification is feasible or not.

In our prior work [2], we started with hypothesis that computers capable of sourcing/sinking data at high rates are typically allocated static IP addresses, which means that the source-destination IP address prefixes can be used to identify α flows. If a NetFlow report for a flow showed that more than H bytes (set to 1 GB) were sent within a fixed time interval (set to 1 min), we classified the flow as an α flow, and stored the source and destination address prefixes (/24 and /32). This NetFlow data analysis is envisioned to be carried out offline on say a nightly basis for all ingress routers. If no flows are observed from a particular source-destination address prefix within an aging-out time period (set to 30 days), then the entry is removed. The effectiveness of this scheme was evaluated through an analysis of 7 months of NetFlow data obtained from an ESnet router. For this data set, 94% (82%) of bytes generated by α flows in bursts would have been identified correctly had /24 (/32) based prefix IDs been used. The results are consistent with findings from NetFlow data collected over 7 months from three other ESnet routers.

Given the effectiveness of this offline α-flow identification scheme, HNTES can provision firewall filters based on source/destination IP address prefixes to automatically detect

 ISBN: 978-1-61208-263-9

packets from α flows at a provider's ingress routers and redirect them to traffic-engineered, QoS-controlled paths.

QoS support in state-of-the-art routers: Multiple policing, scheduling and traffic shaping mechanisms have been implemented in today's routers. We review the particular mechanisms used in ESnet, and hence in our experiments. For scheduling, two mechanisms are used: Weighted Fair Queueing (WFQ) and Priority Queueing (PQ) [10]. With WFQ, multiple traffic classes are defined, and corresponding virtual queues are created on egress interfaces. Bandwidth and buffer space can be strictly partitioned or shared among the virtual queues. WFQ can be combined with PQ as explained later. On the ingress-side, policing is used to ensure that a flow does not exceed its assigned rate (used by the IDC during call admission). For example, in a single-rate two-color (token bucket) scheme, the average rate (which is the rate specified to the IDC in the circuit request) is set to equal the generation rate of tokens, and a maximum burst-size is used to limit the number of tokens in the bucket. The policer marks packets as *in-profile* or *out-of-profile*. Three different actions can be configured: (i) discard out-of-profile packets immediately, (ii) classify out-of-profile packets as belonging to a `Scavenger Service (SS)` class, and direct these packets to an `SS` virtual queue, or (iii) drop out-of-profile according to a WRED profile, but store remaining out-of-profile packets in the same queue as in-profile packets. For example, the drop rate for out-of-profile packets can increase linearly from 0 to 100 for corresponding levels of queue occupancy.

QoS mechanisms applied to TCP flows: Many QoS provisioning algorithms that involve some form of active queue management (AQM) have been studied [11]–[15]. Some of the simpler algorithms have been implemented in today's routers, such as RED [11] and WRED [13], while other algorithms, such as Approximate Fair Dropping (AFD) [15], have been shown to provide better fairness. An analysis of the configuration scripts used in core and edge routers of ESnet and Internet2 shows that these AQM related algorithms are not enabled. This is likely due to the commonly adopted policy of overprovisioning (a 2008 Internet2 memorandum [16] states a policy of operating links at 20% occupancy). Nevertheless, providers have recognized that in spite of the headroom, an occasional α flow can spike to a significant fraction of link capacity (e.g., our GridFTP log analysis showed transfers occurring at over 4 Gbps across paths of 10 Gbps links [8]), and that such spikes can adversely affect other flows. This explains the providers' interest in controlling the path taken by these flows, i.e., directing them to traffic-engineered QoS-controlled paths.

III. EXPERIMENTS

A set of experiments are designed and executed to determine the best combination of QoS mechanisms with corresponding parameter settings in order to achieve our dual goals of reduced delay/jitter for real-time traffic and high throughput for α flows. For the first goal, we formulate a hypothesis as follows: a simple scheduling-only (no policing) scheme that isolates packets from α flows into a separate virtual queue on the egress interface from all other packets is sufficient to keep non-α flow delay/jitter low. *Experiment 1* tests this hypothesis.

In the current OSCARS IDC implementation, four classes-of-service (CoS) with corresponding virtual queues are used on the egress interfaces of routers: `network-control`, `best-effort`, `science-data`, and `scavenger-service`. The transmission rate and buffer allocation assigned to each of these queues is for example, 5%, 20%, 70%, and 5%, respectively. On the ingress side, policing is configured to check conformance of flows that requested circuits to their specified rates. In-profile packets are directed to the science-data queue, while out-of-profile packets are sent to the scavenger-service queue. We planned *experiment 2* to determine if this configuration of QoS mechanisms was suited to meeting our second goal of high-throughput for α flows. The expectation is that most circuit requests for file transfers will be for around 1 Gbps (on 10 Gbps links, this represents a significant fraction for just a single request), but as our prior work [8] showed scientific computing centers have hosts capable of sourcing/sinking data at over 4 Gbps. Policing such flows down to 1 Gbps will thus impact file-transfer throughput.

In *experiment 2*, out-of-profile packets resulting from ingress-side policing are directed to the scavenger-service (SS) queue, while in *experiment 3*, out-of-profile packets are subject to WRED as explained in Section II.

Section III-A describes the experimental setup, the experimental methodology, and certain router configurations that are common to all the experiments. Sections III-B, III-C, and III-D describe the three experiments, respectively.

A. Experimental Setup

The experimental network setup is shown in Fig. 1. The high-performance hosts, `W1` (West 1), `W2` (West 2), and `E1` (East 1), are Intel Xeon Nehalem E5530 models (2.4GHz CPU, 24GB memory) and run Linux version 2.6.33. The application hosts, `WA` (West App-host) and `EA` (East App-host), are Intel Dual 2.5GHz Xeon model and run Linux 2.6.18. The routers, `WR` (West Router) and `ER` (East Router), are Juniper MX80's running JunOS version 10.2. The link rates are 10 Gbps from the high-performance hosts to the routers, and 1 Gbps from the application hosts to the routers, and 10 Gbps between the routers.

This testbed is referred to as the Long Island MAN (LI-MAN), and is supported by ESnet as a DOE-funded testbed for networking research. The West-side hosts and routers are physically located in the Avenue-of-Americas (AoA) location in New York City, while the East-side hosts and routers are physically located in the Brookhaven National Laboratory (BNL) in Long Island, New York.

Each experiment consists of executing four steps: (i) plan the applications required to test a particular QoS mechanism, (ii) configure routers to execute the selected QoS mechanisms with corresponding parameter settings based on the planned

 ISBN: 978-1-61208-263-9

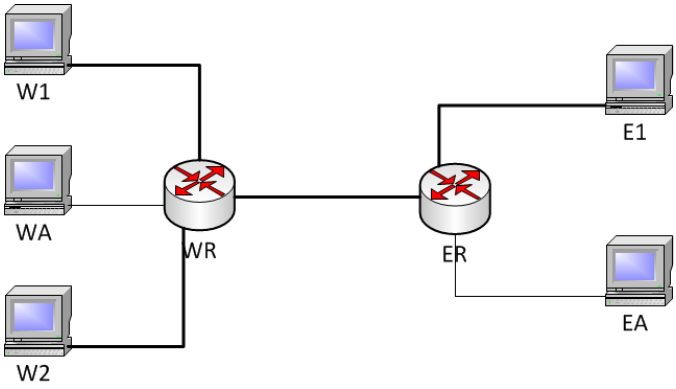


Figure 1. Experiment setup

application flows, (iii) execute applications on end hosts to create different types of flows through the routers, and (iv) obtain measurements for various characteristics, e.g., throughput, packet loss, and delay, from the end-host applications as well as from various counters in the routers.

A preliminary set of experiments were conducted to determine the specific manner in which the egress-side transmitter and buffer space were shared among multiple virtual queues. Theoretically both these resources can be strictly partitioned or shared in a work-conserving manner. If strictly partitioned, then even if there are no packets waiting in one virtual queue, the transmitter will not serve packets waiting in another queue. In this mode, each queue is served at the exact fractional rate assigned to it. In contrast, in the work-conserving mode the transmitter will serve additional packets from a virtual queue that is experiencing a higher arrival rate than its assigned rate if there are no packets to serve in the other virtual queues. Buffer space can similarly be shared in both modes. In all the experiments described below, the transmitter is shared among multiple virtual queues in work-conserving mode, while the buffer is shared in strictly partitioned mode.

Fig. 2 illustrates how a combination of QoS mechanisms was used in our experiments. *First*, incoming packets are classified into multiple classes based on pre-configured firewall filters, e.g., α-flow packets are identified by the source-destination IP address prefixes and classified into the α class. *Second*, packets in some of these classes are directly sent to corresponding egress-side virtual queues, while flows corresponding to other classes are subject to policing. A single-rate token bucket scheme is applied. If an arriving packet finds a token in the bucket, it is marked as being in-profile; otherwise it is marked as being out-of-profile. *Third*, for some policed flows, in-profile and out-of-profile packets are sent to separate egress-side virtual queues, while packets from other policed flows are subject to WRED before being buffered in a single virtual queue. On the egress-side, each virtual queue is assigned a priority level, a transmit rate (fraction of egress link capacity), and a buffer size. As noted in the previous paragraph the buffer allocation is strictly partitioned while the transmitter is shared in work-conserving mode. *Fourth*, the WFQ scheduler decides whether a virtual "queue is in-profile or not," by comparing the rate allocated to the queue and the rate at which packets have been served out of the queue. *Finally*, the PQ scheduler selects the queue from which to serve packets using their assigned priorities, but to avoid starvation of low-priority queues, as soon as a large enough number of packets are served from a high-priority queue to cause the status of the queue to transition to out-of-profile, the PQ scheduler switches to the next queue in the priority ordering. When all queues become out-of-profile, it starts serving packets again in priority order. It is interesting that while the *policer* is marking *packets* as in-profile or out-of-profile on a per-flow basis, the *WFQ scheduler* is marking *queues* as being in-profile or out-of-profile.

B. Experiment 1

1) Purpose and execution: The purpose of this experiment is to determine whether the simple scheduling-only solution of α-flow isolation to a separate virtual queue is sufficient to meet the first goal of keeping non-α flow delay/jitter low.

As per our execution methodology, the first step was to plan a set of applications. We decided to use three flows: a UDP flow, a high-speed TCP flow, and a "ping" flow. The application, `nuttcp`, is used to create both UDP and TCP flows. The UDP flow carries data from host `W2` toward host `W1`, while the TCP flow is from `E1` to `W1`. The TCP version used is H-TCP [17] because it is the best option to create high-speed (α) flows. The ping application sends repeated ICMP ECHO-REQUEST messages, one per second, from application host `EA` to high-performance host `W1`. Therefore, in this experiment, contention for buffer and transmitter resources occurs on the link from router `WR` to host `W1`. Although the high-performance host `W1` is the common receiver for all three flows, there is no CPU/memory resource contention at `W1` because the operating system automatically schedules the three receiving processes to three different cores.

The second step was to configure the routers. For comparison purposes, this experiment required two configurations: (i) `1-queue`: a single virtual queue is defined on the egress interface from `WR` to `W1`, and all three flows are directed to this queue, and (ii) `2-queues`: two virtual queues (α queue and β queue) are configured on the egress interface from `WR` to `W1`, and WFQ scheduling is enabled with the assigned transmitter rate (and buffer) percentages as follows: 60% for α queue and 40% for β queue. The priority of the α and β virtual queues was set to medium-high and medium-low, respectively. In the 2-queues configuration, two additional steps are required. A firewall filter is created in router `WR` to identify the TCP flow packets using its source and destination IP addresses (`E1` and `W1`, respectively). A class-of-service configuration command is used to classify these packets as belonging to the α class and to direct packets from this flow to the α queue on the egress interface from `WR` to `W1`. By default, all other packets are directed to the β queue, which means that packets from the UDP flow and ping flow will be directed to the β queue.

In the third step, the applications were executed as follows. Both the `nuttcp` UDP and ping flow were run for 200

 ISBN: 978-1-61208-263-9

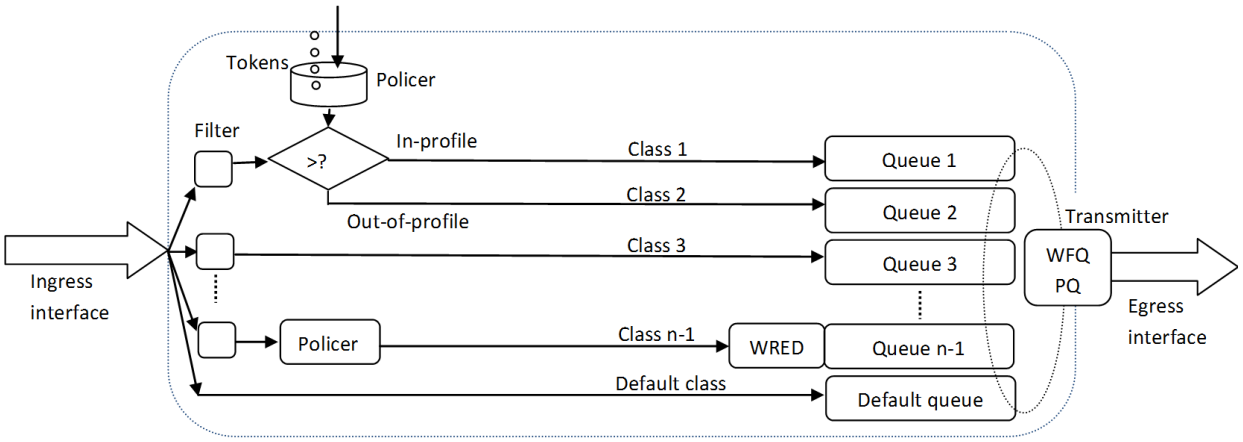


Figure 2. Illustration of QoS mechanisms in a router

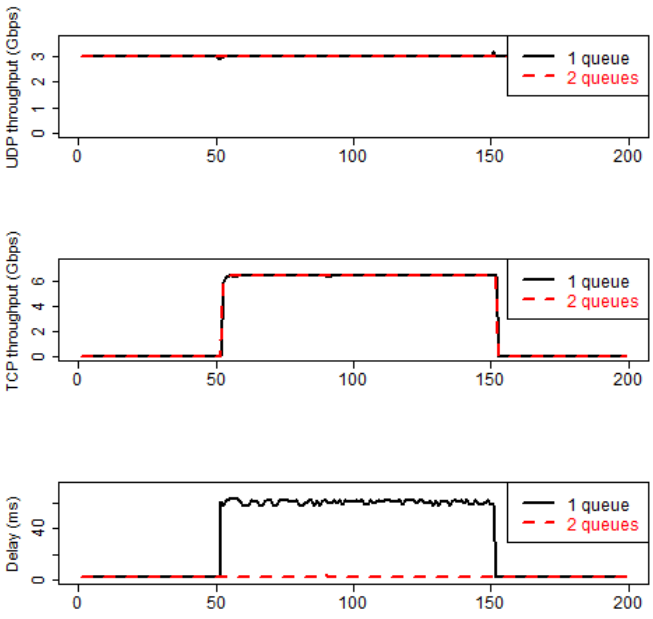


Figure 3. The x-axis is time measured in seconds; the top graph shows that the UDP rate is 3 Gbps in both the 1-queue and 2-queues configurations; the middle graph shows the TCP flow throughput; the bottom graph shows the delays experienced in the ping application.

seconds (from time 1 to time 200), while the `nuttcp` TCP flow was started at time 53 and run for 100 seconds. The rate of the UDP flow was set to 3 Gbps.

Finally, the UDP flow rate and TCP flow throughput reported in the next sub-section were obtained from measurements reported by the `nuttcp` application, and the ping delays were reported by the ping application.

2) Results and discussion: Fig. 3 illustrates that the simple scheduling-only solution of configuring two virtual queues on the shared egress interface and separating out the α flow packets into its own virtual queue leads to reduced packet delay/jitter for the β flows. In the 1-queue configuration, the mean ping delay across the 100 ping packets transmitted while the TCP flow was inactive was 2.28 ms and the standard deviation was 0.08 ms, while the mean and standard deviation of the 100 ping packets sent when the TCP flow was active were 60.6 ms and 1.64 ms, respectively. The ping delay increase is because the TCP (α) flow and the ping flow share the same single queue. In the 2-queues configuration, the mean and jitter of the ping delay were almost the same in the TCP-flow active and inactive periods. A small surge in ping delay to 4.5 ms occurred at time 91, which we ascertained was caused by network control packets exchanged between the routers.

Since the UDP flow rate at 3 Gbps was lower than the 40% assigned rate for the β queue, the latter was in-profile and hence the ping-application packets were served immediately, and not held up α-flow packets even though the α queue was sometimes out-of-profile. As explained in Section III-A, the PQ scheduler only honors priority if a queue is in-profile. It is interesting to note however that if the aggregate traffic directed to the β queue exceeds the β queue rate allocation when one or more α flows are present, then real-time flows could suffer from increased delay. Accurate estimation of the per-queue rate allocations is required.

C. Experiment 2

1) Purpose and execution: This experiment compares a 2-queues configuration (scheduling-only, no policing) with a 3-queues configuration (scheduling and policing), and furthermore compares multiple 3-queues configurations with different parameter settings.

As per our execution methodology, the first step was to plan applications. To study the behavior of the QoS mechanisms, the rate of the background traffic (an `nuttcp` UDP flow) was varied. Specifically, the same three application flows as in experiment 1 were planned, except that the rate of the `nuttcp` UDP flow was varied from 0 Gbps to 3 Gbps, and the `nuttcp` TCP flow was executed for the whole 200 sec.

In the second step, the router WR was configured with the following QoS mechanisms. The 2-queues configuration was the same as in experiment 1 (no policing), except that both queues were given equal weight in sharing the transmitter rate

 ISBN: 978-1-61208-263-9

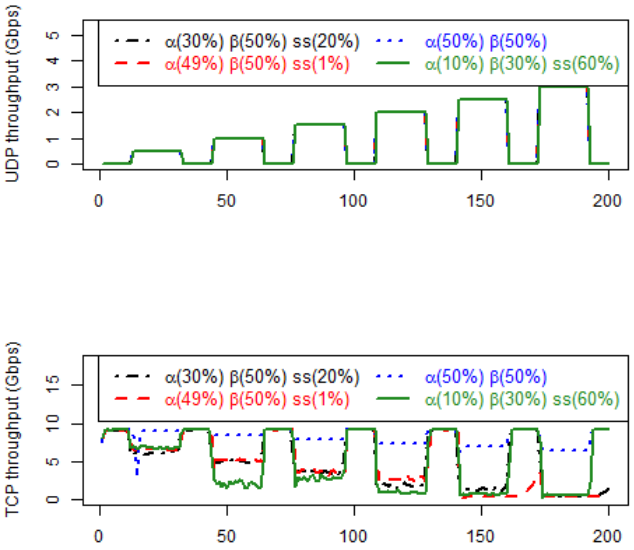


Figure 4. The x-axis is time measured in seconds; the top graph shows the on-off mode in which the UDP rate was varied; the lower graph shows the TCP flow throughput under the four configurations.

TABLE I. α-FLOW THROUGHPUT UNDER DIFFERENT BACKGROUND LOADS (UDP RATE) AND QOS CONFIGURATIONS

UDP rate (Gbps)	α-flow throughput (Gbps)			
	Percentages for 2-queues (α, β) and 3-queues (α, β, *SS*) configurations			
	(50,50)	(49,50,1)	(30,50,20)	(10,30,60)
0	9.12	9.09	9.07	9.12
0.5	8.92	6.62	6.06	6.83
1	8.43	5.22	5	2.12
1.5	7.94	3.78	3.67	2.82
2	7.44	2.7	1.93	0.92
2.5	6.95	0.33	1.38	0.69
3	6.46	0.34	0.38	0.61

and buffer space (50% each). For the 3-queues configurations, the percentages for the three queues (α, β, and *SS*) to which in-profile TCP-flow packets, UDP and ping packets, and out-of-profile TCP-flow packets, were directed, respectively, are shown in Table I. The priority of these three virtual queues is medium-high, medium-low, and low respectively. The policer is configured to direct in-profile TCP flow packets ($\leq$ 1 Gbps and burst-size $\leq$ 31 KB) to the α queue, and out-of-profile packets to the *SS* queue.

In the third step, experiment execution, the UDP flow rate was varied in a particular on-off pattern as seen in the top graph of Fig. 4. Finally, the performance metrics were collected as described for experiment 1.

2) Results and discussion: Fig. 4 shows the TCP throughput under the four configurations (one 2-queues and three 3-queues) for different rates of the background UDP flow. When the UDP flow rate is non-zero, since some of the plots overlap, we have summarized the mean TCP-flow throughput in Table I. When there is no background UDP traffic, the throughput of the TCP flow is around 9.1 Gbps for all four configurations as seen in the first row of Table I. As the background traffic load increases, the throughput of the TCP flow in all the 3-queues configurations drops more rapidly than in the 2-queues configuration, e.g., when the background UDP-flow rate is 3 Gbps, the TCP throughput is around 300-610 Mbps for the 3-queues configurations, while the TCP throughput is 6.5 Gbps for the 2-queues scenario (see last row of Table I).

In addition to explaining the first and last rows of Table I, we provide an explanation for the drop in TCP-flow throughput in the last column of the row corresponding to UDP rate of 1 Gbps, which highlights the importance of choosing the WFQ allocations carefully.

Explanation for the first row of Table I: The explanation for the TCP-flow throughput when there is no background traffic is straightforward in the 2-queues configuration. As there are no packets to be served from the β queue and the transmitter is operating in a working-conserving manner, the β queue's 50% allocation is used instead to serve the α queue, and correspondingly the TCP flow enjoys the full link capacity.

The explanation for the TCP-flow throughput values observed in the 3-queues configurations requires an understanding of the packet pattern incoming to the policer (see Fig. 2) and the rate at which packets leave the policer. When TCP-flow throughput is almost the line rate (over 9 Gbps), then the rate at which in-profile packets leave the policer will be almost constant at 1 Gbps. This is because the token generation rate is 1 Gbps and packet inter-arrival times are too short for a significant collection of tokens in the bucket. Therefore, it appears that in an almost periodic manner, every tenth packet of the TCP flow is marked as being in-profile and sent to the α queue and the remaining 9 packets are classified as out-of-profile and sent to the *SS* queue. Given that in all three 3-queues configurations, the WFQ scheduler will consider the α queue as being in profile (since even with the smallest allocation, this queue is assigned 10%), the PQ scheduler will systematically serve 1 packet from the α queue followed by 9 packets from the *SS* queue thus preserving the sequence of the TCP-flow packets. In the (49,50,1) configuration, 9 packets will be served out of the SS queue in sequence even though the queue would be regarded as out-of-profile after the first packet is served. This is because there are no packets in the β queue and none in the α queue given the policer's almost-periodic direction of 1-in-10 packets to this queue. Since no packets are out-of-sequence or lost, the TCP-flow throughput remains high at above 9 Gbps in all three 3-queues configurations.

Explanation for the last row of Table I: When there is background `nuttcp` UDP traffic at 3 Gbps, in the 2-queues configuration, it is easy to understand that the `nuttcp` TCP

 ISBN: 978-1-61208-263-9

flow is able to use up most of the remaining bandwidth, which is the line rate minus the rate of background `nuttcp` UDP flow, and hence the TCP-flow throughput is about 6.5 Gbps.

The explanation for the low `nuttcp` TCP throughput in the 3-queues configurations is that the opposite of the systematic behavior explained above for the first row occurs here. When the incoming packet rate to the policer is lower than the line rate, the token bucket has an opportunity to collect a few tokens. Therefore, when TCP-flow packets arrive at the policer, a burst of them will be classifed as in-profile (since for every token present in the bucket, one packet is regarded as being in-profile), and sent to the α queue. These will be served in sequence, but because the transmitter has to serve the β queue (for the UDP flow), the pattern in which the policer sends packets to the α queue and *SS* queue is more unpredictable and involves bursts. This results in TCP segments arriving out-of-sequence at the receiver (as confirmed with `tcpdump` and `tcptrace` analyses presented in the next section). Out-of-sequence arrivals triggers TCP's Fast retransmit/Fast recovery algorithm, which causes the sender's congestion window to halve resulting in lower throughput.

Explanation for the last-column entry in the row corresponding to 1 Gbps in Table I: The TCP-flow throughput drops much faster from 6+ Gbps to 2.12 Gbps when UDP rate increases from 0.5 to 1 Gbps in the (10,30,60) 3-queues configuration than in the other two 3-queues configurations. This is explained using the above-stated reasoning that when the TCP-flow packets do not arrive at close to the line rate, the inter-packet arrival gaps allow the token bucket to collect a few tokens, making the policer send bursts of packets to the α queue. In this (10,30,60) configuration, after serving only one packet from each burst, the WFQ scheduler will declare the α queue to be out-of-profile since its allocation is only 10% or equivalently 1 Gbps. This will lead to a greater number of out-of-sequence arrivals at the TCP receiver than in the other two 3-queues configurations, and hence lower throughput.

In summary, the higher the background traffic load, the lower the `nuttcp` TCP-flow packet arrival rate to the policer, the larger the inter-arrival gaps, the higher the number of collected tokens in the bucket, and the larger the number of in-profile packets directed to the α queue. If the WFQ allocation to the α queue is insufficient to serve these in-profile bursts, packets from the α queue and *SS* queue will be intermingled resulting in out-of-sequence packets at the receiver. This fine point notwithstanding, the option of directing out-of-profile packets from the policer to a separate queue appears to be detrimental to α-flow throughput. We conclude that the second goal of high α-flow throughput cannot be met with this policing approach. In the next experiment, a different mechanism of dealing with out-of-profile packets is tested.

D. Experiment 3

1) Purpose and execution: This experiment compares the approach of applying WRED to out-of-profile packets rather than redirecting these packets to a scavenger-service queue as in experiment 2. As per our execution methodology, the planned applications are the same as in experiment 1. The UDP-flow rate is maintained at 3 Gbps throughout the 200 sec time interval.

TABLE II. QOS CONFIGURATIONS FOR EXPERIMENT 3

Configuration	Policing	WFQ allocation `2-queues`:(α,β) `3-queues`:(α,β,`SS`)	WRED
`2-queues`	None	(60,40)	NA
`3-queues + policing1`	OOP to `SS` queue	(59,40,1)	NA
`3-queues + policing2`	OOP to `SS` queue	(20,40,40)	NA
`2-queues + policing + WRED`	WRED	(60,40)	Drop prob. = queue occ.

The next step is router configuration. Four configurations are compared as shown in Table II. OOP stands for Out-of-Profile packets. In the fourth option, OOP packets are dropped probabilistically at the same rate as the fraction of α-queue occupancy. In other words, if the α queue has 50% occupancy, then 50% of the OOP packets are dropped on average.

The applications were executed to generate one `nuttcp` TCP flow and one `nuttcp` UDP flow. Finally, in addition to the previously used methods of obtaining throughput reports from `nuttcp`, two packet analysis tools, `tcpdump` and `tcptrace`, were used to determine the number of out-of-sequence packets at the receiver. Additionally, to find the number of lost packets, a counter was read at router `WR` for the `WR`-to-`W1` link before and after each application run.

2) Results and discussion: The lower graph in Fig. 5 and Table III show that the TCP-flow throughput is highest in the 2-queues (no policing) scenario, with the WRED option close behind. The policing with WRED option performs much better than the options in which out-of-profile (OOP) packets are directed to an *SS* queue. In the WRED-enabled configuration, the TCP flow experiences a small rate of random packet loss, as shown in Table III, while in redirect-OOP-packets-to-*SS*-queue configurations, there are much higher numbers of out-of-sequence packets. The out-of-sequence packets in the WRED-enabled configuration result from the 15 lost packets, and are not independent events.

Surprisingly, even though the number of out-of-sequence packets is larger for the `3-queues + policing1` configuration, the throughput is higher in that configuration. This implies that a fewer number of the out-of-sequence packets caused triple-duplicate ACKs in the first case. But this pattern is likely to change for repeated executions of the experiment.

Finally, Fig. 5 shows that in the `2-queues` (no policing) configuration, there is degradation of throughput soon after the flow starts. Also, Table III shows a loss of 5050 packets (the 4076 out-of-sequence packets were related to these losses) from tcptrace, we found that these losses occur at the start of the transfer. This is explained by the aggressive growth

 ISBN: 978-1-61208-263-9

TABLE III. NUMBER OF OUT-OF-SEQUENCE PACKETS AND LOST PACKETS FOR DIFFERENT QOS SETTINGS

Measure	2queues	3queues+ policing1	3queues+ policing2	2queues+ policing+wred
Average throughput	6 Gbps	0.92 Gbps	0.47 Gbps	5.6 Gbps
Num. of out-of-sequence packets at the receiver	4076	8812	7199	15
Num. of lost packets at the WR-to-W1 router link	5050	0	0	15

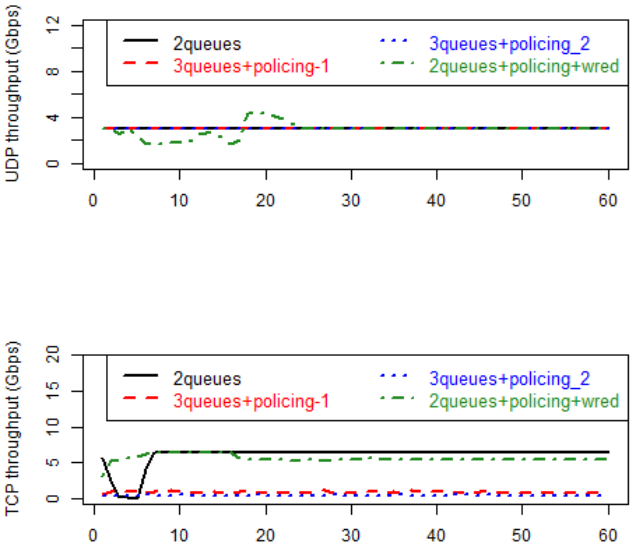


Figure 5. The x-axis is time measured in seconds; the top graph shows the on-off mode in which the UDP rate was varied; the lower graph shows the TCP flow throughput under the four configurations.

of the congestion window (cwnd) in H-TCP, which uses a short throughput probing phase at the start. During the 1st second, the throughput of the nuttcp TCP flow averaged 5.7 Gbps. The 5050 lost packets occurred in the 2^{nd} second. These losses occurred in the WR router buffer on its egress link from WR to W1. If H-TCP increased its cwnd to a large enough value to send packets at an instantaneous rate higher than 7 Gbps, then given the presence of the UDP flow at 3 Gbps, the α queue would fill up. Through experimentation, we determined that the particular router used as WR has a 125 MB buffer. Since the buffer is shared between the α and β queues in a strictly partitioned mode with the 60-40 allocation, the α queue has 75 MB, which means that if the H-TCP sender exceeds the 7 Gbps rate by even 600 Mbps, the α queue will fill up within a second. Inspite of this initial packet loss, the 2-queues no-policing configuration achieves the highest throughput. The 2-queues+policing+WRED configuration will likely be more fair if multiple α flows are directed to the same α queue. For example, the AFD approach [15] offers a dropping mechanism to achieve fairness between TCP flows. In the 2-queues no-policing configuration, α flows will experience the same fairness level as in today's best-effort network, achieve high-throughput while simultaneously not impacting the delay/jitter of real-time flows. A preliminary analysis of ESnet NetFlow data shows that when the defining threshold for α flows is relatively high, it is only on rare occasions that multiple α flows from different transfers share the same link (some transfers use multiple parallel TCP flows as observed in our GridFTP log analysis [8]).

IV. CONCLUSIONS

This paper presented an approach to QoS provisioning for α flows (high-rate, large-sized file transfers) for two purposes: (i) to reduce the adverse effects they can cause on delay-sensitive flows, and (ii) to maximize the throughput of α flows. Several experiments were conducted to compare different QoS mechanisms on state-of-the-art routers. We showed that a simple 2-queue scheme in which α flows are isolated to their own queue is sufficient to achieve the first goal. As for the second goal, we investigated the effects of two policing schemes. A scheme that is commonly deployed in research-and-education networks (REN) separates out in-profile and out-of-profile packets from an α flow into two different virtual queues. The policed rate is determined by the rate requested during circuit setup (REN providers offer a dynamic circuit service that is used by α flows). However, it is difficult to accurately gauge the rate at which a file transfer can be executed, and sometimes α flows exceed their requested rates. When this happens, the solution of using two queues causes a significant number of out-of-sequence packets at the receiver, and TCP's fast retransmit/fast recovery method reduces throughput. An alternative approach is to use Weighted Random Early Detection (WRED) and drop out-of-profile packets probabilistically, but keep the remaining out-of-profile and in-profile packets in the same queue. This mechanism results in higher throughput than the deployed approach, but it nevertheless reduces α-flow throughput. Therefore, to meet our dual goals, we recommend a scheduling-only, no-policing approach.

 ISBN: 978-1-61208-263-9

V. ACKNOWLEDGMENT

The University of Virginia portion of this work was supported by the U.S. Department of Energy (DOE) grant DE-SC0002350, DE-SC0007341 and NSF grants OCI-1038058, OCI-1127340, and CNS-1116081. The ESnet portion of this work was supported by the Director, Office of Science, Office of Basic Energy Sciences, of the U.S. DOE under Contract No. DE-AC02- 05CH11231. This research used resources of the ESnet ANI Testbed, which is supported by the Office of Science of the U.S. DOE under contract DE-AC02-05CH11231, funded through the American Recovery and Reinvestment Act of 2009.

REFERENCES

[1] S. Sarvotham, R. Riedi, and R. Baraniuk, "Connection-level analysis and modeling of nework traffic," ACM SIGCOMM Internet Measurement Workshop 2001, Nov. 2001, pp. 99-104.

[2] Z. Yan, C. Tracy, and M. Veeraraghavan, "A hybrid network traffic engineering system," Proc. of IEEE 13th High Performance Switching and Routing (HPSR) 2012, Jun. 24-27, 2012, pp. 141-146.

[3] Esnet. Retrieved: 02.11.2013. [Online]. Available: http://www.es.net/

[4] On-Demand Secure Circuits and Advance Reservation System (OSCARS). Retrieved: 02.11.2013. [Online]. Available: http://www.es.net/services/virtual-circuits-oscars

[5] (2010, Feb.) Inter-domain Controller (IDC) Protocol Specification. Retrieved: 02.11.2013. [Online]. Available: http://www.controlplane.net/

[6] J. Spragins, "Asynchronous Transfer Mode: Solution for Broadband ISDN, Third Edition [New Books]," Jan./Feb. 1996, pp. 7.

[7] E. R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP)," RFC 2205, Sep. 1997.

[8] Z. Liu, M. Veeraraghavan, Z. Yan, C. Tracy, J. Tie, I. Foster, J. Dennis, J. Hick, Y. Li, and W. Yang, "On using virtual circuits for GridFTP transfers," The International Conference for High Performance Computing, Networking, Storage and Analysis 2012 (SC 2012), Nov. 10-16, 2012.

[9] GridFTP. Retrieved: 02.11.2013. [Online]. Available: http://globus.org/toolkit/docs/3.2/gridftp/

[10] J. Kurose and K. Ross, "Computer networks: A top down approach featuring the internet," Pearson Addison Wesley, 2010.

[11] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," IEEE/ACM Transactions on Networking, Aug. 1993, pp. 397-413.

[12] D. Lin and R. Morris, "Dynamics of random early detection," ACM SIGCOMM Computer Communication Review, 1997, pp. 127-137.

[13] WRED. Retrieved: 02.11.2013. [Online]. Available: http://www.cisco.com/en/US/docs/ios/11_2/feature/guide/wred_gs.html

[14] R. Guérin, S. Kamat, V. Peris, and R. Rajan, "Scalable QoS provision through buffer management," vol. 28, no. 4, 1998, pp. 29-40.

[15] R. Pan, L. Breslau, B. Prabhakar, and S. Shenker, "Approximate fairness through differential dropping," ACM SIGCOMM Computer Communication Review, pp. 23–39, 2003.

[16] R. P. Vietzke. (2008, Aug.) Internet2 headroom practice. Retrieved: 02.11.2013. [Online]. Available: https://wiki.internet2.edu/confluence/download/attachments/17383/Internet2+Headroom+Practice+8-14-08.pdf

[17] D. Leith and R. Shorten, "H-TCP: TCP for high-speed and long-distance networks," Proc. of PFLDnet, Feb. 16-17, 2004.

 ISBN: 978-1-61208-263-9

Novel Rate-Jitter Control Algorithms

Modeling and Analysis

Madhu Babu Sikha, Manivasakan Rathinam
Department of Electrical Engineering,
Indian Institute of Technology Madras,
Chennai-600036, India.
email: ee10d028, rmani@ee.iitm.ac.in

***Abstract*—Time Division Multiplexing over Internet Protocol (TDMoIP) is a transport technology that extends the voice, video and data traffic transparently over IP. When TDM traffic is packetized and injected into a packet switched network (PSN) for transportation, packets arrive at the destination with different inter-arrival times, due to variable delay (jitter) introduced by PSN. This network induced jitter should be minimized, because TDM devices operate at constant bit rate. Problem of jitter from theoretical (competitive and statistical analysis) to more practical view point had been well studied. In this paper, we have proposed two on-line algorithms, algorithm-A and algorithm-B to minimize rate-jitter. We have shown both analytically and by simulation that the rate-jitter achieved by algorithm-A is strictly less than the rate-jitter of an on-line algorithm proposed in a previous work. The simulation results shows that algorithm-B also achieves less rate-jitter than the reference algorithm. We also undertake the statistical analysis of above algorithms, in particular, we have modeled jitter buffer as an $\mathrm{MMPP}/\widetilde{\mathrm{D}}/1/\mathrm{B_{on}}$ queue by making two assumptions on Markov modulated Poisson process (MMPP) and steady state queue length distribution is calculated. The correctness of the analytical results corresponding to our proposed algorithms is verified with simulation results. From the simulation results, it is also shown that the mean waiting time of a packet in the buffer is less for both proposed algorithms compared with the reference algorithm.**

Keywords-TDMoIP; MMPP; rate-jitter; PSN; state dependent service

I. INTRODUCTION

TDM [19] has been the most promising technology over the decades to transmit voice. In the recent years, it is being used to transport video and data also. In TDM, there is a dedicated channel for each user. This channel is used only when the user is making a call or when some data is transmitting. Therefore, the bandwidth is not used effectively in TDM, since the channel is idle most of the time and TDM services are also expensive. On the other hand, PSN [20, 21] uses bandwidth efficiently and is cheap. So, emulating the TDM traffic over a PSN is an effective solution.

TDMoIP is a mechanism to connect two TDM islands through IP network. On the transmitter side, fixed number of TDM frames are packetized and sent across an IP network. So, all the packets with TDM payload are of equal size. At the receiver, the TDM payload is retrieved along with timing and the TDM stream is regenerated before sending downstream. If the TDM payload is sent over connectionless service in IP, reordering is done. When these IP packets with TDM payload traverse through the network, each packet may be routed in different path, so, they encounter different nodes and variable queueing delays. This queueing delay is the dominant part in end-to-end delay of a packet. Finally, they arrive at the receiver with variable inter-arrival times (IATs) as compared with almost constant IAT at the transmitter. This variation in the arrivals (packet delay variation) is called as *jitter*. At the receiver, this jitter causes serious problems for audio playback. To overcome jitter, all the received packets are stored in a buffer called as *jitter-buffer* and associated an algorithm which decides the dequeueing instant of next packet to be transmitted (or service initiation time of each packet). The above algorithm minimizes the output jitter, given the arrival times and/or the number of packets in the queue. This process is called as jitter regulation. The output of jitter regulator is fed to a link scheduler to send the packets on to an outgoing link. Jitter control is the sequence of the two operations: jitter regulation and link scheduling. This work is related to jitter regulation.

There are two main ways to quantify jitter [1]: one measure, called delay-jitter is the maximum difference between arrival times of different packets and the ideal time difference in a perfectly periodic sequence (where packets are spaced exactly X_a time units apart, X_a is the IAT of the packet arrival sequence). The second measure is rate-jitter; it bounds the difference in packet delivery rates at various times. More precisely, it measures the difference between the maximal and minimal IATs. Rate-jitter is a useful measure for many real-time applications such as voice and video broadcast over the Internet.

The rest of the paper is organized as follows: In Section II, we discuss the literature related to jitter control techniques in PSNs and queueing models with state dependent service. In Section III, we briefly discuss about MMPP, the 4-state MMPP model used in this work and two main assumptions about MMPP which makes the analysis of queueing model easier. Section IV discusses the proposed rate-jitter control algorithms and the analytical bound of rate-jitter for algorithm-A. In Section V, we discuss the $MMPP/\widetilde{D}/1/B_{on}$ queue modeling and we give analytical expression for the steady state queue length distribution. In Section VI, we give simulation results and finally, we conclude by summarizing the results and discussing future work in Section VII.

 ISBN: 978-1-61208-263-9

II. Related Work

Mansour *et al*. [1] used *competitive analysis* in order to compare an on-line algorithm with off-line algorithm. An *off-line algorithm* schedules a packet by using future arrivals also. Though off-line algorithm is impractical, it does deliver departure/dequeueing sequence with minimum possible jitter and forms the lower bound. Hence, off-line algorithm is used to compare the performance of any on-line algorithm proposed (for the same packet arrival sequence). An *on-line algorithm* schedules a packet based on the packet arrival times on or before the service initiation instant of the packet in question. Mansour *et al*. proposed an on-line algorithm for rate-jitter control (we call it as Mansour algorithm in rest of the paper), which achieves a rate-jitter bounded by the rate-jitter of an off-line algorithm. Mansour algorithm requires a buffer size of $B_{on} = 2B + h$, where B is the buffer size of the off-line algorithm and h is a space parameter. Mansour algorithm tightly schedules the packets within the given bounds I_{max} and I_{min} and achieves a rate-jitter not more than $I_{max} - I_{min}$, where I_{max} and I_{min} are the maximum and minimum bounds on the inter-departure times (IDTs) of the off-line algorithm, respectively. ElBatt *et al*. [2] proposed a traffic recovery mechanism to control the jitter. A detailed survey of rate-control algorithms can be found in [3]. Hay *et al.* [4] extended [1] to multiple streams and derived tight lower bounds for jitter regulation, both in off-line and on-line cases. An analysis of delay jitter control algorithms can be found in [5].

A new queueing model $G/\widetilde{D}/1/K$ is proposed in [6] to analyze the performance of the proposed adaptive timing method with state dependent service rates. A packet voice multiplexer is modeled as an $M/\widetilde{D}/1/K$ queue in [7], where the least significant bits of a voice packet are dropped during congestion period of multiplexer to reduce the queueing delay. The service time of a packet is determined depending on the buffer occupancy.

The main characteristic of Internet traffic is that its parameters (packet IAT, data transmission rate, etc.) are Long Range Dependent (LRD), i.e., a non-zero correlation exists in long-term time-scales. MMPP is a widely used arrival process in communication networks for modeling traffic whose arrival rate varies with respect to time. It can capture the correlation between IATs in the arrival process. Andersen *et al.* [10] illustrated that the superposition of four two-state MMPP's are sufficient to model the second-order self-similar behavior over large time-scales. The authors also proposed a fitting procedure for matching second-order properties of counts to that of a second-order self-similar process. Muscariello *et al.* [16] proposed a new MMPP traffic model that accurately models the LRD Internet traffic over time-scales. Yoshihara *et al.* [17] also proposed a fitting method for self-similar traffic based on the superposition of two-state MMPP's that matches the variance function over several time-scales. Nogueira *et al.* [18] extended [17] to accurately produce the self-similar traffic. The authors proposed a new fitting procedure that matches the complete distribution (besides variance) at each time scale.

Fischer *et al.* [10] did a detailed survey on MMPP and presented all the results about MMPP and queues with MMPP as input. A 4-state MMPP model is developed in [11] to characterize the behavior of aggregate input traffic in an ATM multiplexer. The performance of an ATM multiplexer with MMPP (using the model in [11]) as input is studied in [12] by making two assumptions on MMPP. The authors modeled the buffer as an $MMPP/D/1/K$ queue and calculated the queue length distribution, mean waiting time and cell loss probabilities. Choi *et al.* [13] analyzed a queueing system $MMPP/G_1, G_2/1/B$ with queue length dependent service times and then applied the results to cell discarding scheme in ATM networks. The authors have defined two service times depending on whether the buffer level is above or below a threshold.

The performance metrics studied in this paper are rate-jitter and mean waiting time of a packet in the queue (jitter-buffer). Our contribution is two-fold: (a) proposed algorithm-A and algorithm-B (both uses the same buffer size as Mansour algorithm) for rate-jitter control, shown that both of them achieves less rate-jitter compared with Mansour algorithm (Mansour work [1] has a lot of practical implication, this is the main reason for comparing our proposed algorithms with Mansour algorithm) and (b) statistical modeling of jitter-buffer as $MMPP/\widetilde{D}/1/B_{on}$ queue for the performance analysis of proposed algorithms. In the literature, queues with MMPP input and state dependent service are not studied thoroughly. To the best of our knowledge, this queueing model is not considered so far. We have defined a service time corresponding to each level of buffer occupancy and state of the MMPP, so, all the service times are distinct. Steady state queue length distribution at departure epochs is calculated from the queueing model. Simulation results are in congruent with the derived analytical results. Even though the algorithms are proposed for TDMoIP application, we believe that they will work for any application in PSN.

III. 4-State MMPP Model

The MMPP is a doubly stochastic Poisson process whose arrival rate varies according to an M-state irreducible continuous time Markov chain (CTMC) [10]. MMPP can be viewed as a super-position of 'M' independent Poisson processes, where switching among the processes is governed by an M-state CTMC, i.e. when the MMPP is in state i, arrival process is Poisson with rate λ_i.

The analysis presented here is based on two assumptions, like in [12]: (i) MMPP changes its state at departure epochs and (ii) probability of two or more state changes between two successive departures is essentially zero (here state change means switching from one Poisson process to other). If the MMPP is in state j after a departure, then the arrival rate is λ_j until the next departure (i.e. until the next state change).

The states of a 4-state MMPP are labeled from 0 to 3, as shown in Fig. 1, taken from [11]. State transitions occur only between adjacent states with the rates specified in Fig. 1. The duration of state j is exponentially distributed with parameter

 ISBN: 978-1-61208-263-9

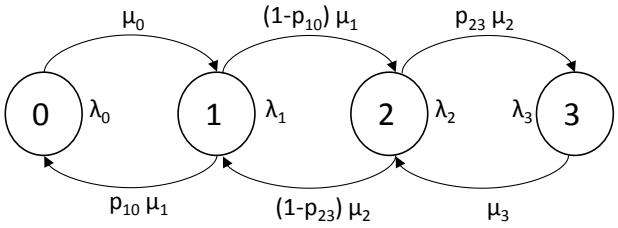


Figure 1: State transition diagram of a 4-state MMPP

μ_j and the arrivals in state j follow Poisson process with parameter λ_j.

Let the MMPP has M states, denoted by $j\,(0 \leq j \leq M-1)$, and let λ_j and $1/\mu_j$ be the arrival rate and mean state duration of state j respectively and $S_{j,l}$ be the service time of a packet when the buffer level is l and the MMPP is in state j. To satisfy the two assumptions, we need the mean time between arrivals and time between two successive departures (i.e. service time) should be much smaller than the mean state duration of the MMPP (this condition will be useful when the buffer is *non-empty*). Since the service times are multiple and distinct and number of states are M (so, mean state durations are also multiple), for the service time to be smaller than mean state duration, maximum of the service times should be smaller than the minimum of mean state durations, i.e. $max\{S_{j,l}\}$ should be smaller than $min(1/\mu_j) = 1/\mu_{max}$, where $j \in \{0, 1, ..M-1\}$ and $l \in \{1, 2, ..B_{on}-1\}$. Similarly, $1/\lambda_j$ (mean time between arrivals), $\forall j$ should be smaller than $1/\mu_i$, $\forall i$ where $i,\ j \in \{0, 1, ..M-1\}$, i.e. $1/\lambda_{min}$ should be much smaller than $1/\mu_{max}$ (this condition will be useful when the buffer is *empty*).

IV. Rate Jitter Control

For a given times sequence, the rate-jitter can be calculated from Equation (1) as follows:

$$Rate - jitter = \max_{0 \leq i,j \leq n} \{|(t_{i+1} - t_i) - (t_{j+1} - t_j)|\} \quad (1)$$

where t_j is the arrival/departure instant of j^{th} packet. When packets arrive at the destination, they are stored in the jitter-buffer and released some time later as discussed earlier. The release time (departure time) of a packet is determined by the rate-jitter control algorithm. Both the proposed algorithms starts with a buffer loading stage, the first packet is released only after the arrival of the $(B+1)^{th}$ packet and from there onwards packets are scheduled according to the algorithms.

A. Assumptions

- Packets are of equal size, which is true in TDMoIP as mentioned previously.
- Buffer can hold integral number of packets.
- Packets arrive at destination in the order in which they are injected into PSN.
- Packets are processed in the FIFO discipline.

B. Parameters, Notations and Definition

B	buffer size of an off-line algorithm
$1 \leq h < B$	space parameter for the on-line algorithm, such that $B_{on} = 2B + h$
I_{max}, I_{min}	maximum and minimum bounds on the IDT of an off-line algorithm
X_a	average IAT in the input (and also the output) sequence and $I_{min} << X_a << I_{max}$
L_k	buffer level at the k^{th} packet service initiation instant
$\widetilde{D}$	set of $B_{on}-1$ state (queue length) dependent service times
d_k	inter-departure time between k^{th} packet and $(k-1)^{th}$ packet
$S(k)$	service time of k^{th} packet

Define,

$$\delta_k \stackrel{\triangle}{=} \left(\frac{B_{on} + 1 - L_k}{2B}\right) X_a \quad (2)$$

C. Algorithm-A

This algorithm requires a buffer size of B_{on}, for each value of buffer occupancy L_k, it computes an IDT d_k, as given in the definition.

Algorithm 1 Algorithm-A

1) Calculate δ_k using Equation (2)
2) **if** $0 \leq L_k \leq h$ **then**
   $d_k \leftarrow I_{max}$
3) **else if** $\delta_k > I_{min} + \frac{X_a}{B}$ and $L_k > h$ **then**
   $d_k \leftarrow \delta_k$
4) **else if** $\delta_k < I_{min} + \frac{X_a}{B}$ and $L_k > h$ **then**
   $d_k \leftarrow \delta_k + I_{min}$
5) **end if**

X_a, I_{max} and I_{min} are requirements for this algorithm. Assuming that X_a is known in advance (like in ATM standard) is reasonable for real-time connections. Similarly I_{max} and I_{min} are the worst-case rate-jitter bounds, can be set based on the jitter bounds of the TDM traffic.

The following theorem calculates the rate-jitter bound of algorithm-A and proves that it is strictly less than that of [1].

Theorem 1: The rate-jitter of algorithm-A is bounded by $I_{max} - I_{min} - \frac{X_a}{B}$, which is strictly less than the rate-jitter bound $I_{max} - I_{min}$ of Mansour algorithm.

Proof: From the definition of δ_k, we can observe that it is a discrete valued function, whose value decreases linearly (with slope $-\frac{X_a}{2B}$) with an increase in buffer level. The maximum and minimum values of δ_k are $\frac{B_{on}}{2B}X_a$ and $\frac{1}{2B}X_a$, they occur at $L_k = 1$ and $L_k = B_{on} - 1$ respectively.

We can observe from algorithm-A that the IDT in the output is I_{max} when the number of packets L_k are less than or equal to h at the service initiation epoch of k^{th} packet.

For any $L_k > h$, the IDT is less than I_{max}, this can be seen observed by substituting $L_k = h + 1$.
So, the maximum IDT in the output sequence is I_{max}.

 ISBN: 978-1-61208-263-9

Now consider $L_k = B_{on} - 1$, then

$\delta_k = \frac{2B+h+1-(2B+h-1)}{2B} X_a = \frac{X_a}{B} < I_{min} + \frac{X_a}{B}$

So, the IDT is $I_{min} + \frac{X_a}{B}$ from the definition.

For any $L_k < B_{on} - 1$, IDT is greater than $I_{min} + \frac{X_a}{B}$, so, this is the minimum IDT in the output sequence.

As mentioned previously, rate-jitter is calculated as the difference between maximum and minimum IDTs, as given in Equation (1). Therefore, the bound on the rate-jitter is, $I_{max} - (I_{min} + \frac{X_a}{B}) = I_{max} - I_{min} - \frac{X_a}{B}$, which is less than the rate-jitter of Mansour algorithm. ■

D. Algorithm-B

This algorithm also requires a buffer size of B_{on}. The service time of a packet k at the head of the buffer is a function of the number of packets L_k at its service initiation instant. The service time $S(k)$ of k^{th} packet is calculated as given in the definition.

Algorithm 2 Algorithm-B

1) Calculate δ_k using Equation (2)
2) **if** $\delta_k > I_{min} + \frac{X_a}{B}$ and $L_k \leq B - h$ **then**
   $S(k) \leftarrow \delta_k + \frac{hX_a}{B}$
3) **else if** $\delta_k > I_{min} + \frac{X_a}{B}$ and $L_k > B - h$ **then**
   $S(k) \leftarrow \delta_k$
4) **else if** $\delta_k < I_{min} + \frac{X_a}{B}$ and $L_k > B - h$ **then**
   $S(k) \leftarrow \delta_k + I_{min}$
5) **end if**

The service times of algorithm-B (or IDTs of algorithm-A) are state (queue length) dependent. If the number of packets (L_k) in the queue increases, δ_k increases, so, the the service time (or IDT) decreases i.e. service rate increases. Service time (or IDT) is maximum when there are fewer number of packets in the buffer and minimum when the buffer is full/nearly full.

V. $MMPP/\widetilde{D}/1/B_{on}$ QUEUE MODEL

We now model the jitter buffer with limited capacity B_{on} as a queueing model with MMPP input and queue length dependent service times. When a packet departs from the queue, queue length can take any one of the values from 0 to $B_{on} - 1$. If the length of the queue is zero, then the algorithm has to wait for the next arrival and start serving it. So, buffer level L_k cannot take zero while calculating service times. If the length of the queue is non-zero, the service time for the next packet is calculated from the queue length and present state of the MMPP at that instant. So, buffer level L_k at the k^{th} packet service initiation epoch takes any one of the values from 1 to $B_{on} - 1$.

Let us re-define,

$$\delta_{k,j} = \left(\frac{B_{on} + 1 - L_k}{2B} \right) X_{M_j} \quad (3)$$

where X_{M_j} is the average inter-arrival time when the MMPP is in state j, so, $X_{M_j} = 1/\lambda_j$.
At the service initiation instant of k^{th} packet, $\delta_{k,j}$ is calculated from Equation (3). Depending on the value of $\delta_{k,j}$, the service time of k^{th} packet is calculated according to the algorithm we use. So, $(B_{on} - 1)$ deterministic service times are possible for each state of the MMPP. Therefore, $M(B_{on} - 1)$ service times: $\{S_{j,l} : 0 \leq j \leq M-1, 1 \leq l \leq B_{on} - 1\}$ are possible. This allows us to model the jitter buffer as an $MMPP/\widetilde{D}/1/B_{on}$ queue, where $\widetilde{D}$ represents a set of $M(B_{on} - 1)$ state dependent service times.

We observe the state of the MMPP and the number of packets in the queue at departure epochs. Let J_k and L_k be the state of the MMPP and the number of packets in the queue at (i) $(k-1)^{th}$ packet departure or (ii) k^{th} packet service initiation instant, both are same except if the $(k-1)^{th}$ packet leaves the system empty. Let L_{k+1} be the same for $(k+1)^{th}$ packet and let A_k be the number of arrivals during the service time of k^{th} packet. Then the following recursive relation holds good:

$$\hat{L_{k+1}} = \begin{cases} L_k + A_k - 1, & if\, L_k > 0 \\ A_k\,, & if\, L_k = 0 \end{cases} \quad (4)$$

Now,

$$L_{k+1} = min(B_{on} - 1,\, \hat{L_{k+1}}) \quad (5)$$

From Equations (4) and (5), it is clear that L_{k+1} depends only on L_k and A_k. Since arrivals in each state of the MMPP follows Poisson process, A_k is i.i.d and the state transitions of the MMPP are governed by a CTMC as mentioned previously. Therefore, $\{J_k, L_k,\, k \geq 0\}$ forms an embedded Markov chain (EMC) with the finite state space $\{0, 1, 2, ..., M-1\} X \{0, 1, 2, ..., B_{on} - 1\}$. At any arbitrary time instant the state of the system is represented by the pair (j, l). The 1-D representation of this state space is $\{(0,0), (0,1), ..., (0, B_{on} - 1), (1,0), (1,1), ..., (1, B_{on} - 1), ...(M-1, 0), (M-1, 1), ..., (M-1, B_{on}-1)\}$. So, we can model the state (j, l) as an an EMC, the transition probability matrix P of the EMC is given in Equation (6),

$P = [P_{ij}] = P(present\, state = i,\, next\, state = j)$

$$P = \begin{bmatrix} P_{0,0} & P_{0,1} & \cdots & \cdots & P_{0,M-1} \\ P_{1,0} & P_{1,1} & \cdots & \cdots & P_{1,M-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ P_{M-1,0} & P_{M-1,1} & \cdots & \cdots & P_{M-1,M-1} \end{bmatrix} \quad (6)$$

where the elements P_{ij} are sub-matrices of size $B_{on} X B_{on}$ [12] and P is of size MXM.

Since we have assumed that multiple state transitions cannot occur between two successive departures, $P_{i,j} = 0$, for $|i - j| > 1$. So, P matrix will have elements in main diagonal and in its two co-diagonals (one above and one below the main diagonal). Elements of the sub-matrix $P_{i,j}$ are denoted as $P_{i,j}(m, l)$ given by,

$$P_{i,j}(m, l) = P(J_{k+1} = j,\, L_{k+1} = l \mid J_k = i,\, L_k = m) \quad (7)$$

which is the probability of l packets in the queue when the MMPP is in state j after a departure given that there were m

$$P_{i,j} = \begin{bmatrix} \alpha(0;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & \alpha(1;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & \alpha(2;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & \cdots & \alpha(B_{on}-2;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & 1-\sum_{k=0}^{B_{on}-2}\alpha(k;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) \\ \alpha(0;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & \alpha(1;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & \alpha(2;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & \cdots & \alpha(B_{on}-2;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) & 1-\sum_{k=0}^{B_{on}-2}\alpha(k;S_{j,1},\lambda_j)\beta(i,j;S_{j,1}) \\ 0 & \alpha(0;S_{j,2},\lambda_j)\beta(i,j;S_{j,2}) & \alpha(1;S_{j,2},\lambda_j)\beta(i,j;S_{j,2}) & \cdots & \alpha(B_{on}-3;S_{j,2},\lambda_j)\beta(i,j;S_{j,2}) & 1-\sum_{k=1}^{B_{on}-2}\alpha(k;S_{j,2},\lambda_j)\beta(i,j;S_{j,2}) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \alpha(0;S_{j,B_{on}-1},\lambda_j)\beta(i,j;S_{j,B_{on}-1}) & 1-\alpha(0;S_{j,B_{on}-1},\lambda_j)\beta(i,j;S_{j,B_{on}-1}) \end{bmatrix} \quad (8)$$

packets in the queue and the MMPP was in state i after the previous departure.

Because of the assumptions we made, $P_{i,j}(m,l)$ can be calculated as the product of $P(L_{k+1}=l \mid J_k=i, L_k=m;$ in the service time $S_{j,l})$ and $P(J_{k+1}=j \mid J_k=i;$ in the service time $S_{j,l})$, where $S_{j,l}$ is the service time of a packet when the buffer level is l and MMPP is in state j. The sub-matrix $P_{i,j}$ is given in Equation (8), where $\alpha(k;S_{j,l},\lambda_j)$ is given below:

$$\alpha(k;S_{j,l},\lambda_j) = \frac{e^{-\lambda_j S_{j,l}}(\lambda_j S_{j,l})^k}{k\,!},\ k \geq 0 \quad (9)$$

which is the probability of k arrivals in the service time $S_{j,l}$ when the arrival rate is λ_j; and $\beta(i,j;S_{j,l})$ is the probability of the MMPP to change its state from i to j in the service time $S_{j,l}$. If MMPP changes its state from i to j, this is equivalent to $|(i-j)|$ arrivals (equal to one arrival, according to the assumptions) in the service time $S_{j,l}$. As mentioned previously, the duration of state j is exponentially distributed with parameter μ_j, so, we can write $\beta(i,j;S_{j,l})$ as follows:

$$\beta(i,j;S_{j,l}) = \begin{cases} \text{P(MMPP will not change its state in service time } S_{j,l}), & \text{if } i=j \\ \text{P(MMPP will change its state from i to j in service time } S_{j,l}), & \text{otherwise} \end{cases}$$

$$= \begin{cases} \frac{e^{-\mu_j S_{j,l}}(\mu_j S_{j,l})^0}{0\,!}, & \text{if } i=j \\ \frac{e^{-\lambda_j S_{j,l}}(\lambda_j S_{j,l})^1}{1\,!}, & \text{otherwise} \end{cases}$$

Let $\pi = \{\pi_{j,l} : 0 \leq j \leq M-1, 0 \leq l \leq B_{on}-1\}$ be steady state probability vector of the states at departure epochs, where $\pi_{j,l}$ denotes the steady state probability of the state (j,l). By solving $\pi = \pi P$ [8], we get the steady state probability vector of state. Now, the steady state probability of buffer occupancy $(\pi_l,\ 0 \leq l \leq B_{on}-1)$ is calculated as,

$$\pi_l = \sum_{j=0}^{M-1} \pi_{j,l}. \quad (10)$$

Now, from [12], we can calculate queue length distribution at arrival epochs, from which mean waiting time and packet loss probability calculation is straightforward.

VI. Simulation Results and Discussion

Simulation parameters are taken in such a way that the two assumptions gets satisfied always. We have taken, the average arrival rates in each state of the MMPP as $\lambda_0 = 0.6\,pptu$ (***packets per time unit***), $\lambda_1 = 0.7\,pptu$, $\lambda_2 = 0.8\,pptu$, $\lambda_3 = 0.9\,pptu$, where one time unit depends on the line speed and packet size. Mean state durations are exponentially distributed and are set as $\mu_0 = 0.02$, $\mu_1 = 0.05$, $\mu_2 = 0.015$, $\mu_3 = 0.001$ and, X_a is taken as $1/mean(\lambda_0,\lambda_1,\lambda_2,\lambda_3)$ in the simulation. In Fig. 2, buffer size B varied from 4 to 30, and set $h=2$, so, B_{on} ranges from 10 to 62.

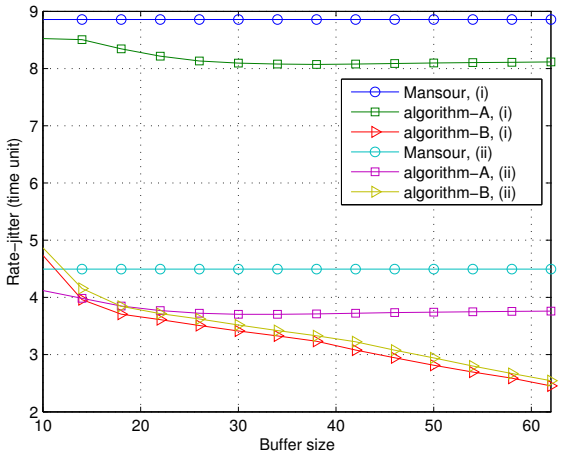


Figure 2: Comparison of rate-jitter for different algorithms in both cases

Choice of I_{max} and I_{min}: One trivial choice for I_{max} and I_{min} is X_{max} and X_{min} [1], where X_{max} and X_{min} are the maximum and minimum IATs in the input sequence, respectively. But by using tighter I_{max} and I_{min}, we may get a stronger rate-jitter guarantee, i.e., we may achieve an I_{max} less than X_{max} and an I_{min} greater than X_{min} using off-line algorithm. It is not possible to give an exact lower bound of I_{max}, since we do not have control over input arrival process. In this paper, we have simulated two **cases**: **(i)** $I_{max} = X_{max}$, $I_{min} = X_{min}$ and **(ii)** $I_{max} = 0.5X_{max}$, $I_{min} = 2X_{min}$. When we reduce I_{max} value further beyond $0.5X_{max}$, we have arrived at a situation where a packet is scheduled before its arrival. The reason for this is the maximum IDT I_{max} is smaller than the IATs of packets arrived in that situation.

From Fig. 2, it is evident that the rate-jitter of algorithms A and B is less compared with Mansour algorithm in both cases. Among the three algorithms, algorithm-B achieves less rate-jitter. Since I_{max} in case (i) is larger than that of case (ii), rate-jitter is more in case (i) compared with case (ii). For one realization of input, the maximum and minimum IDTs

 ISBN: 978-1-61208-263-9

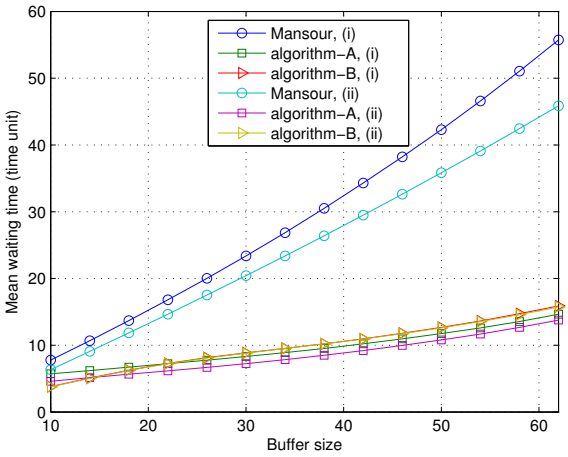


Figure 3: Comparison of mean waiting time of a packet for different algorithms in both cases

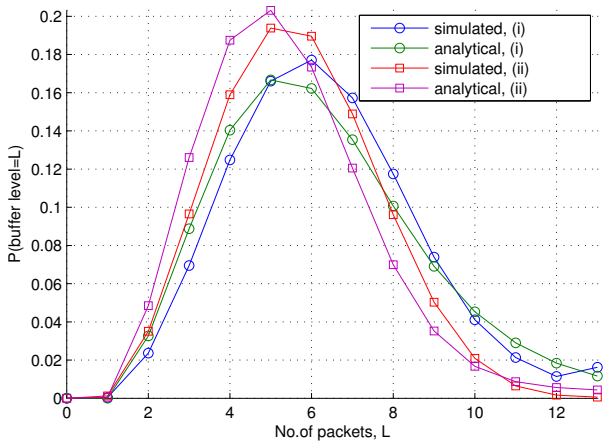


Figure 4: Queue length distribution analytical vs. simulation for both cases when algorithm-A is applied

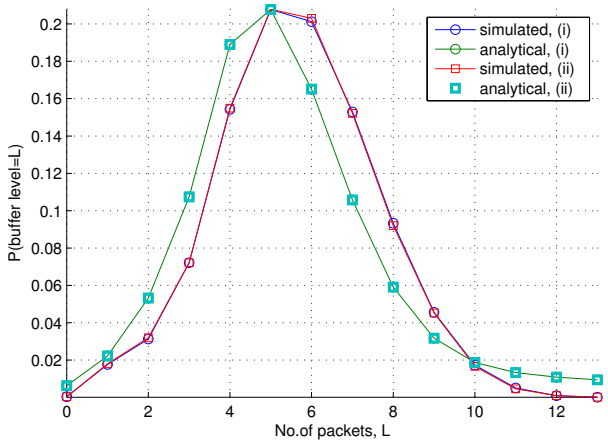


Figure 5: Queue length distribution analytical vs. simulation for both cases when algorithm-B is applied

in the output sequence of Mansour algorithm are I_{max} and I_{min}, respectively, for all buffer sizes. Hence, the rate-jitter is constant.

The maximum IDT is constant in algorithm-A for all buffer sizes, but minimum IDT decreases with an increase in buffer size B up-to some value and from then onwards it is almost constant. Because at fewer values of buffer size, it becomes full and the minimum IDT takes the minimum of the possible values it can take, but at high buffer values it may not happen. Therefore, rate-jitter for algorithm-A decreases up-to some buffer size and almost constant from there in both cases. For algorithm-B, both the maximum and minimum IDTs decreases with an increase in the buffer size. Therefore, rater-jitter keep on decreasing for increasing buffer size in both cases.

The mean waiting time of a packet for different algorithms is calculated from simulation. From Fig. 3, it is observed that in Mansour algorithm, the waiting time increases like an exponential as a function of buffer size B, which is large compared with algorithms A and B in both cases. Mean waiting time of algorithm-A (algorithm-B) is almost equal in both cases.

To simulate the queue length distribution, we have taken $B = 6$, $h = 2$, so, $B_{on} = 14$ for both cases. For algorithm-A, IDTs are taken as service times to find the queue length distribution, this is true as long as the buffer is not empty. Buffer becomes empty rarely because of the following: (a) when the buffer level decreases, algorithm-A schedules the packets in such a way that the IDT is I_{max}, so, the probability of an arrival before the buffer becomes empty is high. (b) since arrivals follow Poisson process in each state of the MMPP, there is a high probability for the IATs (follows Exponential distribution) to take small values. So, most of the time there will be a packet in the buffer. This is true from TDMoIP perspective also because the incoming rate has to be equal to the outgoing rate (utilization, ρ = 1).

From Fig. 4 and Fig. 5, we can observe that the analytical queue length distribution curves for algorithm-A and algorithm-B, respectively, match with the simulation results for both cases. Since algorithm-B doesn't depend on I_{max}, the queue length distribution curves (analytical or simulated) are almost same irrespective of the case, which can be observed from Fig. 5. Even though algorithm-B depends on I_{min}, its value is so small that cannot influence the curves.

VII. Conclusions and Future Work

In this paper, we have proposed two rate-jitter control algorithms for TDMoIP application. For algorithm-A, the bound on rate-jitter is calculated analytically and is shown that is smaller as compared with Mansour algorithm. The simulation results show that the performance of algorithm-A and algorithm-B is better than Mansour algorithm with respect to mean waiting time and rate-jitter. Therefore, algorithm-B is superior to both algorithm-A and Mansour algorithm, but the analytical bound of algorithm-B is yet to be proved. We have modeled the jitter-buffer as an $MMPP/\widetilde{D}/1/B_{on}$ queue, derived an expression for the queue length distribution. The simulation results for both algorithms match the analytical queue length distribution.

 ISBN: 978-1-61208-263-9

The future work is directed towards the study of performance of proposed algorithms in multiple streams environment. The statistical analysis of algorithm-B is under study. We also aim to calculate an analytical expression for variance of the departure process of the proposed queueing model. As a future work, we are extending this work with different self-similar processes as input, because the real-time traffic in IP can be best modeled using these arrival processes. We also aim to determine the waiting time distribution.

REFERENCES

[1] Y. Mansour and B. Patt-Shamir, "Jitter control in QoS networks," IEEE/ACM Trans. on Networking, Vol.9, No.4, Aug 2001, pp. 492-502.

[2] T. A. ElBatt, S. El-Henaoui, and S. Shaheen, "Jitter recovery strategies for multimedia traffic in ATM networks," in Proc. GLOBECOM'96, vol.2, 1996, pp. 1202-1206.

[3] H. Zhang and S. Keshav, "Comparison of rate-based services disciplines," in Proc. ACM SIGCOMM, 1991, pp. 113-121.

[4] D. Hay and G. Scalosub, "Jitter regulation for multiple streams," ACM Trans. on Algorithms, Vol.6, No.1, Article 12, Dec 2009, pp. 12.1-12.19, doi: 10.1145/1644015.1644027.

[5] S. Jagadish and R. Manivasakan, "Analysis of jitter control algorithms in QoS networks," in Proc. Second Asian Himalayas International Conference on Internet (AH-ICI), Nov 2011.

[6] Z. Yan and J. Yih-Chyun, "Performance analysis of adaptive timing method for voice over ATM using queuing models," in Proc. of Winter International Symposium on Information and Communication Technologies (WISICT'04), 2004, pp. 1-8.

[7] K. Sriram and D. M. Lucatoni, "Traffic smoothing effects of bit dropping in a packet voice multiplexer," IEEE Trans. on Comm.,Vol.37, No.7, July 1989, pp. 703-712.

[8] U. Narayan Bhat, "An Introduction to queueing theory: Modeling and analysis in applications," Springer, 2008.

[9] D. Gross and C. M. Harris, "Fundamentals of queueing theory," New York: Wiley, 1985, 2nd ed., pp. 279-285.

[10] W. Fischer and K. Meier-Hellstern, "The Markov-modulated Poisson process (MMPP) cookbook," in Perf. Eval, Elsevier, Vol.18, No.2, Sep 1993, pp. 149-171.

[11] F. Yegenoglu and B. Jabbari, "Modeling of aggregated bursty traffic sources in ATM multiplexers," in Proc. ICC'93, May 1993, pp. 1703-1707.

[12] F. Yegenoglu and B. Jabbari, "Performance evaluation of MMPP/D/1/K queues for aggregate ATM traffic models," in Proc. INFOCOM'93, 1993, pp. 1314-1319.

[13] B. D. Choi and D. I. Choi, "Queueing system with queue length dependent service times and its application to cell discarding scheme in ATM networks," in Proc. IEE, 1996, pp. 5-11.

[14] Madhu Sikha and R. Manivasakan, "Novel Rate-Jitter Control Algorithms for TDMoIP, " in IEEE National Conference on Communications (NCC), 2013, pp. 1-5.

[15] A. T. Andersen and B. F. Nielsen, "A Markovian approach for modeling packet traffic with long-range dependence," IEEE Journal on Selected Areas in Comm., Vol.16, No.5, June 1998, pp. 719-732.

[16] L. Muscariello, M. Mellia, M. Meo, M. Ajmone Marsan, and R. Lo Cigno, "Markov models of internet traffic and a new hierarchical MMPP model," Journal on Computer Communications, Elsevier, Vol.28, No.16, Oct 2005, pp. 1835-1851.

[17] T. Yoshihara, S. Kasahara, and T. Takahashi, "Practical time-scale fitting of self-similar traffic with Markov-modulated Poisson process," Telecommunication Systems, Vol. 17, No. 1-2, 2001, pp. 185-211.

[18] A. Nogueira, P. Salvador, R. Valadas, and A. Pacheco, "Modeling self-similar traffic through Markov modulated Poisson processes over multiple time scales," in Proc. 6th IEEE International Conference on High Speed Networks and Multimedia Communications (HSNMC'03), July 2003.

[19] Y. Stein, R. Shashoua, R. Insler, and M. Anavi,"Time Division Multiplexing over IP (TDMoIP)," RFC 5087, Dec 2007.

[20] P. M. Fernandez, "Circuit Switching in the Internet'," Doctoral thesis, Chapter 2, Stanford University, June 2003.

[21] Y. Stein and E. Schwartz, "Circuit Extension over IP: An Evolutionary Approach to Transporting Voice and Legacy Data over IP Networks," RAD Data Comm., March 2001.

 ISBN: 978-1-61208-263-9

Coherent Pre-Distortion of Low-Frequency PLC Carriers

Stan McClellan
Ingram School of Engineering
Texas State University
San Marcos, TX USA
stan.mcclellan@txstate.edu

Michael L. Casey
Ingram School of Engineering
Texas State University
San Marcos, TX USA
mlcasey@txstate.edu

Matthias Chung
Dept. of Mathematics
Virginia Tech
Blacksburg, VA USA
mcchung@vt.edu

***Abstract*—The use of power lines for communication is a topic of great practical interest as well as active research and standardization activities. The introduction of communications signals onto an energized power line can cause significant distortion of the signal, especially at low frequencies. This paper describes a form of coherent pre-distortion for such communication signals which reduces the destructive interference. The approach can be optimized in several ways to mitigate distortion and achieve more efficient data transfer, and may be important in cases where low-frequency, low-rate carriers are used.**

***Keywords*-Power line communications; PLC; pre-distortion;**

I. INTRODUCTION

Transmission of data via an active power line is a difficult task, particularly at carrier frequencies below 2kHz [1]. As a result, the majority of activity in Power Line Communications (PLC) for Smart Grid applications has been in higher frequency bands [2]. At Very Low Frequencies (VLF), the pre-existing power signal on the line (the 50Hz or 60Hz "fundamental") causes a number of problems in the system, including pseudo-stationary interference from harmonics of the fundamental and a form of "blowback" into the data transmitter. However, data communications at low frequencies can be very useful in a number of applications, such as Automatic Meter Reading (AMR), Advanced Metering Infrastructure (AMI) and similar command/control and data retrieval scenarios [1], [3].

Curiously, the structure of the power line network seems to interact in a specific fashion with low-frequency communication signals. This interaction can appear as a form of amplitude modulation of the communication signal, where the modulation envelope is phase-coherent or time-synchronous with the fundamental. An example of this phenomenon is shown in the top plot of Fig. 3. When present, this distortion envelope seems to be imposed on any/all secondary, low-frequency signals in the channel. We have not found any reference to this phenomenon in the literature. This type of channel-induced distortion is problematic for data communications because amplitude modulation creates harmonically-related images of the carrier frequency. As image signals proliferate, the availability of idle or useable spectrum for additional subcarriers is reduced, and the dynamic placement of subcarriers becomes difficult.

An approach to counteracting such effects is to pre-distort the communication signal prior to introducing it to into the channel. In this fashion, the distortion and pre-distortion effectively cancel each other out. The concept of pre-distortion not new. For example, pre-distortion is often used to counteract nonlinearities in power amplifiers for wireless communications [4]–[6]. However, in the case of VLF PLC, nonlinearities seem to be introduced by the *channel itself*, which is a multi-port, multi-user network with time-varying characteristics, and so is very difficult to characterize completely. Pre-distortion techniques are also particularly important in channels with several subcarriers, or in systems using dynamic spectral allocation [7]–[10]. In the case of VLF PLC without effective pre-distortion, a multi-carrier PLC transmitter may be *prevented* from introducing additional subcarriers because the image signals from each subcarrier interfere very significantly with neighboring spectral bands. Furthermore, VLF PLC transmissions are very bursty and have very low data rates, making conventional equalization or filter-based pre-distortion difficult. Thus, we require a pre-distortion system which has a relatively simple formulation, is instantaneously applicable (no convergence delays), and effectively mediates spectral imaging due to channel effects.

This paper describes a specific approach to pre-distortion of a communication signal prior to introducing it onto an active power line in a VLF PLC system. The specific structure of the pre-distorted signal is important because it must be estimated very quickly from some functions of the signals on the channel. In our experimentation with VLF PLC on a live testbed, we have observed that for particular types of signals, the channel-induced distortion is synchronous with the fundamental. Refer to the top plot in Fig. 3 for an example of this phenomenon. Thus, we derive the form of the pre-distortion envelope from a model which is a linear combination of powers of the fundamental. Using this approach, the pre-distortion function can be computed instantaneously from observed samples of the fundamental. Thus, the method observes the extant voltage on the power network and, using the model of the distortion envelope, computes a pre-distortion function which can be imposed

 ISBN: 978-1-61208-263-9

on the communication signal just prior to introduction to the power network. In this fashion, we achieve a pre-distortion envelope which is a very close approximation to the amplitude distortion envelope imposed by the power network, and synchronous distortion is significantly reduced.

The remainder of this paper presents the approach to pre-distortion, including some motivating factors as well as well as experimental results exploring optimization of model parameters. Section II presents the mathematical formulation of the proposed channel model and pre-distortion function. The structure of the model is motivated using important considerations such as simplicity of form, ease of optimization, and synchronization with the fundamental. Section III presents experimental results using various configurations of the model to suppress channel-induced distortion. The effect of model order is explored using unit-valued coefficients, and the effect of coefficient optimization is discussed. Section IV concludes the paper by proposing extensions of the work to algorithmic optimization and implementation in a real-time system. The data used to develop this predistortion technique was gathered from an experimental powerline communication system which is being implemented on a production distribution grid. Simulations were developed to model the channel effects based on the acquired data, and were tested on a non-real-time PLC system. Implementation and optimization of the pre-distortion technique in a real-time, multi-carrier PLC system is ongoing.

II. Mathematical Formulation

The mathematical formulation of the pre-distorted signal and the method by which it is derived from observations of the fundamental are important factors in understanding the technique. In effect, the process creates a set of basis functions from an observation of the fundamental. These basis functions are combined and used to calculate the pre-distortion envelope, which is then applied to the communication signal. If the set of basis functions and their combination are a reasonable model of the unknown process creating amplitude distortion in the channel, then the two effects will counteract each other, leaving the communication signal in the channel undistorted. So, we propose a model based on a linear combination of functions of the power signal, and we show via simulation and deployment on a live testbed that this model effectively suppresses particular types of channel-induced distortion in VLF PLC.

Let $p(t)$, or simply p, be a power signal in the time domain with Fourier Transform $P(\xi) = \mathcal{F}(p)$. In a perfect channel, p would be simply a sinusoidal power signal with frequency 50Hz or 60Hz (the fundamental). In a realistic channel, p is the entire signal observed, which is largely sinusoidal but may contain some harmonic content, noise, etc. Much of the nonsinusoidal content is due to the electrical structure of the channel and the fundamental excitation.

Let $x(t)$, or x, be a communication signal in the time domain with Fourier Transform $X(\xi) = \mathcal{F}(x)$. In the simplest case, x might be a sinusoid at some frequency other than 50Hz or 60Hz. In other cases, x might be a more complex passband carrier, modulated via analog or digital means to carry a message signal or data bits.

The objective is to transmit communication signal x through the power line system so that the resulting signal can be recovered by the receiver without error. Unfortunately, during transmission the power line system distorts x via an unknown transfer function $H_p(x)$. Due to the nature of the system, the transfer function does not affect the power signal p. In fact, the transfer function H_p seems to depend on p in some fashion, and although H_p cannot be observed directly, the structure of H_p can be partially estimated via observation of p. Since the effect of H_p is similar to conventional amplitude modulation, appropriate pre-distortion of x via an inverse-function H_p^{-1} will approximately account for the channel's envelope distortion, thereby suppressing any unwanted noise or spectral artifacts which are related to the interaction of x, p, and H_p. Mathematically,

$$H_p(H_p^{-1}(x)) = x. \tag{1}$$

Computing H_p^{-1} precisely is impossible. Fortunately, we can estimate the structure of H_p using powers of the fundamental p. This produces the approximation $\hat{H}_p^{-1}$ and the pre-distorted signal $y = \hat{H}_p^{-1}(x)$ so that (1) becomes

$$H_p(y) = \hat{x} \approx x. \tag{2}$$

By choosing the structure of $\hat{H}_p^{-1}$ carefully (the model), any errors in the approximation of H_p can be reduced significantly via one of several well-known methods [11]–[13].

A. Modeling H_p

Although H_p is difficult to model, we have observed that the channel produces a coherent amplitude modulation of secondary signals such as x. Refer to the top plot of Fig. 3 for an example of this phenomenon. By analyzing data acquired from an experimental PLC system which evidences this behavior, the authors have discovered that the channel-induced modulation can be partially modeled using a linear combination of powers of p, or

$$q(t, \boldsymbol{\alpha}) = \sum_{j=1}^{N} \alpha_j \left[p(t)\right]^j, \tag{3}$$

where coefficients $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)^\top$ are initially unknown and must be optimized for each instance of the channel or re-optimized over time. A representative p and q are shown in Fig. 1 where the coefficients $\boldsymbol{\alpha}$ have been estimated manually, p and q are normalized to unit amplitude, and q is formulated using only odd powers of p (i.e. $\alpha_j = 0$ for j even).

 ISBN: 978-1-61208-263-9

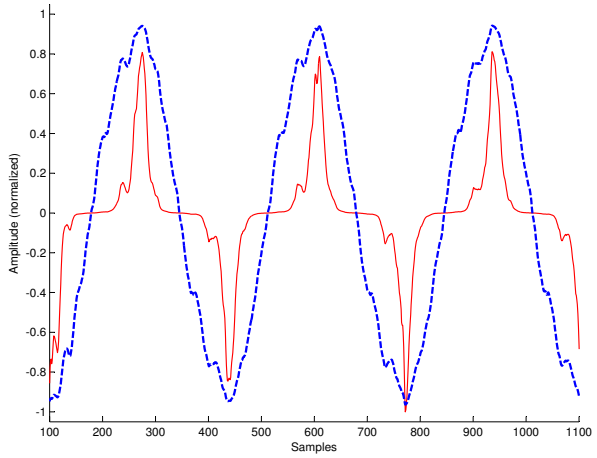


Figure 1. p (dashed line) and q (solid line).

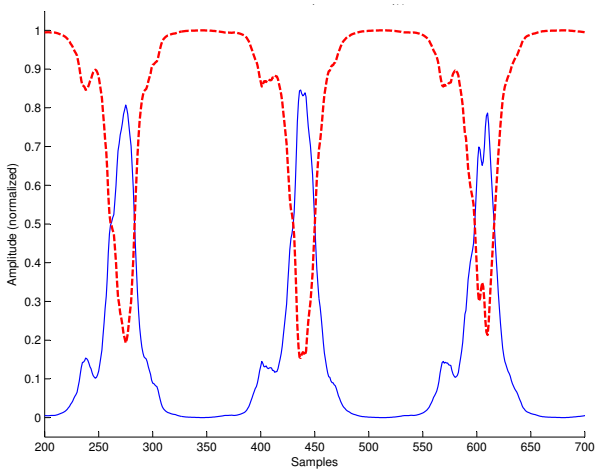


Figure 2. Estimated channel modulation envelope H_p (dashed line) and corresponding pre-distortion envelope $\hat{H}_p^{-1}$ (solid line). Both signals are normalized to unit amplitude, and are shown prior to scaling for appropriate modulation depth.

Using terms familiar to communications systems, the channel constructs a "false message" which is imposed on communication signal x as it transits the channel. Here, we use the unknown transfer function H_p to represent the false message signal. Estimating H_p via q provides an easily-constructed relationship given by:

$$\hat{H}_p(t, \boldsymbol{\alpha}) = -|q(t)| + \delta. \tag{4}$$

In (4), the false message H_p is estimated by $\hat{H}_p$, which depends on q through coefficients $\boldsymbol{\alpha}$. The false message can be counteracted by pre-distorting x with $\hat{H}_p^{-1}$. As in typical modulation practices, all envelope functions are normalized to unit amplitude prior to use. In (4), δ is a constant related to the modulation depth, as is customary for amplitude modulation envelopes [14]. A representative modulation envelope H_p recovered from an actual power signal is shown in Fig. 2, along with the estimated inverse envelope $\hat{H}_p^{-1}$. In the figure, the signals are shown normalized to unit amplitude, and have not been scaled for correct modulation depth, i.e. $\delta = 0$.

A representative distorted signal $H_p(x)$ is shown in the top plot of Fig. 3. In this case, x is a sinusoid with frequency of roughly 900Hz. Note the subtle amplitude modulation effects of H_p on x. The amplitude envelope of the modulated signal has periodic notches which are synchronized with the peaks of p. These time-domain attributes are implicitly modeled very accurately and simply by q and hence $\hat{H}_p$. Thus, the envelope can easily be translated to an optimized pre-distortion implementation $\hat{H}_p^{-1}$ which does not require complex feedback loops or phase discrimination/locking techniques.

B. Coherent pre-distortion

To achieve effective communication, the modulation imposed by the channel via H_p must be pre-distorted by an inverse function. The pre-distortion approach involves estimating and optimizing the coefficients of q and then formulating an estimated false message signal $\hat{H}_p$ which can be applied to x prior to introduction to the channel via $\hat{H}_p^{-1}$. Upon introduction to the channel, the channel transfer function re-imposes the false message H_p onto the pre-distorted signal. In this fashion, the distortion due to the channel can be modeled as in (2), subject to the fidelity of $\boldsymbol{\alpha}$, q, and hence $\hat{H}_p$. The outcome of this process is shown in Fig. 3, which displays plots of the channel signal $H_p(x)$, the pre-distorted signal y, and the resulting suppressed signal $\hat{x} \approx x$. For the figure, the communication signal x (not shown) was a low-rate BPSK-modulated carrier at approximately 900Hz, and the resultant signals are shown offset for clarity and normalized to approximately unit amplitude.

The optimization of the pre-distortion envelope $\hat{H}_p^{-1}$ and the coefficients $\boldsymbol{\alpha}$ so that $\hat{x} \approx x$ is an extremely complex problem, and the subject of further study. Clearly, when $\hat{H}_p$ equals H_p exactly, the communication signal x will transit the channel undistorted by coherent amplitude modulation. Unfortunately, as mentioned previously, the form of H_p is unknown, and must be modeled via q, so that determining an envelope function $\hat{H}_p \approx H_p$ is not trivial.

The pre-distortion envelope $\hat{H}_p^{-1}$ will be used in a conventional double-sideband amplitude modulation (DSB-AM) [14], and so must be normalized to unit amplitude before use. Thus, a formulaic representation of $\hat{H}_p^{-1}$ is:

$$\hat{H}_p^{-1}(t, \boldsymbol{\alpha}) = \frac{|q(t)| + \varepsilon}{c}, \tag{5}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)^\top$, $\varepsilon > 0$, and $c = \max\{|q(t)| + \varepsilon\}$ to ensure unit amplitude. A representative envelope is shown in Fig. 2 (solid line), and the effect of applying the envelope to a communication signal x is shown in Fig. 3 (middle plot).

 ISBN: 978-1-61208-263-9

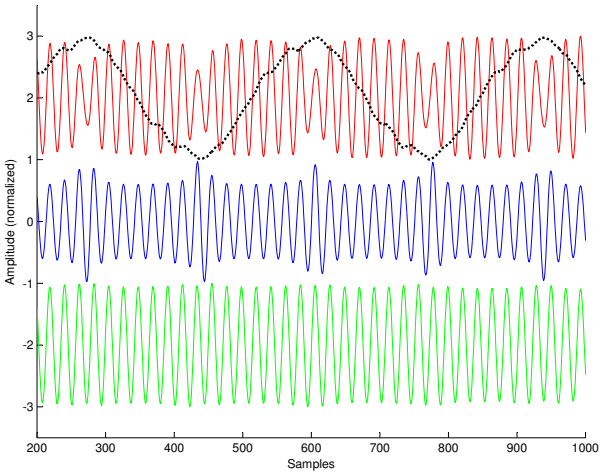


Figure 3. Top: distorted signal $H_p(x)$ and power signal $p(t)$ (dotted line); Middle: pre-distorted signal y; Bottom: resulting signal $\hat{x}$, which approximates x. All signals are normalized to unit amplitude, and have been offset for clear display.

In Fig. 3 the coefficients $\boldsymbol{\alpha}$ have been estimated manually. However even with non-optimal coefficients, the fidelity of the pre-distortion process leads to accurate reconstruction of the signal x, as can be seen from the bottom plot in Fig. 3 ($\hat{x}$) and the spectral plot in Fig. 5.

Estimation and inversion of the false message reduces to an optimization problem which depends on a linear combination of basis functions and the set of coefficients $\boldsymbol{\alpha}$. A number of well-known methods exist for optimizing coefficients of this form, such as Least Mean Squares (LMS), Recursive Least Squares (RLS), etc. [15], [16]. However, optimization of the coefficients $\boldsymbol{\alpha}$ leads to a non-smooth optimization problem which needs to be targeted either by direct search methods [11] or by gradient-based optimization methods after a smoothing process. Note that the signal p is changing slowly over time, so the introduction of a windowed or framed approach is likely optimal, but forward-adaptive or backward-adaptive methods for re-optimizing $\boldsymbol{\alpha}$ are subjects of further study.

III. Results Using Non-Optimized Parameters

The optimal form of the pre-distortion problem is important, but is also very difficult and the subject of considerable research effort by the authors using both simulations and testing on an experimental PLC system. However, even in the absence of complete optimization we can demonstrate the utility of the proposed model using *manually optimized* parameters. In the case of manual optimization, there are two primary dimensions in (3) that must be considered: (a) the values of $\boldsymbol{\alpha}$, and (b) the order of the model, N. Following subsections describe evaluation of the parameters of (3) and the overall predistortion process via model order and coefficient selection.

A. Effect of Model Order

Important optimizations for the pre-distortion system are the selection of "best" model order (N) and resulting coefficient structure. We have noticed via both simulation and experimentation in a live PLC system that the effect of odd-powers and even-powers of p in (3) and the pre-distortion envelope in (5) is pronounced and important. Thus, we briefly examine the effects of model order and coefficient structure in the formulation of q. To evaluate the noted effects, we simulate the pre-distortion system using model configurations with unit-valued coefficients and odd-only or even-only powers of p and model orders $N < 100$. Using this simulation, we compute the usual zero-mean signal-to-noise ratio (SNR) between the compensated signal $\hat{x}$ and the original signal x according to (6), where $x[n]$ denotes the discrete-time (sampled) signal x.

$$\text{SNR (dB)} = 10\ log_{10}\left(\frac{\sum x[n]^2}{\sum (x[n]-\hat{x}[n])^2}\right) \qquad (6)$$

Fig. 4 summarizes the simulation results for distortion versus model structure using (3) and the resultant pre-distortion process. In the figure, the curve labeled "all coefficients" has $\alpha_j = 1\ \forall\ j$, whereas the curve labeled "even coefficients" has $\alpha_j = 1$ for j even, and $\alpha_j = 0$ otherwise (similarly for "odd coefficients"). Note from the figure that although the combined odd/even model seems to result in reasonable distortion for lower model orders ($N < 10$), the effect of model order is very pronounced. So, for lower model orders and combined odd/even model construction, the system is extremely sensitive to variations in model order and input data. Conversely, for moderate model orders ($10 < N < 50$) both the odd-only and even-only model constructions perform better than the combined construction, and exhibit very little sensitivity to model order. Notably, for moderate model orders, the odd-only construction ((3) with $\alpha_j = 1$, j odd) reaches a *higher maximum* SNR, and evidences a *smoother ascent*. As a result, the process of optimizing odd-only coefficient vectors $\boldsymbol{\alpha}$ may be less prone to local extrema. Additionally, for large model orders ($N > 50$), the differential distortion between odd-only, even-only, and odd-even model constructions is insignificant. From this result, we conclude that the use of odd-only powers of p in (3) produces harmonic suppression which is more effective for subbands which are more prevalent in the distortion characteristic of the channel.

B. Manual Optimization

Illustrative results for manually optimized coefficients $\boldsymbol{\alpha}$ are shown in Fig. 3, where an approximately sinusoidal signal x with frequency around 900Hz is introduced to the channel. The figure uses data acquired from our VLF PLC testbed. In the figure, the effect of the coherent, channel induced distortion $H_p(x)$ is evident in the top plot, and the pre-distorted signal y in the middle plot is fed to the channel,

 ISBN: 978-1-61208-263-9

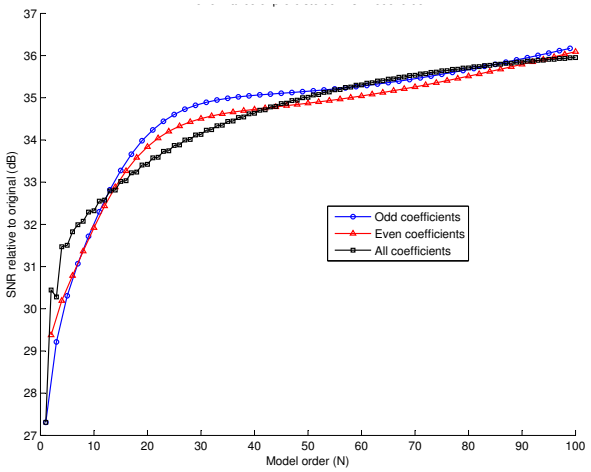


Figure 4. Distortion versus model order for odd-only, even-only, and odd/even model constructions with unit-valued coefficients.

resulting in the bottom plot where $\hat{x} \approx x$ and the distortion is suppressed.

In the top plot of Fig. 3, note the subtle modulation of the amplitude of x due to H_p. The periodic notches in the amplitude of $H_p(x)$ are synchronous with the peaks of p, which is shown overlaid on the top plot of Fig. 3 with unit amplitude. In the middle plot, note the corresponding inverse amplitude modulation of x due to $\hat{H}_p^{-1}$. The periodic peaks in the amplitude of y are aligned with the peaks of p, and hence are also aligned with the periodic notches in $H_p(x)$. These time-domain attributes are accurately modeled by q and hence y, which results in a straightforward pre-distortion approach. Thus, when the pre-distorted signal y is introduced to the channel, $H_p(y) = \hat{x} \approx x$ as in (2), and as shown in the bottom plot of Fig. 3.

The effect of this process becomes particularly clear in Fig. 5 which overlays spectra of the idle channel, the channel with distorted input, and the compensated signal $\hat{x}$, and the inset, which overlays spectra of the distortion before and after the pre-distortion process. In the figures, a low-rate BPSK signal x is introduced to the channel with a carrier frequency near 900Hz. When the pre-distortion scheme is not used, the images or sidelobes of the introduced signal are clearly evident in both figures at 120Hz harmonic offsets from the carrier (i.e. $900\text{Hz} \pm (n \times 120\text{Hz})$. However, when the pre-distortion scheme is used, the images of x are suppressed significantly. Also note the roughly 20dB suppression of the image signals nearest the 900Hz carrier (cf. 400-800Hz and 1000-1400Hz). In this case, the differential distortion between the original communication signal x and the pre-distorted signal $\hat{x}$ is less than 0.5dB. Note also that imperfections due to non-optimal coefficients $\boldsymbol{\alpha}$ results in two areas which need additional optimization: (a) spurious peaks at distant 120Hz harmonic offsets from the 900Hz carrier (cf. 1600-1800 and 100-200Hz), and (b) a slowly varying spectral envelope. In these tests, we do not compare end-to-end performance metrics such as bit-error rate (BER) because the effect of the channel-induced distortion can easily be mitigated for single-carrier systems via the use of a high-quality receive filter. Instead, our VLF PLC system is targeted for multi-carrier architecture using dynamic channel selection, and the suppression of coherent images is a critical first-step in the implementation of that architecture.

IV. Conclusion

The suppression of image signals in low-frequency, narrowband PLC systems can be important. When optimized and deployed in a system which continuously re-optimizes the coefficients $\boldsymbol{\alpha}$, the near-field suppression approach described here may yield significant benefits for transmission schemes which rely on large numbers of low-rate carriers, such as frequency-division modulation (FDM) or computationally efficient equivalent approaches based on transforms, such as orthogonal FDM (OFDM).

The pre-distortion model proposed here has been shown to be efficiently realized and capable of instantaneous implementation with no requirements for phase-locking or delay due to convergence of an adaptive filter. These implementation details are extremely important in a VLF PLC system with bursty, low-rate transmissions. In initial testing of the approach using non-adaptive, hand-optimized coefficients, we have achieved greater than 20dB suppression of channel-induced distortion in critical subbands, making these areas of the spectrum available for dynamic allocation by the transmitter.

The authors are actively pursuing algorithmic optimizations of critical model parameters, including model order, coefficient selection, and adaptation rate for implementation in a real-time, low-frequency PLC communication system being implemented on a local distribution grid.

References

[1] D. Rieken, "Periodic noise in very low frequency power-line communications," in *IEEE Int. Symp. Power Line Comms and Appl. (ISPLC)*, Apr. 2011, pp. 295 – 300.

[2] M. Nassar, J. Lin, Y. Mortazavi, A. Dabak, I. H. Kim, and B. L. Evans, "Local utility powerline communications in the 3-500 kHz band: Channel impairments, noise, and standards," *IEEE Sig. Proc. Mag. Special Issue on Sig. Proc. Techn. for the Smart Grid*, vol. 29, no. 5, pp. 116–127, Sep. 2012.

[3] S. Galli, A. Scaglione, and Z. Wang, "Power line communications and the Smart Grid," in *IEEE Int. Conf. Smart Grid Comms (SmartGridComm)*, Oct. 2010, pp. 303 – 308.

[4] Y. Y. Woo, J. Kim, J. Yi, S. Hong, I. Kim, and B. Kim, "Adaptive digital feedback predistortion technique for linearizing power amplifiers," *IEEE Trans. Microw. Theory Tech.*, vol. 55, no. 5, pp. 932–940, 2007.

 ISBN: 978-1-61208-263-9

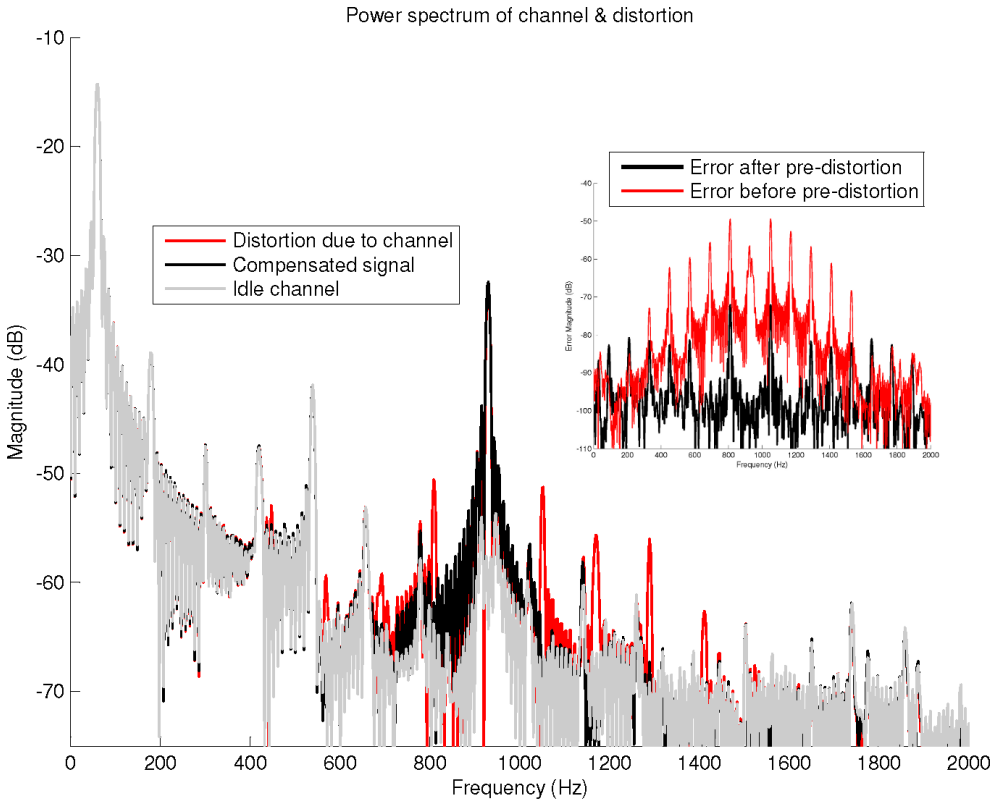


Figure 5. Power spectrum of channel before and after pre-distortion process via estimates of q and application of pre-distortion envelope $\hat{H}_p^{-1}$. The error signal spectra are shown in the inset. The spectrum of the compensated signal $\hat{x}$ and the spectrum of the stimulus signal x are almost identical, and the channel-induced distortion is effectively suppressed.

[5] S. Chung and J. L. Dawson, "Digital predistortion using quadrature $\Delta\Sigma$ modulation with fast adaptation for wlan power amplifiers," in *IEEE Microwave Symposium Digest (MTT)*, June 2011, pp. 1–4.

[6] J. X. Qiu, D. K. Abe, T. M. Antonsen, B. G. Danly, B. Levush, and R. E. Myers, "Linearizability of TWTAs using predistortion techniques," *IEEE Trans. Electron Devices*, vol. 52, no. 5, pp. 718–727, 2005.

[7] J. Shen, S. Liu, Y. Wang, G. Xie, H. Rashvand, and Y. Liu, "Robust energy detection in cognitive radio," *IET Communications*, vol. 3, no. 6, pp. 1016–1023, June 2009.

[8] P. D. Sutton, K. Nolan, and L. Doyle, "Cyclostationary signatures in practical cognitive radio applications," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 13–24, 2008.

[9] B. Farhang-Boroujeny and R. Kempter, "Multicarrier communication techniques for spectrum sensing and communication in cognitive radios," *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 80–85, Apr. 2008.

[10] P. Pawelczak, K. Nolan, L. Doyle, S. W. Oh, and D. Cabric, "Cognitive radio: Ten years of experimentation and development," *IEEE Commun. Mag.*, vol. 49, no. 3, pp. 90–100, 2011.

[11] J. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, pp. 308–313, 1965.

[12] R. Fletcher, *Practical Methods of Optimization*. New York, NY: Wiley, 1987.

[13] H. Matthies and G. Strang, "The solution of non linear finite element equations," *Int. J. Num. Methods in Engr.*, vol. 14, pp. 1613–1626, 1979.

[14] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*. Prospect Heights, IL: Waveland Press, 1990.

[15] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall, 1986.

[16] B. Widrow and S. Stearns, *Adaptive Signal Processing*. Prentice-Hall, 1985.

 ISBN: 978-1-61208-263-9

Performance of Turbo Coded 64-QAM with Joint Source Channel Decoding, Adaptive Scaling and Prioritised Constellation Mapping

Tulsi Pawan Fowdur, Yogesh Beeharry and K.M. Sunjiv Soyjaudah
Dept of Electrical and Electronic Engineering
University of Mauritius
Reduit, Mauritius
e-mail: p.fowdur@uom.ac.mu, yogesh536@hotmail.com, ssoyjaudah@uom.ac.mu

Abstract— **Turbo coded 64-QAM systems have been adopted by standards such as CDMA-2000 and Long Term Evolution (LTE) to achieve high data rates. Although several techniques have been developed to improve the performance of Turbo coded QAM systems, combinations of these techniques to produce hybrids with better performances, have not been fully exploited. This paper proposes a combination of Joint Source Channel Decoding (JSCD), adaptive Sign Division Ratio (SDR) based scaling and prioritised constellation mapping, to improve the performance of Turbo coded 64-QAM. JSCD exploits a-priori source statistics at the decoder side and SDR based scaling provides a scale factor for the extrinsic information as well as a stopping criterion. Additionally, prioritised constellation mapping exploits the inherent Unequal Error Protection (UEP) characteristic of the 64-QAM constellation and provides greater protection to the systematic bits of the Turbo encoder. Simulation results show that at Bit Error Rates (BERs) above 10^{-1}, the combination of these three techniques achieves an average gain of 2.5 dB over a conventional Turbo coded 64-QAM system. However, at BERs below 10^{-1}, the combination of only JSCD and SDR scaling provides an average gain of 1 dB.**

Keywords- Turbo Code; QAM; JSCD; SDR; Priortised Mapping.

I. INTRODUCTION

Since the inspection of Turbo codes by Berrou *et.al* in 1993 [1], several communication standards have adopted this powerful near Shannon limit error correcting code. For example, Turbo coded 64-QAM systems have been widely exploited to achieve reliable transmission at high data rates in several standards such as Long Term Evolution (LTE) [2],[3], CDMA 2000 [4] and HomePlug Green PHY [5]. These systems have also been reported to be promising for IEEE 802.11a [6]. The major impact of Turbo codes has led to the emergence of several techniques such as Joint Source Channel Decoding (JSCD) [7], [8], [9], [10], extrinsic information scaling and iterative detection [11], [12], [13], [14], to improve its error performance and lower its decoding complexity. Moreover, certain characteristics of the 64-QAM constellation have also been exploited to improve the performance of Turbo coded QAM [15]. An overview of these techniques is given below.

JSCD essentially involves the use of a-priori source statistics and the exploitation of residual redundancy to enhance the channel decoding process. For example, Murad and Fuja [7] proposed a composite trellis, made up of a Markov source, a Variable Length Code (VLC), and a channel decoder's state transitions, to exploit a priori source statistics. A low complexity version of the technique in [7] was developed by Jeanne *et.al* [8] and more recently Xiang and Lu [9] proposed a JSCD scheme for Huffman encoded multiple sources, which could exploit the a-priori bit probabilities in multiple sources. Also, Fowdur and Soyjaudah [10] proposed a JSCD scheme with iterative bit combining, which incorporated two types of a-priori information, leading to significant performance gains. On the other hand, extrinsic information scaling aims at improving the Turbo decoder's performance by scaling its extrinsic information with a scale factor. For example, Vogt and Finger [11] used a fixed scale factor to improve the Max-Log-MAP Turbo decoding algorithm, while Gnanasekaran and Duraiswamy [12] proposed a modified MAP algorithm using a fixed scale factor. Interestingly though Lin *et.al* [13] proposed a scaling scheme that extended the Sign Division Ratio (SDR) technique of Wu *et.al* [14] to adaptively determine a scaling factor for each data block at every iteration. Finally, the Turbo decoding process can be further enhanced by exploiting the UEP characteristic of the 64-QAM constellation to give more protection to the systematic bits of the Turbo encoder. This technique has been applied to LTE Turbo codes by Lüders *et.al* [15].

In contrast with previous works, which have mostly considered the schemes developed to improve the performance of Turbo codes independently, this paper analyses the performance of a Turbo coded 64-QAM scheme, which integrates three different techniques. Firstly, at the encoder side, prioritized constellation mapping [15] is performed so that the systematic bits output by the Turbo encoder are given the highest protection when they are mapped onto the 64-QAM constellation. The second technique employed is JSCD [7], [10], which exploits a-priori source statistics during Turbo decoding. The final technique used is adaptive extrinsic scaling based on the SDR criterion [13]. Significant performance gains are obtained for both iterative and non-iterative decoding with the combination of these three techniques.

The organization of this paper is as follows. Section II describes the complete system model. Section III presents the simulation results and analysis. Section IV concludes the paper and lists some possible future works.

 ISBN: 978-1-61208-263-9

II. SYSTEM MODEL

The complete transmission system is shown in Fig. 1.

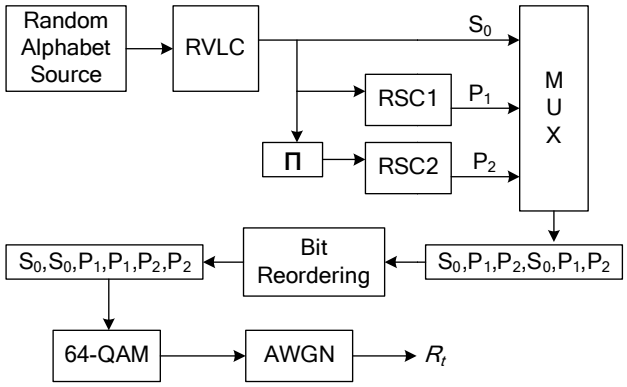


Fig. 1 Transmitter with prioritised constellation mapping.

A random alphabet source is first generated with a non-uniform probability distribution and then encoded into bits with the Reversible Variable Length Code (RVLC) of [16]. The coded bits are fed to a Turbo encoder, which consists of a parallel concatenation of two Recursive Systematic Convolutional (RSC) encoders, RSC1 and RSC2, separated by an interleaver, $\prod$. The Turbo encoder generates a systematic stream, S_0 and two parity streams P_1 and P_2. To achieve prioritized constellation mapping, such that the systematic bits, S_0, are placed at the most strongly protected points on the 64-QAM constellation, bit reordering [15] must be performed after the multiplexing process. The bit reordering is performed on a group of six bits at a time since six bits are mapped onto one complex 64-QAM symbol.

From Fig. 1, it is observed that after bit re-ordering, the parity bits S_0 occupy the first two positions of the six bits that are mapped on one symbol of the 64-QAM constellation shown in Fig. 2. In this constellation, the bits found in the first two positions are most protected, while the bits found in the last two positions receive the lowest protection.

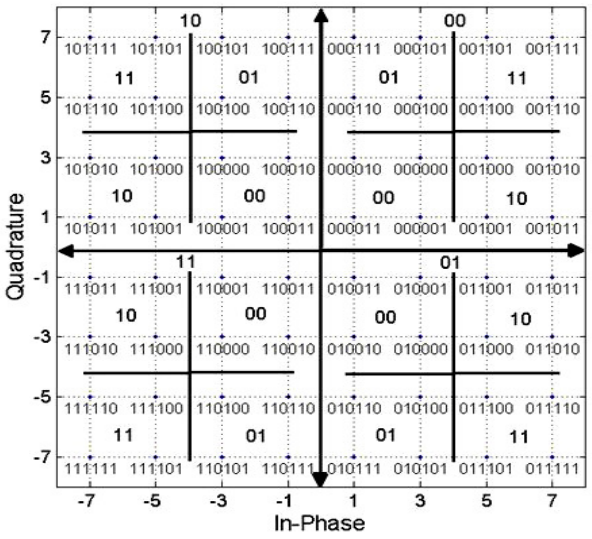


Fig. 2 64-QAM constellation with major and minor quadrants.

This can be explained by considering the four major and 16 minor quadrants of this constellation. The major quadrants are distinguished by the first two bits of the constellation point, for example, in the upper right major quadrant, the first two bits are 00. Hence, if the de-mapper only distinguishes between the four quadrants correctly, the first two bits are correctly de-mapped. Each major quadrant is divided into four minor quadrants, which are distinguished using the 3rd and 4th bits of the constellation points. Therefore, with bit ordering [15], the systematic bits S_0 receive the highest protection while the second parity bits, P_2, receive the lowest. Since the systematic bits of a Turbo encoder have the greatest impact on its performance, the re-ordering scheme improves the performance of the Turbo decoder. The modulated 64-QAM symbols are then transmitted over a complex Additive White Gaussian Noise (AWGN) channel and the corresponding received sequence is denoted by R_t.

The complete system model for the receiver is shown in Fig. 3. The received symbols R_t are fed to a soft-output 64-QAM de-mapper to produce soft bits. These soft bits are then de-multiplexed and sent for Turbo decoding. The first Turbo decoder is modified so that it can incorporate a-priori source statistics by combining the trellis of the Turbo decoder with the trellis of the RVLC decoder as described in [7] and [10]. This results into a composite trellis structure with which JSCD can be performed. With JSCD the computation of the branch transition probability is modified.

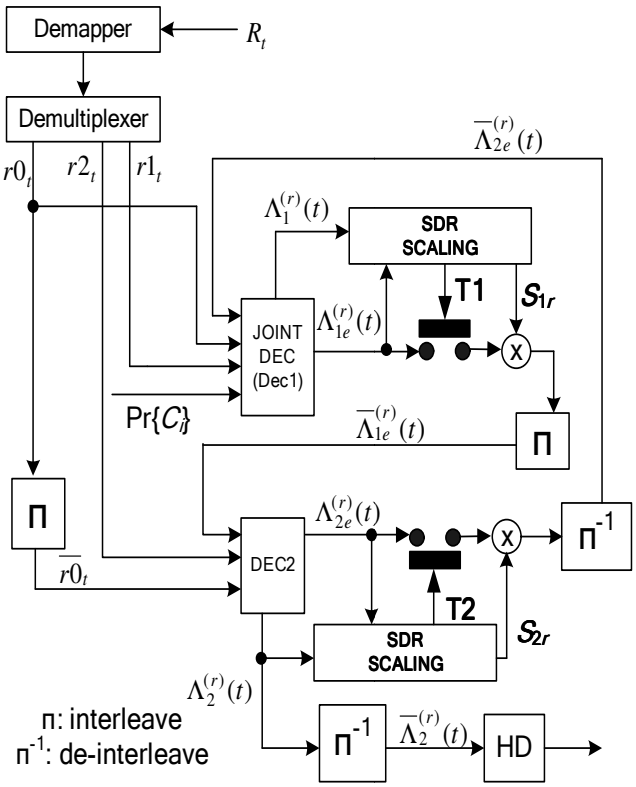


Fig. 3 Turbo decoding system with JSCD and SDR scaling.

Assuming that the Max-Log-MAP algorithm [17] is used, the branch metric probability for the joint decoder is computed as follows:

$$\overline{\gamma_t^{1(i)}}(l',l) = \log\left[p_t^1(i).\exp\left(-\frac{[r0_t - x0_t]^2 + [r1_t - x1_t]^2}{2\sigma^2} \right).\Pr\{C_i\} \right]$$
$$= \log\left[p_t^1(i)\right] - \left(\frac{[r0_t - x0_t]^2 + [r1_t - x1_t]^2}{2\sigma^2} \right) + \log\left[\Pr\{C_i\}\right] \quad (1)$$

where,

$\overline{\gamma_t^{1(i)}}(l',l)$ is the branch transition probability from state *l'* to *l* of bit *i* (*i* = 0 or 1) at time instant *t*,

$p_t^1(i)$ is the a-priori probability of bit *i* derived from the channel extrinsic information and input to the joint (first) decoder,

$\Pr\{C_i\}$ is the a-priori probability of bit *i* obtained from source statistics,

$r0_t$ and $r1_t$ are the de-mapped soft bits corresponding to the bipolar equivalent of the transmitted systematic bits, $x0_t$ and first parity bits, $x1_t$. σ^2 is the noise variance [10].

With the joint decoder, the a-priori statistics, $\Pr\{C_i\}$ can be incorporated into the Turbo decoding process. The derivation of the a-priori source statistics for the RVLC source given in Table I is now explained. The RVLC decoder's bit-level trellis is shown in Fig. 4 [10].

From the bit level trellis, the probability of the transition from state $M_{t-1} = l'$ to $M_t = l$ where l', $l \in (F,IA,IB,IC,ID,IE,IF)$, given an input bit *i* at time instant t, can be derived for all possible state transitions. For example, the probability of the transition from the final state F to the intermediate state IA, is given by [10]:

$$\Pr(M_t = IA, i = 0 \mid M_{t-1} = F) = PA + PB = P01 \quad (2)$$

For simplicity, the state transition probability for any state corresponding to bit *i* is denoted as $\Pr\{C_i\}$ and the joint decoder exploits this probability in computing the branch metric probability as per equation (1) [10]. The forward recursive variable, $\overline{\alpha_t^1}(l)$, at time *t* and state *l* is computed as follows for a joint decoder with M_j states:

$$\overline{\alpha_t^1}(l) = \max\left(\overline{\alpha}_{t-1}^1(l') + \overline{\gamma_t^{1(i)}}(l',l) \right) for \;\; 0 \le l' \le M_j - 1 \quad (3)$$

TABLE I. RVLC CODEWORDS

Symbol	Probabilty	RVLC [16]
A	0.33 (PA)	00
B	0.30 (PB)	01
C	0.18 (PC)	11
D	0.10 (PD)	1010
E	0.09 (PE)	10010

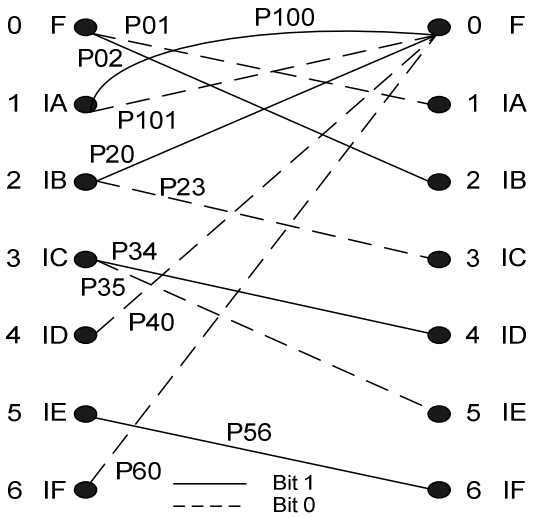


Fig. 4. Bit level trellis of RVLC decoder [10].

The number of states of the joint decoder, M_j is greater than the number of states, M_s of the second decoder (DEC2), because the joint decoder is obtained by merging the states of the RVLC decoder with the states of the Turbo decoder as described in [10]. The backward recursive variable, $\overline{\beta_t^1}(l)$, is computed as follows:

$$\overline{\beta_t^1}(l) = \max\left(\overline{\beta}_{t-1}^1(l') + \overline{\gamma_t^{1(i)}}(l,l') \right) for \;\; 0 \le l' \le M_j - 1 \quad (4)$$

The Log-Likelihood Ratio (LLR), $\Lambda_1^{(r)}(t)$ at iteration *r* and time *t* for the joint decoder is computed as follows:

$$\Lambda_1^{(r)}(t) = \max\left(\overline{\alpha}_{t-1}^1(l') + \overline{\gamma_t^{1(1)}}(l',l) + \overline{\beta}_t^1(l) \right) \quad (5)$$
$$- \max\left(\overline{\alpha}_{t-1}^1(l') + \overline{\gamma_t^{1(0)}}(l',l) + \overline{\beta}_t^1(l) \right) for \;\; 0 \le l' \le M_j - 1$$

The extrinsic information $\Lambda_{1e}^{(r)}(t)$ at iteration *r* and time *t* for the joint decoder is computed as follows:

$$\Lambda_{1e}^{(r)}(t) = \Lambda_1^{(r)}(t) - \frac{2}{\sigma^2} r0_t - \overline{\Lambda}_{2e}^{(r-1)}(t) \quad (6)$$

where, $\overline{\Lambda}_{2e}^{(r-1)}(t)$ is the de-interleaved extrinsic information obtained from the second decoder at iteration *r*-1.

The extrinsic information, $\Lambda_{1e}^{(r)}(t)$ and the LLR, $\Lambda_1^{(r)}(t)$ are then sent to a SDR scaling mechanism, which computes a scale factor S_{1r} as follows:

$$S_{1r} = \frac{1}{N} \sum_{t=1}^{N} f\left(\Lambda_{1e}^{(r)}(t), \Lambda_1^{(r)}(t) \right) \quad (7)$$

where, $f\left(\Lambda_{1e}^{(r)}(t),\Lambda_{1}^{(r)}(t)\right)$=1 if $\Lambda_{1e}^{(r)}(t)$ and $\Lambda_{1}^{(r)}(t)$ have the same sign, otherwise $f\left(\Lambda_{1e}^{(r)}(t),\Lambda_{1}^{(r)}(t)\right)$=0. *N* is the frame size in bits.

When S_{1r} takes its maximum value of 1.0, the switch T1 is opened, the iterative decoding process is stopped and a hard decision is made on $\Lambda_{1}^{(r)}(t)$. However, when $S1r$ is less than one, T1 remains closed and the extrinsic information $\Lambda_{1e}^{(r)}(t)$ is scaled with S_{1r} and interleaved to obtain $\overline{\Lambda}_{1e}^{(r)}(t)$. Hence, the SDR scaling mechanism acts both as a stopping criterion and a scale factor generator. The mechanism is derived from the one proposed in [13], but, in this work $\Lambda_{1e}^{(r)}(t)$ and $\Lambda_{1}^{(r)}(t)$ are used to compute the scale factor and not $\overline{\Lambda}_{2e}^{(r-1)}(t)$ and $\Lambda_{1e}^{(r)}(t)$. Another difference is that in this work only the extrinsic information has been scaled and not the soft channel inputs, as was the case in [13]. The a-priori probability, $p_t^2(i)$, is computed as follows and sent to decoder 2:

$$p_t^2(i)=\left\{\begin{array}{l}\dfrac{\exp(\overline{\Lambda}_{1e}^{(r)}(t))}{1+\exp(\overline{\Lambda}_{1e}^{(r)}(t))}\ for\ \ i=1\\ \dfrac{1}{1+\exp(\overline{\Lambda}_{1e}^{(r)}(t))}\ for\ \ i=0\end{array}\right\} \quad (7)$$

The branch metric probability for the second decoder is computed as follows:

$$\overline{\gamma_t^{2(i)}}(l',l\ )=\log\left[p_t^2(i)\right]-\left(\frac{[\overline{r0}_t-x0_t]^2+[r2_t-x2_t]^2}{2\sigma^2}\right) \quad (8)$$

where, $r2_t$ is the de-mapped soft bits corresponding to the bipolar version of the transmitted second parity bits $x2_t$ and $\overline{r0}_t$ is the interleaved counterpart of $r0_t$.

The forward and backward recursive variable, $\overline{\alpha_t^2}(l)$ and $\overline{\beta_t^2}(l)$ at time *t* and state *l* are computed as follows:

$$\overline{\alpha_t^2}(l)=\max\left(\overline{\alpha}_{t-1}^2(l')+\overline{\gamma_t^{2(i)}}(l',l)\right)for\ \ 0\le l'\le M_s-1 \quad (9)$$

$$\overline{\beta_t^2}(l)=\max\left(\overline{\beta}_{t-1}^2(l')+\overline{\gamma_t^{2(i)}}(l,l')\right)for\ \ 0\le l'\le M_s-1 \quad (10)$$

The LLR, $\Lambda_2^{(r)}(t)$ and extrinsic information, $\Lambda_{2e}^{(r)}(t)$ at iteration *r* and time *t* are computed as follows:

$$\Lambda_2^{(r)}(t)=\max\left(\overline{\alpha}_{t-1}^2(l')+\overline{\gamma_t^{2(1)}}(l',l)+\overline{\beta}_t^2(l)\right)$$
$$-\max\left(\overline{\alpha}_{t-1}^2(l')+\overline{\gamma_t^{2(0)}}(l',l)+\overline{\beta}_t^2(l)\right)for\ \ 0\le l'\le M_j-1 \quad (11)$$

$$\Lambda_{2e}^{(r)}(t)=\Lambda_2^{(r)}(t)-\frac{2}{\sigma^2}\overline{r0}_t-\overline{\Lambda}_{1e}^{(r)}(t) \quad (12)$$

The scale factor S_{2r} is computed as follows:

$$S_{2r}=\frac{1}{N}\sum_{t=1}^{N}f\left(\Lambda_{2e}^{(r)}(t),\Lambda_{2}^{(r)}(t)\right) \quad (13)$$

where, $f\left(\Lambda_{2e}^{(r)}(t),\Lambda_{2}^{(r)}(t)\right)$=1 if $\Lambda_{2e}^{(r)}(t)$ and $\Lambda_{2}^{(r)}(t)$ have the same sign. Finally, the a-priori probability, $p_t^1(i)$, is computed as per equation (7) but using $\overline{\Lambda}_{2e}^{(r)}(t)$. If S_{2r} = 1.0, T2 is opened to stop the iterative decoding process and a hard decision, (HD) is made on $\overline{\Lambda}_{2}^{(r)}(t)$.

The combination of prioritized constellation mapping, JSCD and adaptive scaling certainly lead to an enhanced Turbo coded 64-QAM system, but at the cost of greater computational complexity and delay. The complexity increase due to the bit re-ordering scheme is negligible and may even be integrated with the multiplexer. JSCD on the other hand leads to the greatest increase in complexity and delay because as mentioned previously the joint decoder is obtained by merging the states of the RVLC decoder with the states of the Turbo decoder. The number of computations involved in computing S_{1r} and S_{2r} to perform adaptive scaling also increase the delay. However, this is compensated by the faster convergence achieved with the use of the scale factor and the possibility of stopping the iterative decoding process once convergence is achieved. This prevents the decoder from performing unnecessary iterations.

III. SIMULATION RESULTS AND ANALYSIS

The performances of the following four Turbo coded 64-QAM schemes are compared:

Scheme 1 – The Turbo coded 64-QAM system with JSCD, adaptive scaling and prioritised constellation mapping. The encoding and decoding frameworks are given in Fig. 1 and Fig. 3, respectively.

Scheme 2 - This scheme only uses prioritised constellation mapping. The encoding is as per Fig. 1, but the decoding does not include JSCD or adaptive scaling.

Scheme 3 – This scheme uses JSCD and adaptive scaling and its decoder is similar to Scheme 1. However, prioritised constellation mapping is not performed, as such, the bit re-ordering block of the encoder shown in Fig. 1 is omitted.

 ISBN: 978-1-61208-263-9

Scheme 4 – This scheme is a conventional Turbo coded 64-QAM system without prioritised constellation mapping, JSCD and SDR scaling.

In all simulations, a random alphabet source with the probability distribution given in Table I has been used. After generating the alphabets, they are grouped into packets of size P = 64 symbols. The packets are then Reversible Variable Length Coded to obtain an RVLC bit-stream as shown in Fig. 1. Normally, the length in bits, L, of each packet is transmitted as side-information because L is different for each packet. The packetization is important to prevent error propagation. The RVLC bit-streams of all packets are grouped into blocks of 4056 bits since an interleaver size of 4056 bits has been used in the simulations. The parameters for the Turbo code used are as follows:

Generator: G = [1, g1/g2], where g0 = 7 and g1 = 5 in Octal.
Interleaver size, N = 4056 bits.
Maximum number of iterations, I = 12.
Code-rate = 1/3 and channel model: Complex AWGN.

The graphs of Bit Error Rate (BER) as a function of Eb/No have been plotted separately over a low Eb/No range: 0 dB ≤ Eb/No ≤ 3 dB and a high Eb/No range: 3.5 dB 0 ≤ Eb/No ≤ 6.5 dB in steps of 0.5 dB. Eb/No is the ratio of the bit energy, Eb to the noise power spectral density, No. It is to be noted that the transition from the low Eb/No range to the high Eb/No range is essentially a continuity from 3 dB to 3.5 dB and up to 6.5 dB. The performance analysis has also been made for both iterative and non-iterative decoding.

Fig. 5 shows the graph of BER against Eb/No for iterative decoding over the low Eb/No range. It is observed that the Turbo coded 64-QAM system with JSCD, adaptive scaling and prioritised constellation mapping (Scheme 1) provides the best performance with an average gain of 2.5 dB for BER > 10^{-1} over the conventional Turbo coded system (Scheme 4). At an Eb/No of 1 dB, Scheme 1 also provides a gain of about 1.5 dB over Scheme 3, which does not employ prioritised constellation mapping.

Moreover, Scheme 2, which uses only prioritised constellation mapping outperforms Scheme 3 by 1dB at an Eb/No = 1 dB. It is to be noted from a theoretical point of view the performance of the system for a BER > 10^{-1} is important because it is revealing a new characteristic of the system whereby it is seen that significant gains can be obtained in this BER region using the proposed technique. However, from a practical point of view the performance of the system for BER > 10^{-1} is not really relevant.

Fig. 6 shows the graph of BER against Eb/No for iterative decoding over the high Eb/No range. In this range it is observed that prioritised constellation mapping is not beneficial. For example, Scheme 2 provides the worst performance while the performance of Scheme 1 is comparable to that of Scheme 4. A possible explanation is that with iterative decoding, in the high Eb/No range, convergence takes place. As such, giving more protection to the systematic bits does not provide further gains. Also, since lower protection has been given to the parity bits, this can lead to performance degradation. Over this Eb/No range, Scheme 3 which uses only JSCD and adaptive scaling provides the best performance with an average gain of 1 dB in Eb/No over Scheme 1 and Scheme 4. It is to be noted that Scheme 3 outperforms Scheme 1 over this high Eb/No range because Scheme 1 suffers from a performance loss, which results from the use of prioritised constellation mapping after convergence.

Fig. 7 shows the graph of average number of iterations versus Eb/No over the range 3 dB ≤ Eb/No ≤ 6.5 dB. Scheme 1 and Scheme 3, which employ SDR based scaling with a stopping criterion, show a progressive decrease in the number of iterations required as the Eb/No increases. For example at an Eb/No of 5.5 dB, Scheme 3 consumes six iterations less than Scheme 2 and Scheme 4. However, Scheme 1 consumes on average 1.5 iterations more than Scheme 3 due to performance loss as a result of using prioritised mapping after convergence. The number of iterations required by Schemes 2 and 4 remains fixed at 12.

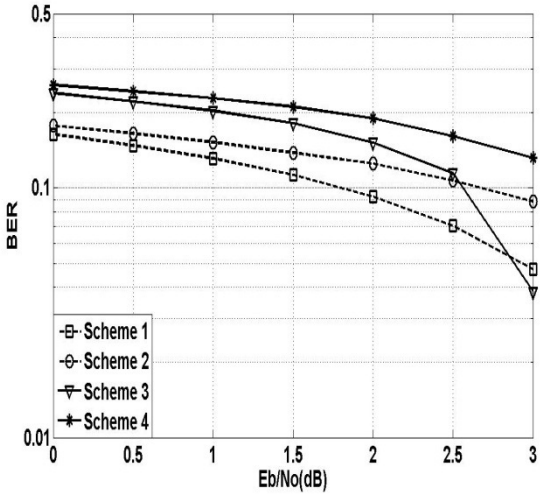


Fig. 5. Iterative low Eb/No performance with N = 4056.

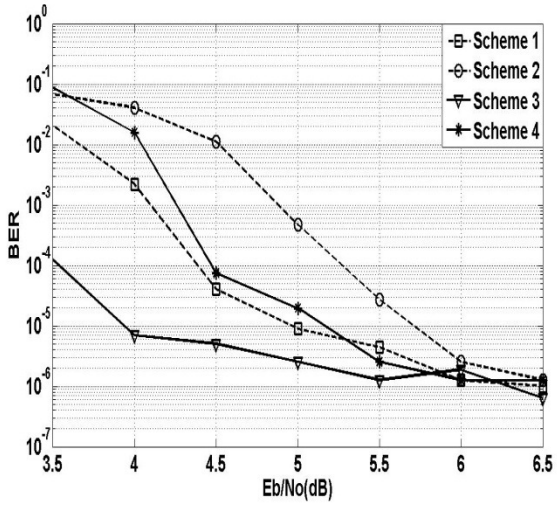


Fig. 6. Iterative high Eb/No performance with N = 4056.

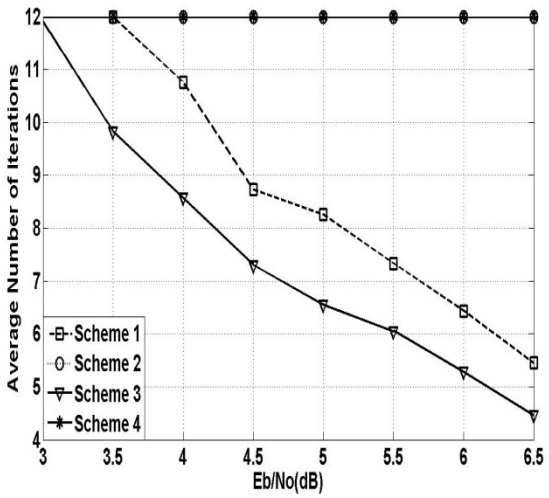


Fig. 7. Average number of iterations vs Eb/No for N = 4056 .

Fig. 8 shows the graph of BER against Eb/No for non-iterative decoding over the low Eb/No range. It is observed that Scheme 1 outperforms all the other schemes with an average gain of 2.5 dB over Scheme 4 and 2 dB over Scheme 3. However, with non-iterative decoding, convergence does not take place and the use of prioritized constellation mapping does not lead to degradation at BERs below 10^{-1}. This is observed in Fig. 9 whereby Scheme 1 outperforms Scheme 3 by 0.5 dB on average and Scheme 4 by almost 1.5 dB. It is to be noted that in [15], whereby only bit-reordering was used, it was also observed that with non-iterative decoding a performance gain is obtained throughout the Eb/No range whereas with iterative decoding convergence takes place at a certain point. Hence when prioritized constellation mapping is used the iterative scheme does not present a similar relation as the non-iterative.

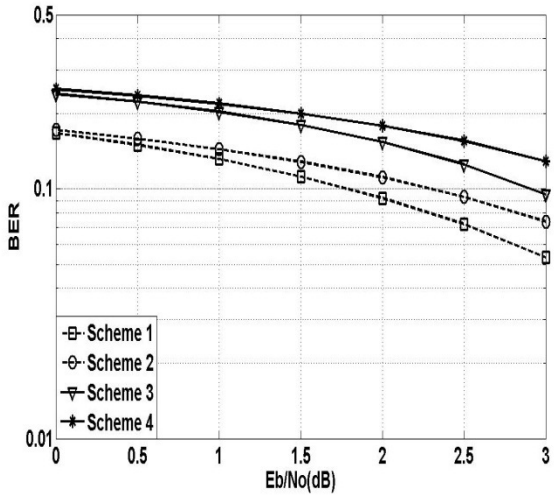


Fig. 8. Non-Iterative low Eb/No performance with N = 4056.

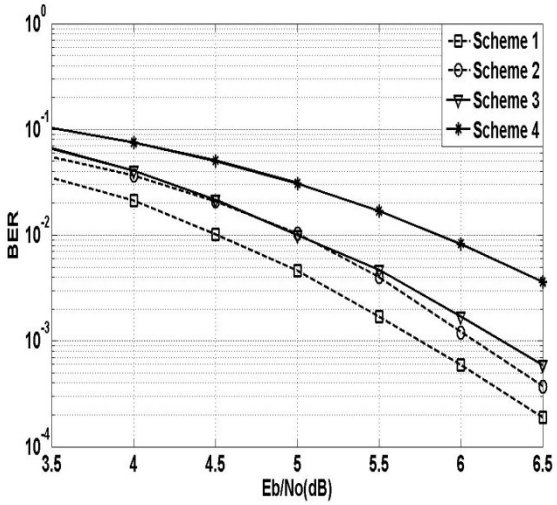


Fig. 9. Non-Iterative high Eb/No performance with N = 4056.

IV. CONCLUSION AND FUTURE WORK

This paper proposed an efficient Turbo coded 64-QAM scheme with JSCD, adaptive scaling and prioritised constellation mapping. At the encoder side a re-ordering mechanism is used to map the systematic bits of the Turbo encoder on the most strongly protected points of the 64-QAM constellation. To enhance the decoding process, JSCD is used to incorporate a-priori source statistics and adaptive SDR based scaling is also performed. At BERs above 10^{-1}, the proposed scheme provides a significant performance gain of 2.5 dB with iterative decoding over a conventional Turbo coded scheme. For BERs below 10^{-1}, the use of prioritised constellation mapping degrades performance as a result of convergence. Hence, for BERs below 10^{-1}, it is preferable to use only JSCD and SDR scaling, which achieves a gain of 1 dB on average over a conventional Turbo coded scheme. However, with non-iterative decoding, the proposed scheme, outperforms a conventional Turbo coded scheme at all BERs because there is no performance degradation due to prioritised constellation mapping. Overall, the combination of prioritised constellation mapping with JSCD and SDR based scaling appears promising for Turbo coded 64-QAM systems.

Several interesting future works can be envisaged from the scheme proposed in this work. A straightforward extension would be to assess its suitability for communication systems such as LTE. A more challenging future work would be to use JSCD schemes, which are less complex and hence do not incur significant delays while still allowing the exploitation of source statistics. The prioritised constellation mapping scheme could also be optimised so that performance gains could be obtained in the high Eb/No range also. Finally, investigations could be made on how to extend the scheme to block Turbo codes and also on the possibility of using bit interleaved coded modulation.

 ISBN: 978-1-61208-263-9

ACKNOWLEDGMENT

The authors would like to thank the University of Mauritius for providing the necessary facilities for conducting this research as well as the Tertiary Education Commission of Mauritius.

REFERENCES

[1] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes", IEEE International Conference on Communications, ICC 93. Geneva, vol. 2, 23-26 May 1993, pp.1064-1070.

[2] S. Sesia, I. Toufik and M. Baker, LTE – The UMTS Long Term Evolution: From Theory to Practice, John Wiley & Sons, Ltd, 2009, ISBN: 978-0-470-69716-0.

[3] 3GPP, "Technical Specifications Rel.8.", 2009.

[4] 3GPP2 C.S0024-B, "CDMA 2000 High Rate Packet Data Air Interface Specification" Version 1.0, May 2006. Available Online: http://www.3gpp2.org/Public_html/specs/C.S0024-B_v1.0_060522.pdf [Accessed: November 2012].

[5] J. Zyren, "Home Plug Green PHY Overview", Technical Paper, Atheros Communications, 2010.

[6] I. Lee, C.E.W. Sundberg, S. Choi and W. Lee, "A modified medium access control algorithm for systems with iterative decoding", IEEE Transactions on Wireless Communications, vol. 5(2), 2006, pp.270-273.

[7] A.H. Murad and T.E. Fuja, "Joint source‐channel decoding of variable length encoded sources", Proceedings of the Information Theory Workshop (ITW). Killarney, Ireland, June. 1998, pp. 94‐95.

[8] M. Jeanne, J.C. Carlach and P. Siohan, "Joint source‐channel decoding of variable length codes for convolutional codes and turbo codes", IEEE Trans Commun vol. 53(1), 2005, pp.10‐15.

[9] W. Xiang and P. Lu, "Bit-Based Joint Source-Channel Decoding of Huffman Encoded Markov Multiple Sources", *Journal of Networks*, vol. 5(4), 2010, pp. 443-450.

[10] T.P. Fowdur and K.M.S. Soyjaudah "Performance of joint source–channel decoding with iterative bit combining and detection", Ann. Telecommun. vol. 63, 2008, pp.409-423.

[11] J. Vogt and A. Finger, "Improving the MAX-Log-MAP Turbo decoder," Electr. Lett., vol. 36, no. 23, Nov. 2000, pp. 1937-1939.

[12] T. Gnanasekaran and K. Duraiswamy, "Performance of Unequal Error Protection Using MAP Algorithm and Modified MAP in AWGN and Fading Channel," Journal of Computer Science, vol. 4 (7) ,2008, pp. 585-590.

[13] Y. Lin, W. Hung, W. Lin, T. Chen, E. Lu, "An Efficient Soft-Input Scaling Scheme for Turbo Decoding," IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing Workshops, vol. 2, 2006, pp.252-255.

[14] Y. Wu, B. Woerner, and J. Ebel, "A Simple Stopping Criterion for Turbo Decoding," *IEEE Commun. Lett*., vol. 4, no. 8, Aug. 2000, pp. 258-260.

[15] H. Lüders, A. Minwegen, and P. Vary, "Improving UMTS LTE Performance by UEP in High Order Modulation", 7th International Workshop on Multi-Carrier Systems & Solutions (MC-SS 2009), Herrsching, Germany, 2009, pp.185-194.

[16] Y. Takishima, M. Wada, and H. Murakami, "Reversible variable length codes," *IEEE Trans. on Commun.*, vol. 43, 1995, pp.158–162.

[17] B. Vucetic and J. Yuan, Turbo Codes: Principles and Applications, Kluwer Academic Publishers, 2000.

 ISBN: 978-1-61208-263-9

A Vehicle Ad-Hoc Network for Traffic Information System

Jaehong Ryu, Byeongcheol Choi
Lab. of Convergence Technology
Electronics and Telecommunications
Research Institute, Daejeon, Korea
{jhryu, bcchoi}@etri.re.kr

Dongwon Kim
Dept. of Electronics Information
Chungbuk Provincial University,
Chungbuk, Korea
Corresponding author:won@cpu.ac.kr

Mihee Yoon
Dept. of Computer Information
Chungbuk Provincial University,
Chungbuk, Korea
mihee@cpu.ac.kr

***Abstract*— We propose a vehicle ad-hoc network for traffic information service. It allows collecting and analyzing traffic status of large areas without any infrastructure, e.g., probe cars, road side unit, and server. The proposed scheme uses multi-channel and broadcasting technique. Vehicle terminal simply needs the existing navigation systems in vehicles and wireless communication devices for vehicle-to-vehicle communication. Communication and networking algorithm is given and experimented on the testbed. It effectively collects accurate traffic information, and is able to provide real-time traffic information propagation by using only vehicle-to-vehicle(V2V) communications without infrastructure.**

Keywords-VANET; Traffic Information Service; Driver Assistance; Navigation.

I. INTRODUCTION

Recently, many researchers have been studying on the active vehicle-safety services based on inter-vehicle communication to improve drivers' safety [1]. This safety service aims at collision warning, collision avoidance, providing traffic information, etc. For instance, G. Held [1] discussed the traffic information service that assists drivers with the information of driving paths detouring accident zones or traffic jams in real-time manner.

The traffic information service roughly divides into Intelligent Road and Vehicle Ad-hoc Network (VANET) in terms of inter-vehicle communication [2-4]. In Intelligent Road, the traffic information is wirelessly collected from vehicles to access points which are deployed along the roads, sent to the information center, processed, and then broadcasted to drivers on the roads. An example of this scheme is Traffic and Travel Information via Transport Protocol Expert Group (TPEG) service. However, it has several disadvantages, e.g., it costs a lot to incorporate facilities for the service along the roads. In addition, even though a number of information providers including taxis and buses participate in TPEG, real-time service is not easy to offer due to the time delay of more than 5 minutes for updating the information. Moreover, the traffic information provided by the agencies is not free of charge. In VANET, on the other hand, the traffic information is delivered directly from vehicle to vehicle. It can be delivered to the vehicle following in the same lane or to the vehicles, so called messengers, running in the opposite lane. However, VANET typically relies on broadcasting which may cause the channel collision that degrades communication efficiency or generate too many messages overloading the systems.

In this paper, we propose a traffic information service based on vehicle ad-hoc network. The proposed scheme collects, processes, and distributes traffic information without establishing a specific infrastructure, e.g., server or separated monitoring systems. We design communication and networking algorithm. It utilizes three wireless channels. It only allows the last vehicle in a traffic jam to communicate with the vehicles in the opposite lane. This multichannel communication scheme minimizes the probability of message collision and simultaneously enables accurate and reliable data delivery.

In Section II, a concept of traffic information service for traffic congestion avoidance is proposed. In Section III, method of channel operation for efficient use of resources is presented and Finally, in Section IV, conclusion of this study and description of future work are provided.

II. TRAFFIC INFORMATION SERVICE CONCEPT

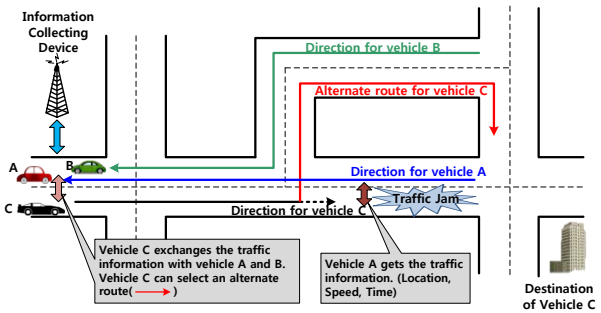


Figure 1. Traffic Information Service Concept

Figure 1 shows how the vehicle C avoids the traffic jam and arrives to the destination. Vehicle A, B, and C generate the traffic information based on location data supported by Global Positioning System (GPS). The collecting devices transfer and retrieve various information including vehicle speeds, traffic environment, etc. through the wireless network. The destination is where the vehicle C goes toward and the traffic jam is where the heavy traffic occurs.

 ISBN: 978-1-61208-263-9

The service scenario is as follows: (1) Vehicle C is on the way to the destination; (2) Vehicle A has been moving in "the direction of vehicle A" as shown in Figure 1. Since vehicle A went through the jammed area, it has obtained the traffic information, e.g., speed change of vehicles over the passed locations including the jammed area; (3) Vehicle B has been moving in "the direction of vehicle B" as shown in Figure 1. The traffic information that vehicle B has obtained is likely to be normal since it did not pass through the jammed area; (4) Vehicle A, B, and C regularly transmit the traffic information on their respective paths; (5) If information collecting devices are incorporated on the road, they can serve to retrieve, process, and regularly transmit the traffic information to users; (6) Based on the delivered information from vehicle A and B, vehicle C can be aware of the traffic jam on the path of vehicle A in advance; and thus, (7) Vehicle C can choose the right path avoiding the heavy traffic.

In summary, when vehicle C does not have a priori knowledge on the traffic ahead, it has possibility to meet the jammed area before it arrives to the destination. On the opposite lane, vehicle A has come from the jammed area and vehicle B has come without suffering any traffic jam. Since vehicle C receives traffic information from vehicle A and B on two possible paths in advance, it can choose its preferred path to the destination avoiding traffic jam.

III. SERVICE METHODS

A. Channel Operation Scheme

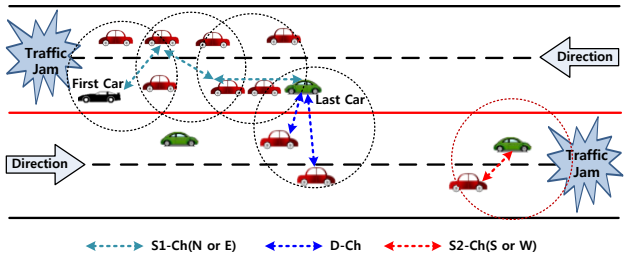


Figure 2. An Example of Wireless Communication Channels

Figure 2 shows the wireless communication channels of vehicles running on roads near traffic jammed areas. The vehicle's terminal operates with three wireless communication chanels. The first channel (S1-Ch) is for communicating with the vehicles driving in north or east directions. The second channel (S2-Ch) is for communicating with vehicles driving in south or west directions. The third channel, called D-Ch, is for communicating with the vehicles driving in the opposite direction.

B. Message Format

The header and trailer are determined by the adopted PHY/MAC protocol. Note that all messages are broadcasted without any join and association procedure between vehicles.

We define four types of messages that are classified by the value in Msg_Type field as follows.

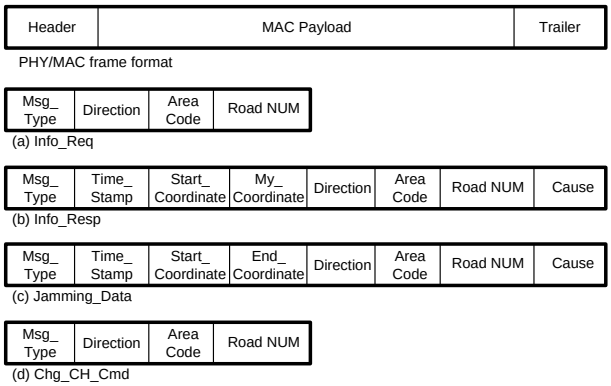
① Info_Req indicates that the message is a request for the vehicle ahead to send the traffic information through S-Ch.

② Info_Resp indicates that the message contains traffic information and is sent to the rear car through S-Ch.

③ Jamming _Data indicates that the message contains traffic information about jammed area and is sent to the cars on the other lane through D-Ch.

④Chg_CH_Cmd indicates that the message is a request for the last car to change the communication channel from D-Ch to S-Ch. This message is issued by a newly joining car to collect traffic information when it overhears the messages about jammed areas that the last car is sending through D-Ch.

These messages assist driving vehicles with spatiotemporal traffic information contained in various message fields. For example, Start_Coordinate, Time_Stamp, Cause fields contain starting points, starting time, cause of traffic jam, respectively. The length and route of traffic jam can be derived from Start_Coordinate with End_Coordinate and Area_Code with Road_NUM, respectively.



Msg_Type (1byte): message type
Time_Stamp (2bytes): current time(ddhhmmss)
Start_Coordinate (8bytes): latitude/longitude coordinate of the first car
End_Coordinate (8bytes): latitude/longitude coordinate of the last car
Direction (1byte): East->0, West->1, South->2, North->3
Area_Code (2bytes): area code managed by nation
Road_NUM (2bytes): road number managed by nation
Cause (1byte): cause of the traffic jam

Figure 3. Frame and Message Format

C. Communication & Networking Algorithm

Figure 4 shows the vehicle states, channel operations, communication algorithm in several situations. The five states of vehicles are defined as follows.

- NOR is the state of the normal car that drives faster than its predetermined speed.
- FST is the state of the first car at the starting point of jammed area, which generates the traffic information including location, time, and cause of the traffic jam.
- LST is the state of the last car at the last point of jammed area, which delivers traffic information to the vehicles on the opposite lane.

 ISBN: 978-1-61208-263-9

- FWD is the state of the vehicles in the middle of jammed area, which forwards information in Broadcast Mitigation Technique(BSMT) procedure for a multi-hop broadcast-based communication. Since FWD is temporary, it eventually switches into LST or MDL.

- MDL is the state of the vehicles in the middle of jammed area. In BSMT procedure, the vehicles that are not selected to be on FWD become MDL.

In Figure 4, if a vehicle on NOR drives slower than the predetermined speed owing to traffic jam, it checks on a Jamming_data message reception to see whether there has already been a vehicle on LST.

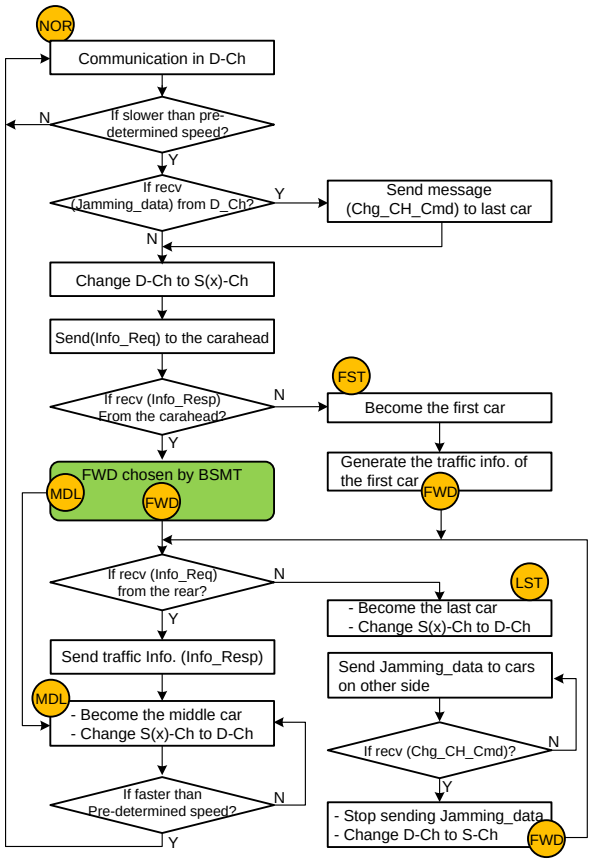


Figure 4. The Channel-Operating and Networking Algorithm

If it receives Jamming_Data through D-Ch, it sends Chg_CH_Cmd message to the car ahead on LST for requesting channel change from D-Ch to S(x)-Ch, where x is 1 or 2. Then, it also changes its own channel to S(x)-Ch and transmits Info_Req message requesting traffic information ahead, and waits for Info_Resp.

On the other hand, if it receives no Jamming_data message, it changes its channel to S(x)-Ch and transmits Info_Req message requesting traffic information ahead, and waits for Info_Resp.

If it does not receive Info_Resp for the time interval, T_d, defined by (3), it goes into FST and starts generating traffic information including its GPS location, cause of the traffic jam. And then it starts operating as the car on FWD. If it does receive Info_Resp, BSMT procedure determines whether it goes into MDL or FWD.

If it is determined to be on MDL, it changes its channel to D-Ch. On the other hand, if it is determined to be on FWD, it checks whether it receives Info_Req. If it does, it sends Info_Resp containing traffic information coming from the vehicle on FST to the vehicle in the rear and it changes its state into MDL

However, if it does not receive Info_Req, it goes into LST, changes its channel to D-Ch, and start to periodically send Jamming_Data built with traffic information from itself and the vehicle on FST to the car on the opposite lane. The transmission period is given by (10) below.

When the car on LST receives Chg_CH_Cmd while periodically sending Jamming_data, it stops sending Jamming_data, changes its channel to S(x)-Ch, and starts operating as the car on FWD.

And if all vehicles drive faster than the predetermined speed, they go into NOR.

D. Broadcasting by Broadcast Storm Mitigation Technique

The multi-channels as with S(x)-Ch and D-Ch in this paper, generally show higher performance than the single-channel since it can mitigate collision and interference. In our scheme, the traffic information from the first car should be broadcasted to the cars in the rear through S(x)-Ch or sometimes it should be propagated to the cars in the middle of traffic jam through multi-hop communication. Note that the broadcast-based scheme may cause serious problems, e.g., broadcast storm, which results in severe performance degradation. To address this problem, we utilize Slotted 1-Persistence Broadcasting Rule [5]. We briefly introduce the rule for understanding.

Upon receiving a packet, a node checks the packet ID and rebroadcasts with the pre-determined probability 1 at the assigned time slot TS_{ij}, as expressed by Eq. 1, if it receives the packet for the first time and has not received any duplicates before its assigned time slot. Given the relative distance between nodes i and j, D_{ij}, the average radio range of a wireless link, R, and the predetermined number of slots N_S, TS_{ij} can be calculated as

$$TS_{ij} = S_{ij} \cdot \tau \qquad (1)$$

where τ is the estimated one-hop delay, which includes the medium access delay and propagation delay, and S_{ij} is the assigned slot number, which can be expressed as

$$S_{ij} = N_s \left\{ 1 - \left[\frac{min(D_{ij}, R)}{R} \right] \right\} \qquad (2)$$

In BSTM procedure, each car becomes on MDL if it receives Info_Resp within its own waiting time derived from (1). Otherwise, it becomes on FWD and retransmits Info_Resp. After retransmitting, if it receives Info_Resp from the rear within T_d, it goes into MDL since it is a forwarder in the middle of the multi-hop message path. Otherwise, it goes into FWD since it is the last forwarder in the multi-hop message path.

 ISBN: 978-1-61208-263-9

Here, the waiting time for which the car should wait to determine if it is the last on FWD after sending Info_Resp is as follows.

$$T_d = N_s \cdot \tau \quad (3)$$

E. Broadcast Rate Control based on the car speed

In this section, we consider the broadcast rate (B_R) of traffic information through D-Ch. If the broadcast rate increases, the probability of message collision increases due to increasing interference and traffic load within radio range. However, if the broadcast rate decreases, it becomes increasingly possible that the cars driving fast may have no chance to communicate each other. Thus, the broadcast rate should be carefully determined by vehicle speed, radio range, message length, transmission speed of media.

For example, if the vehicle speed is high, the broadcast rate, B_R, of each vehicle should be high because the vehicle density in coverage range probabilistically decreases. On the other hand, if the vehicle speed is low, the broadcast rate of each vehicle should be low because the vehicle density in coverage range probabilistically increases. In other words, it is desirable to keep the network within a stable range of the total network load by adjusting the broadcast rate. Additionally, when the vehicle speed becomes lower, the cars stay longer in communication range and thus have the increasing communication success rate even with low broadcast rate. On the other hand, if the vehicle speed is higher, the cars stay shorter in communication range and the high broadcast rate is desirable to increase the communication success rate.

Therefore, B_R of D-Ch is determined by (10) under assumption that the maximum ($\max S_v$) and minimum of the vehicle speed is 200km/h and 0km/h, respectively.

S denotes the transmission speed of a wireless link. L_f means the length of frame. The transmission time of one frame, T_p, is as follows.

$$T_p = \frac{L_f}{S} \quad (4)$$

The time interval, time measure of two cars on the opposite lanes staying in the successful communication range, is given as follows.

$$T_h = \frac{R}{2 \cdot S_v} \quad (5)$$

The maximum number of transmittable packets during T_h is determined as (6).

$$N_B = \frac{T_h}{T_p} \quad (6)$$

The maximum transmission rate, $\lambda_{\max}$, is the maximum number of transmittable packets per second as (7).

$$\lambda_{\max} = \frac{1}{T_p} = \frac{S}{L_f} = \frac{N_B}{T_h} \quad (7)$$

Given λ as the sum of all broadcast rates within a communication range, the offered load, ρ, is as follows.

$$\rho = \lambda \cdot T_p \quad (8)$$

$$\rho_{\max} = \lambda_{\max} \cdot T_p \quad (9)$$

In A. Tanenbaum [6], CSMA-based wireless networks show high performance under low offered load, where $\rho < 0.1\ \rho_{max}$. Thus, B_R of D-Ch is given as (10).

$$B_R = 0.09\lambda_{\max} \cdot \frac{S_v}{\max S_v} + 0.01\lambda_{\max} \quad (10)$$

F. Experimental System and Test

Figure 5 shows our experimental setup comprised of a main module and a communication module for vehicle-to-vehicle communication. The main module adopts S5PX100 with ARM Cortex-A8 Core. The peripheral includes a TFT LCD, a General Camera Interface, and a GPS. It also includes USB host/client, IEEE802.11a/b/g, UART for interfacing with external devices.

Android is used as the operating systems for the main module. Several functions are established on the main module including user interface, navigation/GPS showing the jammed area by the proposed algorithm, message transmitting and receiving blocks.

We use ATmega1281 MCU and CC2520 supporting IEEE802.15.4 for the communication module. It also supports UART communication to cooperate with the main module. The proposed vehicle-to-vehicle communication is based on IEEE802.15.4 but modified to rely on broadcast while excluding Join and Association procedures.

Figure 5. Experimental Setup

We conducted experiments where S and L_f are set to 250kbps and 54bytes, respectively. R is approximately 200m. Considering GPS accuracy, N_s is set to 5. The car speed, S_v, was selected randomly from 30 to 70 km/h at the start coordinate of latitude 36.385216 and longitude 127.359576. We monitored communication messages by using cc2420EB packet sniffer.

Our monitoring results showed that the vehicle speed of 50km/h causes no packet error and no packet loss although it slightly affects Link Quality Indicator calculated by RSS(LQI). As the vehicle speed becomes higher, the packet error and the packet loss occur more frequently because of wireless signal fading, which is a unique feature of

 ISBN: 978-1-61208-263-9

IEEE802.15.4. Therefore, IEEE802.11p is considerable for communication between vehicles driving at high speed.

IV. CONCLUSIONS

We proposed a traffic information service simply relying on the existing Navigation/GPS systems in vehicles and wireless communication devices for vehicle-to-vehicle communication, rather than on a separately established server. The proposed scheme collects traffic information over inter-vehicle networks, processes it to minimize the size, and transmits it to the destinations. This scheme uses three wireless communication channels and only a single selected last-vehicle is allowed to transmit the traffic information to the opposite lanes, which reduces the probability of wireless communication collision and mitigates the broadcast storm. Compared to the existing TPEG service, it has more advantages that it provides traffic information in timely manner and it can offer no charge service as well. We tested using IEEE802.15.4 based Wireless Sensor Network (WSN) platform. For future work, we will apply the algorithm to IEEE802.11p based Wireless Access in Vehicular Environment(WAVE) platform.

ACKNOWLEDGMENT

This work was partly supported by the IT R&D program of MKE/KEIT [10040027, Development of the International Standard-Based IVI System Commercialization Technology Using Open Source for Enhancing Car Infotainment Research Project]. and [10035380, Development of Low Power Consumption Sensor Network].

REFERENCES

[1] G. Held, “Inter- and Intra-Vehicle Communications”, Boca Raton: Auerbach Publications, 2008

[2] P. Bures, “The architecture of traffic and travel information system based on protocol TPEG”, Proceedings of the 2009 Euro American Conference on Telematics and Information Systems: New Opportunities to Increase Digital Citizenship, EATIS '09 , art. no. 1551743

[3] J. Misener and S. Shladover, “PATH Investigations in Vehicle-Roadside Cooperation and Safety: A Foundation for Safety and Vehicle-Infrastructure Integration Research”, Proceedings of IEEE Intelligent Transportation Systems, pp. 9 - 16, 2006

[4] D. Reichardt, et al, “CarTALK 2000: safe and comfortable driving based upon inter-vehicle-communication”, IEEE Intelligent Vehicle Symposium, vol. 2, pp. 545 - 550, 2002

[5] N. Wisitpongphan, O. Tonguz, J. Parikh, P. Mudalige, F. Bai, and V. Sadekar, “Broadcast Storm Mitigation Techniques in Vehicular Ad Hoc Networks”, IEEE Wireless Communications, Dec. 2007, pp. 84-94.

[6] A. Tanenbaum, “Computer Networks 3rd Edition”, Prentice Hall, Mar, 1996

 ISBN: 978-1-61208-263-9

Highlights on a Multiobjective Routing Method for Multiservice MPLS Networks with Traffic Splitting

Rita Girão-Silva, José Craveirinha
DEEC-FCTUC, INESC-Coimbra
Pólo II, Coimbra, Portugal
Email: {rita,jcrav}@deec.uc.pt

João Clímaco
INESC-Coimbra
Coimbra, Portugal
Email: jclimaco@inescc.pt

M. Eugénia Captivo
FCUL, CIO-UL
Lisboa, Portugal
Email: mecaptivo@fc.ul.pt

***Abstract*—A multiobjective routing model for Multiprotocol Label Switching networks with multiple service classes and considering traffic splitting is presented. The routing problem is formulated as a multiobjective mixed-integer program, and an exact resolution method based on the classical constraint method is outlined. Some experimental results on network performance measures, resulting from the application of the routing method in a reference test network, are presented. These results confirm the potential advantages of using a multiobjective optimization model in this routing problem, as we get a compromise solution that tries to balance the two considered objectives.**

***Keywords*—Routing models; Multiobjective optimization; Telecommunication networks; Network flow approach; Traffic splitting.**

I. INTRODUCTION

The routing calculation and optimization problems in modern multiservice networks are quite challenging, as the performance and cost metrics in these networks are multidimensional and often conflicting. There are potential advantages in formulating routing problems in these types of networks as multiple objective optimization problems, because the trade-offs among distinct performance metrics and other network cost function(s) (potentially conflicting) can be analyzed in a consistent manner. In multiobjective optimization problems, see e.g. [1], one seeks to find non-dominated solutions (or Pareto solutions), i.e., feasible solutions such that it is not possible to improve the value of an objective function without worsening the value of at least one of the other objective functions.

In a Multiprotocol Label Switching (MPLS) network, packets are forwarded through Label Switched Paths (LSP). An important problem in traffic engineering is to distribute the traffic trunks, i.e., the aggregation of traffic flows of the same Forwarding Equivalence Class (FEC) on the network by the possible LSPs. This procedure is known as traffic splitting [2], as the traffic trunks are split and mapped onto different paths in the network, satisfying the constraints of the bandwidth required by the traffic trunk of a given service class. This procedure is useful to obtain a balanced distribution of the load in the network and/or a reduction in the routing costs, but it entails the establishment of more LSPs and an increase in the complexity of the network management.

We can mention other works concerned with load balancing. A multiobjective problem formulated in the context of off-line routing in telecommunication networks is presented in [3]. In [4], it is shown that when multimedia traffic flows characterized as batch Markovian arrival processes, are split, the network performance (measured in terms of end-to-end delay, delay variance and cell loss probability) tends to improve. A survey on several multipath routing techniques in the Internet is presented in [5]. According to A. Dixit et al. [6], a fine grained traffic splitting technique used in data center networks leads to a better load-balanced network, when compared to techniques using equal-cost multipath routing.

In our work, a global routing problem i.e. involving the simultaneous calculation of the LSPs for all node-to-node traffic flows is considered. In this type of network-wide optimization approach, the objective functions of the route optimization model depend explicitly on all traffic flows in the network, see [7], [8]. Earlier works focused on routing optimization with traffic splitting are [7], [9]. A bi-objective lexicographic routing problem is formulated in [7]. The objectives are the maximization of the Quality of Service (QoS) traffic revenue and of the Best Effort (BE) traffic revenue. The resolution method is a lexicographic optimization method. At first, only the QoS traffic is considered; afterwards, the BE flows are taken into account, considering only the remaining available bandwidth. A model with three objectives (including the minimization of traffic splitting) is proposed in [9]. The bi-objective routing problem includes a constraint on the total number of paths used in the network. The resolution method is based on a lexicographic weighted Chebyshev metric method. A review on multiobjective routing models can be seen in [10].

This short paper presents an overview on current work on a multiobjective routing model for MPLS with traffic splitting. We consider a mixed-integer programming (MIP) formulation of the routing optimization model considering two objective functions (global routing cost and load cost) and a constraint on the maximal number of LSPs per flow, as suggested in [9]. The major contribution of the research work concerns: the extension of the aforementioned model to a multiservice case; the development of an exact resolution method for the calculation of non-dominated solutions, with special features related to the nature of the model; an experimental study for evaluation of the results of the method in terms of network performance measures, using a reference test network.

In this paper we describe the addressed model and its MIP formulation in Section II, and outline an exact resolution method (MCC) based on the classical constraint method [11] in Section III. In Section IV, some results with a reference test

 ISBN: 978-1-61208-263-9

network are analyzed. Finally, some conclusions are drawn.

II. Model Description

A network $(\mathcal{N}, \mathcal{A})$ with unidirectional arcs (or links) is considered, where $\mathcal{N}$ is the set of nodes in the network and $\mathcal{A}$ is the set of links in the network. The capacity of each network link k is given by u_k [Mbit/s], $k \in \mathcal{A}$.

Let $\mathcal{S}$ be the set of services of the network. Considering that the point-to-point offered bandwidth matrix T_{ij} [Mbit/s], $i, j \in \mathcal{N}$, and the percentage of bandwidth associated with each service ($q_s, s \in \mathcal{S}$, with $\sum_{s \in \mathcal{S}} q_s = 1.0$) are given, the value of the bandwidth offered by each flow $t \equiv (i, j, s)$ (corresponding to the traffic from service $s \in \mathcal{S}$ originating in node i and destined to node j) is $d_t = q_s T_{ij}$. The set of all network flows is $\mathcal{T}$. Let $\mathcal{P}_t = \{p_t^0, p_t^1, \cdots, p_t^{L_t-1}\}$ be the set of L_t feasible paths for flow t.

For each link $k \in \mathcal{A}$, an additive cost per unit of bandwidth, c_k, is considered; C_t^l is the cost of using path p_t^l, the l-th feasible path for flow t; the decision variable x_t^l is the part of the bandwidth offered by flow t which will be carried in the l-th path, hence specifying the traffic splitting solution.

The first objective function is the minimization of the total cost of carrying the bandwidth of all the flows offered to all the feasible paths:

$$\min F_1 = \sum_{t \in \mathcal{T}} \sum_{l=0}^{L_t-1} C_t^l x_t^l \tag{1}$$

with

$$C_t^l = \sum_{k \in p_t^l} c_k, \quad \forall l = 0, \cdots, L_t - 1, t \in \mathcal{T} \tag{2}$$

$$\sum_{l=0}^{L_t-1} x_t^l = d_t, \quad \forall t \in \mathcal{T} \tag{3}$$

$$x_t^l \geq 0, \quad \forall l = 0, \cdots, L_t - 1, t \in \mathcal{T} \tag{4}$$

where the constraint (3) guarantees that the total bandwidth required by flow t is carried by the assigned LSPs.

The second objective function is the minimization of the load cost in all the network links. In this way, a more balanced distribution of load in the network may be accomplished, so as to maximize the possibility of the network accepting more traffic requests in the future [12]. A piece-wise linear cost function ϕ_k is defined for each link $k \in \mathcal{A}$ (see (6)-(11)) as in [13], based on its utilization rate $\frac{f_k}{u_k}$, where f_k is the total load carried in the link. Hence, the second objective function is:

$$\min F_2 = \sum_{k \in \mathcal{A}} \phi_k \tag{5}$$

with

$$\phi_k \geq f_k, \quad \forall k \in \mathcal{A} \tag{6}$$

$$\phi_k \geq 2f_k - 0.5u_k, \quad \forall k \in \mathcal{A} \tag{7}$$

$$\phi_k \geq 5f_k - 2.3u_k, \quad \forall k \in \mathcal{A} \tag{8}$$

$$\phi_k \geq 15f_k - 9.3u_k, \quad \forall k \in \mathcal{A} \tag{9}$$

$$\phi_k \geq 60f_k - 45.3u_k, \quad \forall k \in \mathcal{A} \tag{10}$$

$$\phi_k \geq 300f_k - 261.3u_k, \quad \forall k \in \mathcal{A} \tag{11}$$

$$f_k \leq u_k, \quad \forall k \in \mathcal{A} \tag{12}$$

$$f_k = \sum_{t \in \mathcal{T}} \sum_{l=0}^{L_t-1} a_{t,l}^k x_t^l, \quad \forall k \in \mathcal{A} \tag{13}$$

where (12) guarantees that the link capacity is not exceeded. The parameter $a_{t,l}^k$ is binary and specifies whether a link k belongs to path p_t^l, i.e., $a_{t,l}^k = 1$ iff $k \in p_t^l, k \in \mathcal{A}, l = 0, \cdots, L_t - 1, t \in \mathcal{T}$ and $a_{t,l}^k = 0$ otherwise.

A third objective function minimizing the number of used paths for each flow can also be considered. If the number of used paths per flow increases, then the network routing control and management may become increasingly costly and complex because the signaling and processing tasks increase. Let y_t^l be the binary variable representing whether the path p_t^l is used, i.e., $y_t^l = 1$ iff the l-th path $p_t^l, l = 0, \cdots, L_t - 1$ is used by flow $t \in \mathcal{T}$ and $y_t^l = 0$ otherwise. Therefore, the third objective function is

$$\min F_3 = \max_{t \in \mathcal{T}} \left\{ \sum_{l=0}^{L_t-1} y_t^l \right\} \tag{14}$$

with

$$x_t^l \leq d_t y_t^l, \quad \forall l = 0, \cdots, L_t - 1, t \in \mathcal{T} \tag{15}$$

$$y_t^l \in \{0; 1\}, \quad \forall l = 0, \cdots, L_t - 1, t \in \mathcal{T} \tag{16}$$

A constraint on the maximal number of links D_s for the paths associated with a service $s \in \mathcal{S}$ is also considered: for flows with QoS requirements in real-time, e.g., voice and video services, D_s is the network diameter (maximal number of links of the shortest paths for all the network pairs of nodes); for flows of QoS services without real-time requirements, e.g., *Premium* data services, D_s is the network diameter + 1; for BE service flows, e.g., plain data services, no technical limits on the maximal number of links are imposed, so $D_s = |\mathcal{N}| - 1$.

The multiobjective routing problem may be formulated as

$$\min\{F_1, F_2, F_3\} \tag{17}$$

$$\text{subject to:} \quad \text{(3)-(4), (6)-(13), (15)-(16)} \tag{18}$$

$$\text{constraint on } D_s, \forall s \in \mathcal{S} \tag{19}$$

An important change to this main problem is that the third objective function F_3 will no longer be an objective and will rather be included in the constraints. The number of used paths per flow should be limited in practice for technical reasons to prevent excessive overheads related to control and signaling costs. Let $N_L \in \mathbb{N}$ be the maximal value allowed for the total

number of paths used by any traffic flow. The new problem P_0 to be addressed will be

$$\min\{F_1, F_2\} \tag{20}$$

$$\text{subject to:} \quad \sum_{l=0}^{L_t-1} y_t^l \leq N_L, \forall t \in \mathcal{T} \tag{21}$$

$$(18)\text{-}(19) \tag{22}$$

The total number L_t of feasible paths for each flow t can now be written as $L_t = \min\{N_L, N_t\}$. The maximal number of paths in the network for flow t, N_t, satisfies a constraint on the maximal number of links $D_s, s \in \mathcal{S}$. This constraint is usually defined for technical reasons, associated with transmission or traffic engineering and signaling requirements related to service type. For the generation of the set $\mathcal{P}_t$ for each flow t, the K-shortest path MPS algorithm [14] was used.

In [9], $c_k = 1, \forall k \in \mathcal{A}$. We have chosen to consider $c_k = \frac{\alpha}{u_k} + \beta l_k$ with $\alpha, \beta > 0$ and l_k [km] representing the length of the link. The first term reflects the economy of scale and the decrease in transmission times associated with increased capacity. The second term is related to propagation delays, which increase with the physical length of the link.

III. Resolution Method

For solving P_0 we developed an algorithm based on the constraint method [11], where a feature for the exploration of a specific part of the Pareto front was added, allowing for the choice of an adequate non-dominated solution to the problem.

With the classical constraint method [11], only one objective is optimized, while all the other objectives are constrained to some value. The obtained single objective problem can be solved by conventional methods. The optimal solution to this problem is a non-dominated solution to the original multi-objective problem (see [11]). The bounds that are imposed on the constrained objectives have to be carefully chosen, so that a single optimal solution to the obtained single objective problem exists and so as to guarantee that different non-dominated solutions may be obtained.

In Fig. 1, an example of the application of the MCC method is presented. We consider a single objective problem of minimization of the objective function F_2, whereas a constraint is formulated for the other objective function, i.e., $F_1 \leq F_{1\text{lim}}$. This constraint establishes a new feasible region where we seek to optimize F_2. In this figure the extreme solutions of the Pareto front are shown, where $X \equiv (F_1^{\min}, F_2^{\max})$ and $Y \equiv (F_1^{\max}, F_2^{\min})$.

In the resolution method proposed here, problem P_0 is initially solved by the classical constraint method, where we consider a total of Δ different constraints. The Δ solutions obtained when solving this problem are non-dominated and constitute an approximation to the Pareto front. In Fig. 2 an example of the result after the initial resolution of the routing problem is presented. Note that the proposed algorithm enables that unsupported non-dominated solutions, i.e., non-dominated

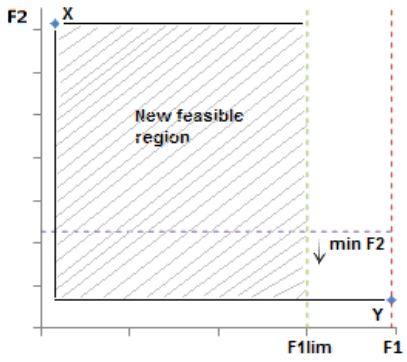


Figure 1. Example of the application of the classical constraint method

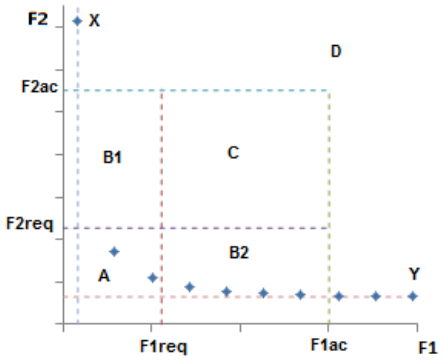


Figure 2. Example of the definition of priority regions in the bidimensional objective function space

solutions located in the interior of the convex hull of the feasible solution set, may be found.

Afterwards, an area of the Pareto front that deserves to be more thoroughly analyzed is chosen, by considering preference regions in the bidimensional objective function space obtained from aspiration and reservation levels (preference thresholds) defined for the two objective functions (see Fig. 2): $F_\varrho^{\text{req}} = \frac{F_\varrho^{\min} + F_\varrho^{\text{av}}}{2}$ and $F_\varrho^{\text{ac}} = \frac{F_\varrho^{\max} + F_\varrho^{\text{av}}}{2}$, with $F_\varrho^{\text{av}} = \frac{F_\varrho^{\min} + F_\varrho^{\max}}{2}$, $\varrho = 1, 2$.

The ideal optimum is obtained when both objective functions are optimized separately. In the 1st priority region A, the requested (req) levels are satisfied for both objective functions; in the 2nd priority regions B_1 and B_2, only one of the requested values is satisfied and an acceptable (ac) value is guaranteed for the other objective function; in the 3rd priority region C, only acceptable values are guaranteed for both objective functions. The least priority region is D. Considering these priority regions, an area of the Pareto front that will be looked into with more detail can be chosen. Firstly, region A will be considered; if there is no possible solution in region A, then region B_1 will be considered; and so on, exploring in succession, regions B_2 and C, if necessary.

After exploring the chosen area of the Pareto front in more detail (again using the constraint method), a few more non-dominated solutions will have been obtained. Finally, the algorithm will proceed to the choice of the most satisfactory

 ISBN: 978-1-61208-263-9

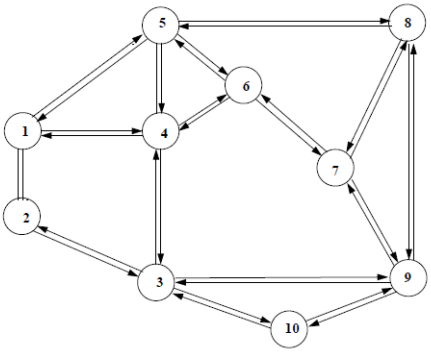


Figure 3. Network in [9, Fig.2]

non-dominated solution in the Pareto front. For this purpose, a Chebyshev weighted metric will be used in the context of priority regions: the approach chosen to select the "best" solution in the Pareto front relies on the minimization of a weighted Chebyshev distance to a reference point, following a method as in [15]. Therefore, this approach will allow us to choose the non-dominated solution whose maximum weighted distance to the reference point is minimum. Notice that the Chebyshev weighted metric will only be applied to the non-dominated solutions found in the best possible priority region. With this approach, we are considering that in the best possible priority region both objective functions F_1 and F_2 have equal importance.

IV. Experimental Results

Experimental results for the network in Fig. 3 (given in [9, Fig.2]), with 10 nodes and 32 unidirectional links, are presented. The capacities of the links and the offered traffic between the different nodes are in [9]. We have superimposed the network on a rectangular grid with 400*240 points where the mesh space unit corresponds to 10 km, as in [15]. Therefore, the maximal horizontal distance in the grid is $l_{\max} = 4000$ km. With this value as reference, we have obtained values for $l_k, k \in \mathcal{A}$.

In [9], the link capacity is the same for all the links, so we have decided not to include it in the link cost c_k, as it affects all the links in the same way. We assumed that $c'_k = \alpha + \beta l'_k$, with a normalized value of l_k: $l'_k = \dfrac{l_k - \min_{\kappa \in \mathcal{A}} l_\kappa}{\max_{\kappa \in \mathcal{A}} l_\kappa - \min_{\kappa \in \mathcal{A}} l_\kappa}$.

The values of network performance measures, relevant from a teletraffic engineering point of view, for the routing solutions obtained with the algorithms were calculated. Some of these performance parameters are 'standard' measures of network performance often used in the evaluation of routing models, such as the one in [16]: total fraction of used capacity, $FUC = \frac{\sum_{k \in \mathcal{A}} f_k}{\sum_{k \in \mathcal{A}} u_k}$; sum of the link utilizations, $SLU = \sum_{k \in \mathcal{A}} \frac{f_k}{u_k}$; maximal link utilization, $MLU = \max_{k \in \mathcal{A}} \left\{ \frac{f_k}{u_k} \right\}$. Other performance measures allow for a comparison of the final solutions with the ideal solutions that would be obtained if a single objective problem was considered: relative variation,

TABLE I. NETWORK PERFORMANCE MEASURE VALUES, FOR THE NETWORK IN [9]

Method	F_1	F_2	RV_1	RV_2	FUC	SLU	MLU
S_1	368.73	5913.53		446.54%	0.5928	18.9710	0.9960
S_2	415.51	1082.00	12.69%		0.5391	17.2500	0.7000
S_{MCC}	378.08	2017.48	2.54%	86.46%	0.5752	18.4073	0.8000

$RV_\varrho = \left| \frac{F_\varrho^{\text{sol}} - F_\varrho^{\text{opt}}}{F_\varrho^{\text{opt}}} \right|$ (with $\varrho = 1, 2$) and where F_ϱ^{sol} is the value of F_ϱ calculated for a specific multiobjective solution and F_ϱ^{opt} is the optimal value of F_ϱ for the same problem. Different solutions are obtained: S_1, the solution obtained when only F_1 is minimized; S_2, the solution obtained when only F_2 is minimized; S_{MCC}, the solution obtained when the algorithm based on the constraint method is used to solve the multiobjective problem.

A total of $|\mathcal{S}| = 4$ services were considered: $s = 0$, a QoS video service with $q_0 = 0.1$; $s = 1$, a QoS *Premium* data service with $q_1 = 0.25$; $s = 2$, a QoS voice service with $q_2 = 0.4$; $s = 3$, a BE data service with $q_3 = 0.25$. In the expression for c'_k, we have considered $\alpha = 0.1$ and $\beta = 1 - \alpha = 0.9$. In these experiments, $N_L = 4$ and $\Delta = 10$.

The results for the considered network are in Table I. The execution time of the algorithm was 2.08 s using CPLEX 12.3 in a laptop computer with i7 processor, 2.2 GHz clock and 1 GB of RAM, running on a Linux VM over Windows.

These results confirm that F_1 and F_2 are indeed conflicting, as the minimization of one of them entails an increase in the value of the other objective function. This confirms the potential advantages of using a multiobjective optimization model, rather than a single objective one, in this routing problem as we get a compromise solution that tries to balance the cost of carrying the bandwidth and the global effect of the utilization of the links. When we optimize only F_1 (results identified by S_1) the total cost of carrying the bandwidth of all the flows is indeed lower, but that is accompanied by a noticeable increase in the utilization of the links, as the values of FUC, SLU and MLU tend to be higher than when only F_2 is optimized (results identified by S_2) or when the multiobjective problem is considered (results identified by S_{MCC}). When we optimize only F_2, the utilization of the links is lower, which makes sense as the minimization of the function F_2 tends to minimize the total utilization of the links. The decrease in the utilization of the links can be confirmed not only by the lower value of F_2 but also by the lower values of the performance measures FUC, SLU and MLU. However, the cost of carrying the bandwidth of all the flows greatly increases, as can be seen by analyzing the value of F_1.

When we solve the bi-objective problem, we realize that the obtained solution has compromise values for functions F_1 and F_2 and also for the performance measures, as one would expect. A balance between the two objective functions can be achieved, so as to guarantee that neither the routing cost is too high (which would happen if only F_2 was optimized) nor the load is unbalanced (which would happen if only F_1 was

 ISBN: 978-1-61208-263-9

optimized).

V. Conclusions and Further Work

In this paper, we presented a multiobjective routing model for MPLS networks with different service types. The routing problem is formulated as a multiobjective MIP, where the objectives were the minimization of the bandwidth cost and the minimization of the load cost in the network links. A constraint related to the splitting of traffic trunks was considered. An exact method was developed for solving the formulated problem, the MCC algorithm. Some experiments have allowed us to obtain results on relevant network performance measures.

The obtained results show that F_1 and F_2 are conflicting and confirm the potential advantages of using this multiobjective routing model, rather than solving a single objective formulation. In this way, the trade-offs between F_1 and F_2 can be analyzed and explored.

The proposed routing method can only be applied in a centralized manner. This type of routing method can be implemented at a network management level (for example in a dynamic routing method with a large update routing period), assuming that the information on the available link capacities is provided.

Further work includes the development of an alternative exact method based on the modified constraint method [17] and an extensive experimental study using other reference networks and randomly generated networks.

Acknowledgment

This work was financially supported by programme COMPETE of the EC Community Support Framework III and cosponsored by the EC fund FEDER and national funds (FCT – PTDC/EEA-TEL/101884/2008 and PEst-C/EEI/UI0308/2011).

References

[1] R. E. Steuer, Multiple Criteria Optimization: Theory, Computation and Application, ser. Probability and Mathematical Statistics. John Wiley & Sons, 1986.

[2] F. Le Faucheur and W. Lai, "Requirements for support of differentiated services-aware MPLS traffic engineering," Request for Comments 3564, Network Working Group, Jul. 2003.

[3] J. Knowles, M. Oates, and D. Corne, "Advanced multi-objective evolutionary algorithms applied to two problems in telecommunications," BT Technology Journal, vol. 18, no. 4, Oct. 2000, pp. 51–65.

[4] H.-W. Ferng and C.-C. Peng, "Traffic splitting in a network: Split traffic models and applications," Computer Communications, vol. 27, 2004, pp. 1152–1165.

[5] J. He and J. Rexford, "Towards Internet-wide multipath routing," IEEE Network, vol. 22, no. 2, Mar.-Apr. 2008, pp. 16–21.

[6] A. Dixit, P. Prakash, and R. R. Kompella, "On the efficacy of fine-grained traffic splitting protocols in data center networks," in Proceedings of SIGCOMM11, Toronto (Ontario), Canada, Aug. 15-19 2011, pp. 430–431.

[7] D. Mitra and K. G. Ramakrishnan, "Techniques for traffic engineering of multiservice, multipriority networks," Bell Labs Technical Journal, vol. 6, no. 1, Jan. 2001, pp. 139–151.

[8] J. Craveirinha, R. Girão-Silva, and J. Clímaco, "A meta-model for multiobjective routing in MPLS networks," Central European Journal of Operations Research, vol. 16, no. 1, Mar. 2008, pp. 79–105.

[9] S. C. Erbas and C. Erbas, "A multiobjective off-line routing model for MPLS networks," in Proceedings of the 18th International Teletraffic Congress (ITC-18), Berlin, Germany: Elsevier, Amsterdam, 2003, pp. 471–480.

[10] J. C. N. Clímaco, J. M. F. Craveirinha, and M. M. B. Pascoal, "Multicriteria routing models in telecommunication networks – Overview and a case study," in Advances in Multiple Criteria Decision Making and Human Systems Management: Knowledge and Wisdom, Y. Shi, D. L. Olson, and A. Stam, Eds. IOS Press, 2007, pp. 17–46.

[11] J. L. Cohon, Multiobjective Programming and Planning, ser. Mathematics in Science and Engineering. Academic Press, 1978.

[12] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in Proceedings of INFOCOM 2000, vol. 2. Tel Aviv, Israel, Mar. 26-30 2000, pp. 519–528.

[13] J. M. F. Craveirinha, J. C. N. Clímaco, M. M. B. Pascoal and L. M. R. A. Martins, "Trafic splitting in MPLS networks – A hierarchical multicriteria approach". Journal of Telecommunications and Information Technology, no. 4, 2007, pp. 3–10.

[14] T. Gomes, L. Martins, and J. Craveirinha, "An algorithm for calculating k shortest paths with a maximum number of arcs," Investigação Operacional, vol. 21, 2001, pp. 235–244.

[15] J. C. N. Clímaco, J. M. F. Craveirinha, and M. M. B. Pascoal, "An automated reference point-like approach for multicriteria shortest path problems," Journal of Systems Science and Systems Engineering, vol. 15, no. 3, Sep. 2006, pp. 314–329.

[16] S. Srivastava, G. Agrawal, M. Pióro, and D. Medhi, "Determining link weight system under various objectives for OSPF networks using a Lagrangian relaxation-based approach," IEEE Transactions on Network and Service Management, vol. 2, no. 1, 2005, pp. 9–18.

[17] A. Messac, A. Ismail-Yahaya, and C. A. Mattson, "The normalized normal constraint method for generating the Pareto frontier," Structural and Multidisciplinary Optimization, vol. 25, no. 2, 2003, pp. 86–98.

 ISBN: 978-1-61208-263-9

Partial Co-channel based Overlap Resource Power Control for Interference Mitigation in an LTE-Advanced Network with Device-to-Device Communication

[1]Sok Chhorn, [2]Tae-sub Kim, [3]Mustafa Habibu Mohsini,
Department of Computer and Information Science
Korea University, Korea
{[1]chhorn168, [2]ree31206, [3]mustafa}@korea.ac.kr

[4]Seung-Yeon Kim, [5]Choong-ho Cho
Department of Computer and Information Science
Korea University, Korea
{[4]kimsy8011, [5]chcho}@korea.ac.kr

Abstract—**In Long Term Evolution-Advanced (LTE-A), many techniques for improving the throughput of the system have been suggested. One among these techniques is the deployment of device-to-device (D2D) communication as an underlay to the International Mobile Telecommunications-Advanced (IMT-A) cellular network. However, deploying D2D technology in overlay macro cellular network may generate high interference to macro users (mUEs) as the D2D devices shares the same spectrum resource with mUEs. In this paper, we propose a partial co-channel based overlap resource power control (PC.OVER) scheme by mitigation of co-channel interference between mUE and D2D receiver (D2DR). In the proposed scheme, when more than one D2DR competes for the same resource with mUE, the power for those D2DRs which compete for the resource with mUE is reduced to low power. The simulation results show that the proposed scheme outperforms D2D with partial co-channel scheme in terms of the system throughput and outage probability for mUEs and D2DRs.**

Keywords-LTE-Advanced; Device-to-Device Communication; Interference avoidance; Resource allocation.

I. INTRODUCTION

Recently, there has been an enormous increase in the amount of data traffics treated by cellular networks, due to the increase in mobile multimedia services. The cellular network needs to adopt these fast growing changes which bring about high demands of data rate services. To achieve this purpose, major efforts have been spent on the development of Third Generation Partnership Project (3GPP) LTE for high data rate and system capacity. Different studies showed that the macro base station (mBS) handles more traffics than in the past years. Installing new base station(s) is expensive and the radio resources in cellular networks are limited. In [1], it has been proposed to handle the local peer-to-peer traffic in a reliable, scalable, and cost-efficient manner by enabling direct Device-to-Device (D2D) communication as an underlay to the International Mobile Telecommunications-Advanced (IMT-A) cellular network.

In D2D communication, users communicate directly with each other or via multi-hop without the intervention of the mBS. The spectrum utilization is improved in D2D communication as the D2DRs share the same resource with macro UEs (mUEs). However, when sharing spectrum with mUEs for data transmission, D2D links may generate high interference to mUEs located in their communication areas [2][3].

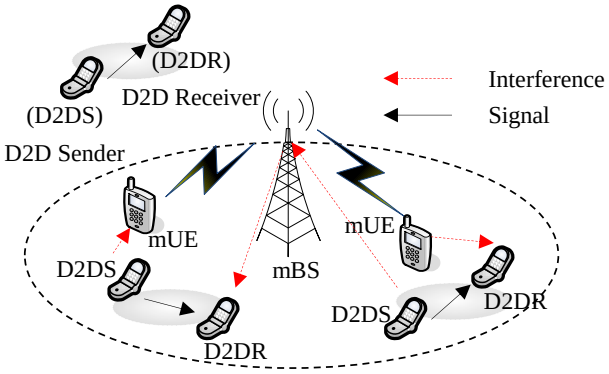


Figure 1. Conceptual diagram of D2D Communication

Interference management in LTE-Advance network with D2D communication is a critical issue. This kind of interference may become even more complex when having great number of D2D pairs across different cells networks [4]. Fig. 1 provides an example of such an interference scenario. For example, there exists an interference from the D2D sender (D2DS) to the mUE (known as inter-tier interference) as indicated by the dashed red arrow.

In this paper, we propose a partial co-channel based overlap resource power control (PC.OVER) scheme aiming to mitigate the co-channel interference between mUEs and D2DR. In low D2DR density, the mUE use any of the available resource block (RB) while the D2DR is restricted to use a portion of the available resources depending on resource allocation ratio (RAR) and use high power for its transmission. In high D2DR density, where more than one D2DRs compete for the same resource with mUE, the power for those D2DRs which compete for the RB with mUE is reduced to low power, P_L. We also consider the spectrum sharing strategies. The simulation results show that there is a significant increase in overall system throughput and the system outage was reduced.

The remainder of this paper is organized as follows: Section II describes the system model. Section III studies the interference scenarios and proposed scheme. In Section IV, three performance measurement indicators have been evaluated as the measurement for our system performance.

 ISBN: 978-1-61208-263-9

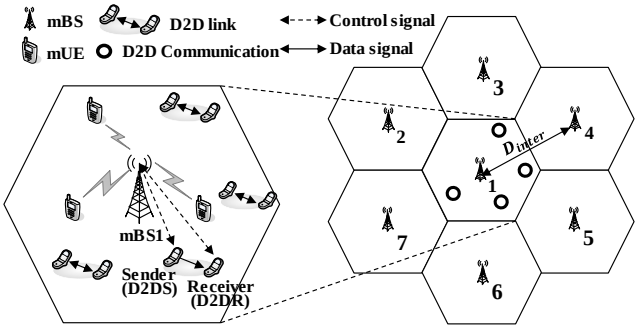


Figure 2. System topology

Section V presents the performance evaluation and Section VI concludes the paper.

II. SYSTEM MODEL

A. System Topology

As shown in Fig. 2, we consider a system topology with 7 hexagonal macrocells where the inter-site distance is D_{inter} designated in meters (m). We assume that each mBS is located at the center of each macrocell and has cell *identification* (ID). mBS denotes an mBS with cell ID = *i* is described as mBS_i . mUEs and D2DSs are randomly deployed in the macrocell coverage and are stationary. Then, D2DRs are separated from their corresponding D2DSs with distance *q*, where *q* is uniform random variable in [1, 20] m. The target cell is the center macrocell, mBS_1 , and interfering neighbor mBSs to mUEs and D2DSs in each cell site of mBS_1.

The physical frame structures in our D2D network is the OFDMA frequency division duplex (FDD). The length of each frame is 10ms and a frame consists of 10 sub-frames. Also, each sub-frame has two slots (a slot is 0.5ms) and each sub channel per slot is the unit of RB [5]. However, in this paper we named it as a sub-channel per symbol RB. The numbers of sub-channels and symbols are S and Z, respectively.

We define RAR, α, between the mBS and D2DSs as

$$\alpha = \frac{D2D\ bandwidth}{Total\ system\ bandwidth}, \quad (1)$$

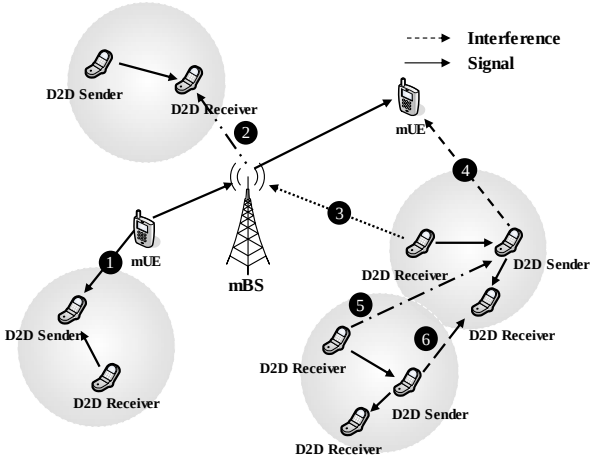
Full frequency bandwidth of the total system bandwidth is allocated to mUE, while D2DS bandwidth depends on the RAR (RAR=0.2).

B. Signal power model

The signal power received, P_r, at mUE and D2DR from mBS and D2DS can be expressed as

$$P_r = P_t * 10\wedge -(PL/10) * L, \quad (2)$$

where P_t is the transmit power of mBS and D2DS, *PL* is path loss, *L* is the shadowing effect, with log-normal distribution with zero-mean and a standard deviation of σ.



Index	Interference Scenario	Interference type	Transmission mode of Macro	Symbol
1	mUE→D2D Sender	Inter-tier	Up link	——→
2	mBS→D2D Receiver	Inter-tier	Down link	·—·→
3	D2D Receiver→mBS	Inter-tier	Up link	·····→
4	D2D Sender→mUE	Inter-tier	Down link	- - -→
5	D2D Receiver→D2D Sender	Intra-tier	Up link	·—·→
6	D2D Sender→D2D Receiver	Intra-tier	Down link	—— →

Figure 3. Interference scenarios for D2D networks

We consider a path loss model in the link between mUE and D2DR [4], where $PL_{mUE_{i,m}}$ is the link between the mBS_i and the *m*-th mUE, $mUE_{i,m}$, in the coverage of mBS_i and $PL_{D2DR_{i,j,h}}$ is the link between the *j*-th D2DS and the *h*-th D2DR, $D2DR_{i,j,h}$, in the *j*-th D2DS coverage of mBS_i, as shown in (3) and (4).

The path-loss is modeled according to the micro-urban models ITU-R report [6]. We apply different path-loss models to D2DRs and mUEs as given in (3) and (4) [7]. The path-losses of the micro-urban models for D2DRs ($PL_{D2DR_{i,j,h}}$) and mUEs ($PL_{mUE_{i,m}}$) are expressed as

$$PL_{D2DR_{i,j,h}} = 40\log_{10} d[km] + 30\log_{10} f_c[MHz] + 49, \quad (3)$$

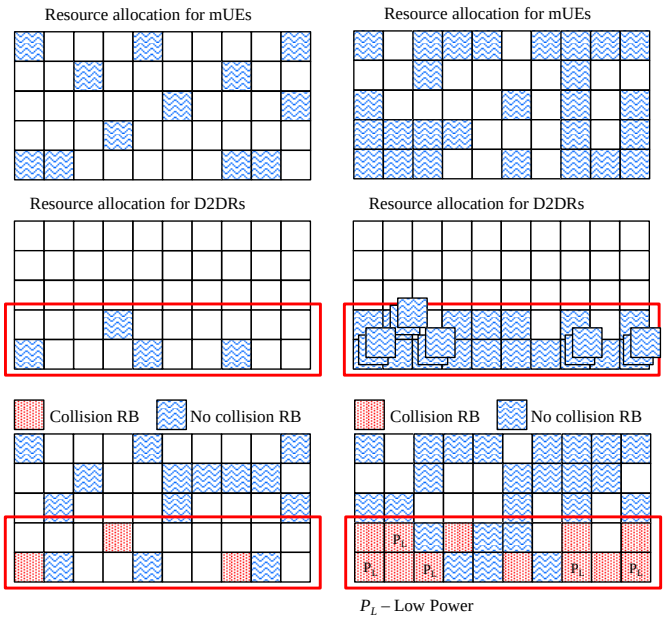
$$PL_{mUE_{i,m}} = 36.7\log_{10} d[m] + 40.9 + 26\log_{10}(f_c[GHz]/5), \quad (4)$$

where *d* represents distance between a sender and a receiver, and f_c means carrier frequency of the system.

III. INTERFERENCE SCENARIOS AND PROPOSED SCHEME

A. Interference Scenarios for D2D Networks

As shown in Fig. 3, we consider two types of interference that occur in a two-tier (Inter-tier and Intra-tier) D2D network architecture. Inter-tier type of interference occurs among network elements that belong to the same tier in the network. In the case of a D2D network, Inter-tier interference occurs between neighboring D2D links. Intra-



a. Sparse Distribution b. Dense Distribution

Figure 4. Proposed Resource Allocation Schemes (RAR=0.2)

tier type of interference occurs among network elements that belong to the different tiers of the network, i.e., interference between D2D links and macrocells.

D2D links are deployed over the existing macrocell network and share the same frequency spectrum with macrocells. Due to spectral scarcity, the D2D links and macrocells have to reuse the total allocated frequency band partially or totally, which leads to inter-tier or co-channel interference. At the same time, in order to guarantee the required QoS to the mUEs, D2DRs should occupy as little bandwidth as possible that leads to intra-tier interference. As a result, the throughput of the network would decrease substantially due to such inter-tier and intra-tier interference.

Fig. 3 illustrates all possible interference scenarios in an orthogonal frequency division multiple access (OFDMA) based D2D network. If an effective interference management scheme can be adopted, then the inter-tier interference can be mitigated and the intra-tier interference can be reduced which would enhance the throughput of the overall network.

B. Proposed Resource Allocation Schemes

The primary goal of this paper is to enhance throughput for both mUE and D2DRs. One way to achieve this is to mitigate interference. In partial co-channel scheme called PC scheme, the mUE transmits in any of the available RB from any of the 50 RBs as showed in Fig. 4(a). The D2DR is restricted to transmit data in the RB depending on the RAR[8].

The PC scheme can be applied to the mitigation of co-channel interference when D2DRs are deployed in a systematic way with low density. However, when multiple mUEs and D2DRs are densely deployed, i.e., having more than one user need to access of the same RB, PC scheme will

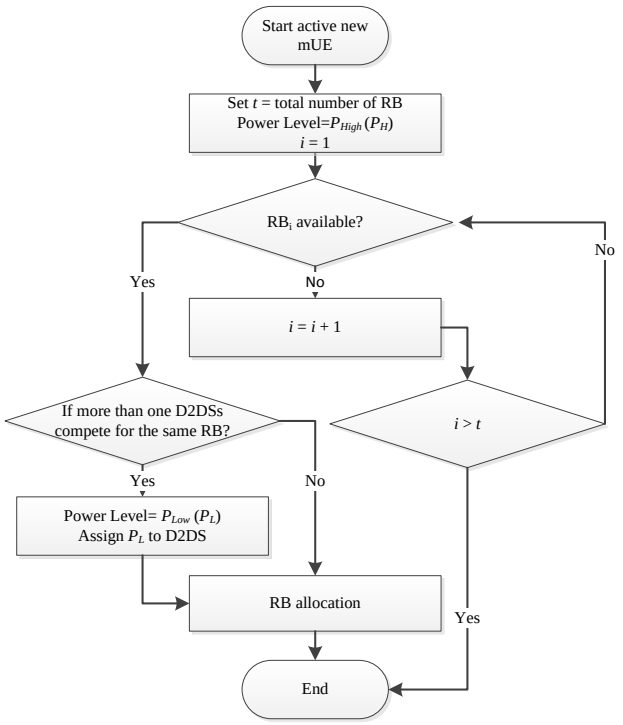


Figure 5. Resource allocation procedure for D2DS

create serious co-channel interference. To solve this problem PC.OVER scheme is proposed to mitigate the interference. In this scheme, the mUE transmits in any of the available RB from those 50 RBs as in PC scheme. The difference is only for the D2D case when we have more than one D2DRs compete for the same RB with the mUE as shown in Fig. 4 (b). The co-channel interference in such kind of situation is severe. Allowing D2DRs to transmit data with their high power will even worsen the interference. To avoid this and further mitigate the interference, for those RBs where more than one D2DRs compete for the same resource with mUE, the power for the competing D2DRs is reduced to low power (P_L) (Fig. 4 (b)). Note that where there is no D2DR competition, we have normal collision as indicated in the red colored RBs. Fig. 5 shows the procedure of how mBSs allocate the RB to D2DRs in the proposed scheme.

IV. PERFORMANCE MEASUREMENT

In this section, the system performance is measured. The detailed explanations of the performance measures used to evaluate the system are described as follows:

A. SINR Model

The SINR model is defined as the ratio of a signal power to the interference power for the b-th RB in the a-th sub-channel, $RB_{a,b}$. We assume that X mBSs are placed in a given area and Y D2DSs are deployed in each macrocell's coverage. Also, L mUEs are serviced by each mBS and F D2DRs are serviced by each D2DS.

 ISBN: 978-1-61208-263-9

Under these assumptions, let $R_{mBS_{i,m}}^{RB_{a,b}}$ and $R_{D2DS_{i,j,h}}^{RB_{a,b}}$ be the power of a received signal for the b-th RB ($1 \le b \le Z$) in the *a-th* sub-channel ($1 \le a \le S$) from the *i-th* mBS ($1 \le i \le X$) to the *m-th* mUE ($1 \le \mathrm{m} \le \mathrm{L}$) in the *i-th* macrocell coverage and from the *j-th* D2DS ($1 \le j \le Y$) to the *h-th* D2DR ($1 \le h \le F$) in the j-th D2DS coverage in the *i-th* macrocell coverage, respectively.

The SINR of the $mUE_{i,m}$ for the $RB_{a,b}$, $\gamma_{mUE_{i,m}}^{RB_{a,b}}$, can be expressed as (5). N_0 is the white noise power. $I_{mBS_{x,m}}^{RB_{a,b}}$ and $I_{D2DS_{x,y,m}}^{RB_{a,b}}$ are the power of the interfering signal from the x-th mBS and from the *y-th* D2DS in the *x-th* macrocell coverage to the $mUE_{i,m}$ for the $RB_{a,b}$. $\omega_{x,m}$ and $\psi_{a,b}$ which are binary values are 1 or 0 if mBS_x is in the group of interfering neighbor mBSs for the *m-th* mUE and the $RB_{a,b}$ is used by the neighbor mBSs or D2DSs, respectively.

The SINR of the $D2DR_{i,j,h}$ for the $RB_{a,b}$, $\gamma_{D2DR_{i,j,h}}^{RB_{a,b}}$, can be expressed as (5).

$$\gamma_{mUE_{i,m}}^{RB_{a,b}} = \frac{R_{mBS_{i,m}}^{RB_{a,b}}}{N_0 + \sum_{x=1,x\neq i}^{X} I_{mBS_{x,m}}^{RB_{a,b}} \cdot \omega_{x,m} \cdot \psi_{a,b} + \sum_{x=1}^{X}\sum_{y=1}^{Y} I_{D2DS_{x,y,m}}^{RB_{a,b}} \cdot \psi_{a,b}},$$

$$\gamma_{D2DR_{i,j,h}}^{RB_{a,b}} = \frac{R_{D2DS_{i,j,h}}^{RB_{a,b}}}{N_0 + \sum_{x=1}^{X} I_{mBS_{x,h}}^{RB_{a,b}} \cdot \psi_{a,b} + \sum_{x=1}^{X}\sum_{\substack{y=1 \\ x=1,y\neq j}}^{Y} I_{D2DS_{x,y,h}}^{RB_{a,b}} \cdot \psi_{a,b}}. \quad (5)$$

B. System Throughput

We analyze the throughputs for $mUE_{i,m}$ and $D2DR_{i,j,h}$, $T_{mUE_{i,m}}$ and $T_{D2DR_{i,j,h}}$, using the Shannon theorem as expressed in (6).

$$T_{mUE_{i,m}} = \sum_{s=1}^{S}\sum_{z=1}^{Z} (RB_{s,z} \cdot \xi_{s,z}) \cdot \log_2(1+\gamma_{mUE_{i,m}}^{RB_{s,z}})$$

$$T_{D2DR_{i,j,h}} = \sum_{s=1}^{S}\sum_{z=1}^{Z} (RB_{s,z} \cdot \xi_{s,z}) \cdot \log_2(1+\gamma_{D2DR_{i,j,h}}^{RB_{s,z}}) \quad , (6)$$

where, $\xi_{s,z}$ is a binary value and $\xi_{s,z} = 1$ else $\xi_{s,z} = 0$ if the $RB_{s,z}$ is used by the $mUE_{i,m}$ and $D2DR_{i,j,h}$.

The system throughputs for mBS and all D2DS, $T_{mBS,i}$ and $T_{D2DS,i}$, in the *i*-th macrocell are calculated by (7).

$$T_{mBS,i} = \sum_{l=1}^{L} T_{mUE_{i,l}} \quad ,$$

$$T_{D2DS,i} = \sum_{y=1}^{Y}\sum_{f=1}^{F} T_{D2DR_{i,y,f}} \quad . \quad (7)$$

C. Outage Probability

We also analyze the outage probabilities, $O_{mBS,i}$ and $O_{D2DS,i}$, for mUEs and D2DRs in the *i*-th macrocell coverage and those are calculated by (8).

$$O_{mBS,i} \approx \frac{N_{mUE,i}^{out}}{L}, O_{D2DS,i} \approx \frac{N_{D2DR,i}^{out}}{Y} \quad , \quad (8)$$

where, $N_{mUE,i}^{out}$ and $N_{D2DR,i}^{out}$ are the numbers of SINR values less than -6dB considering a bit error rate less than 10^{-6} [4] for mUEs and D2DRs, respectively.

V. PERFORMANCE EVALUATION

We investigate the DL performance of the proposed resource allocation scheme using a Monte Carlo simulation. We performed 10,000 independent simulations and evaluated system performance according to the number of mUEs in the analysis. The values of X, S, Z, Y, L, and F are 7, 5, 10, 10~200, 30, and 1, respectively. We assume that the mBS and D2DSs allocate only one RB for each mUE and D2DR, respectively. The mBS does not allocate the same RBs to mUEs in the same cell but D2DSs allocate randomly one RB in allocated channel groups for each D2DR. Log-normal shadow fading is considered with zero mean and standard deviations of 8dB for the link between the mBS and mUEs, and 9dB for the link between the D2DS and D2DRs but multi-path fading is not considered. Table 1 gives the key parameters.

TABLE I. SYSTEM PARAMETERS.

Parameter	Value
Carrier Frequency	2GHz
Bandwidth for DL	10MHz
Bandwidth of sub-channel	180KHz
mBS/D2D radius	866m / 20m
mBS Tx power (P_{mBS})	41.7 dBm(15W)
D2DS Tx power (P_{D2DS})	P_L = 8dBm(6.3mW) P_H= 24dBm(251mW)
Noise power density (N_0)	-174dBm/Hz

We compare the performance of proposed scheme to the network without D2D links and a scheme which allocates radio resource randomly selected from entire frequency band to D2D links. We show the system performance of proposed scheme. The system was evaluated and compared with the conventional scheme where radio resources are randomly selected. 200 D2D pairs were deployed in the region of 30 mUEs. Four cases are considered: consider having only mUEs (w/oD2D), mUEs and D2DRs are randomly allocate

 ISBN: 978-1-61208-263-9

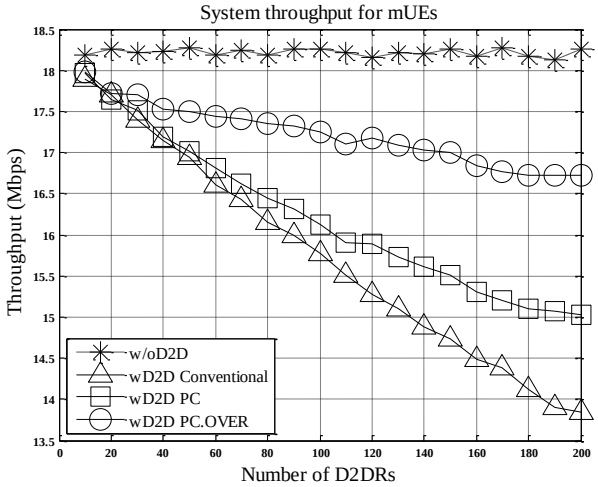


Figure 6. System throughput for mUEs (The number of D2DRs increase)

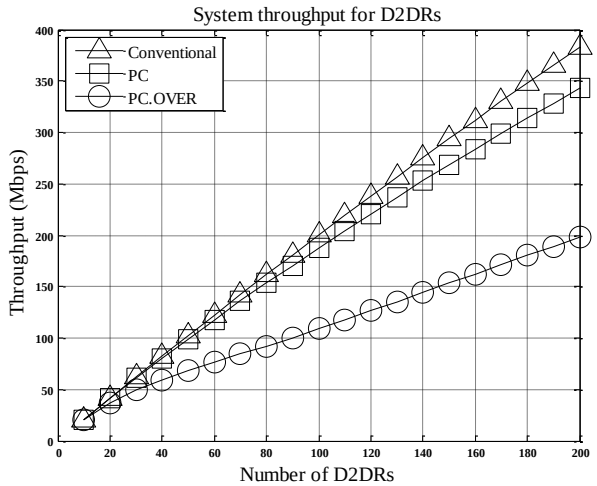


Figure 7. System throughput for D2DRss (The number of D2DRs increase)

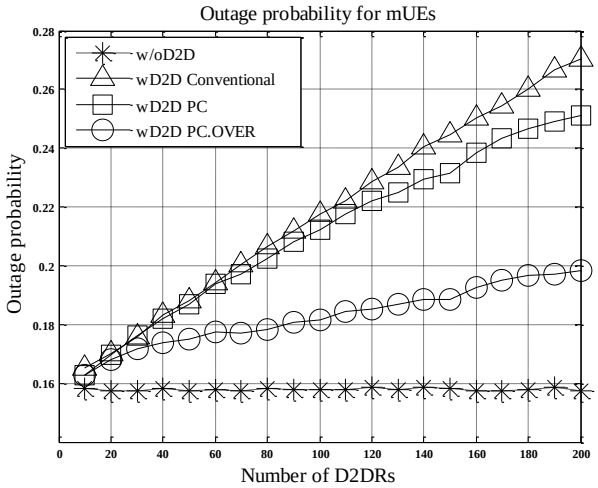


Figure 8. Outage probability for mUEs (The number of D2DRs increase)

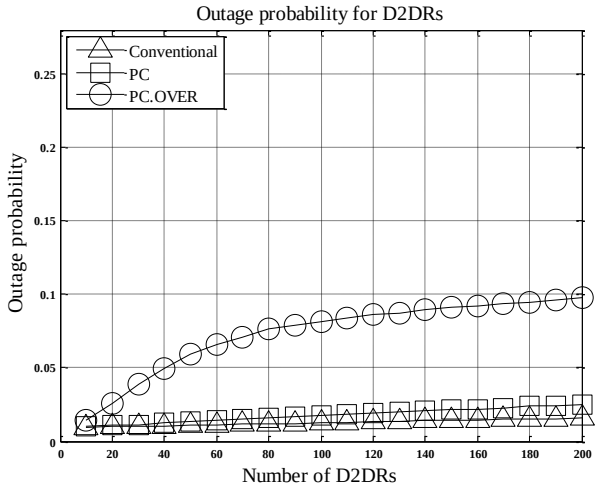


Figure 9. Outage probability for D2DRs (The number of D2DRs increase)

(wD2D conventional), the PC scheme with D2DRs (wD2D PC), and wD2D PC.OVER presenting our proposed scheme.

Fig. 6 shows the system throughput of mUEs for the increasing number of D2DRs per 30 mUEs. Only the DL was simulated. There are several factors that contribute to the throughput variations between the schemes. Since the resources are randomly selected, the throughput for the conventional scheme decreases as the number of D2DR increases. For wD2D PC, there is improvement in system throughput. Though there exist interference between mUE and D2DR for those resources shared by mUE and D2DRs (collision areas), the system throughput is improved because the D2DR transmits only in RBs allocated by RAR. The throughput is further improved in our proposed scheme. This is due to the fact that our scheme avoided further generation of interference by reducing the power of D2DRs competing for the resources with either another D2DR or mUE. Proposed scheme shows better performance than other scheme by mitigating interference between D2DRs and mBS relaying mUE.

Fig. 7 shows the results of D2DRs system throughput that increases linearly, as the number of D2Ds increases. Due to RB exclusion, D2Ds's available RB is less than that of PC scheme and Conventional Scheme; thus, there is a decrease of D2DRs system throughput.

Figs. 8 and 9 show the outage probability for the mUEs and D2DRs respectively. In Fig. 8, comparing with conventional scheme, there is a slight decrease in outage in wD2D PC. Unlike in conventional scheme, the users in wD2D are uniformly distributed which in turn reduces the outage. The outage is further reduced in the wD2D PC. OVER scheme as the users are uniformly distributed and that the D2DRs use low power for data transmission when more than one D2DRs compete for the same RB with mUE. PC. OVER scheme are largely mitigates the interference from D2DRs. The resources also are well utilized in PC.OVER scheme.

 ISBN: 978-1-61208-263-9

In Fig. 9, the outage probability of PC.OVER scheme is generally higher than conventional and PC schemes. The high outage is due to the decrease of the power of the D2DSs which may cause some of the D2DRs to be denied of the services. There is a tradeoff between the system throughput and the outage. Since the system shows significant throughput improvement in mUEs, we still have a strong believe that our proposed scheme performs better than the conventional scheme and PC scheme.

VI. CONCLUSION AND FUTURE WORK

We have studied interference mitigation using partial co-channel based overlap resource power control scheme in LTE-Advance D2D networks. The impact of D2D interference on capacity was investigated. In this paper, we presented a PC.OVER Scheme for D2D outage and throughput. Simulation results showed that the proposed schemes outperform D2D networks in terms of system throughput and outage probability for mUEs and D2DRs. Inter-cell interference is one of the key problems for D2D networks. Thus, in future, we plan to study the improved resource allocation scheme considering Tx power management and the efficiency frequency planning of D2DSs to enhance system performance in future works.

ACKNOWLEDGMENT

This work was supported by the MKE [10035142], Development of an EMM Platform Technology Based on Energy Awareness for High-Efficient Building.

REFERENCES

[1] P. Janis, et al., “Device-to-Device Communication Underlaying Cellular Communications Systems,” International Journal of Communications, Network and System Sciences, vol. 2, no. 3, pp. 169-178, Jun 2009.

[2] K. Doppler, M. P. Rinne, C. Wijting, C. B. Ribeiro, and K. Hugl, "Device-to-Device Communication as an Underlay to LTE-Advanced Networks", IEEE Communications Magazine, vol. 47, no. 12, pp. 42-49, Dec 2009.

[3] K. Doppler, M. Rinne, P. Jnis, C. B. Ribeiro, and K. Hugl, “Device-to-Device Communications; Functional Prospects for LTE-Advanced Networks,” in Proceedings of IEEE International Conference on Communications Workshops, pp. 1-6, Jun 2009.

[4] X. Yanfang, Y. Rui , H. Tao, and Y. Guanding, “Interference-Aware Channel Allocation for Device-to-Device Communication Underlaying Cellular Networks,” in the 1st IEEE International Conference on Communications in China (ICCC), Aug 2012.

[5] 3GPP TS 36.211 v10.0.0, “Evolved universal terrestrial radio access (E-UTRA); Physical channels and modulation,” Jan 2011.

[6] ITU-R report M.2135, “Guidelines for evaluation of radio interface technologies for IMT-Advanced,” Nov 2008.

[7] H. Xing and S. Hakola, “The Investigation of Power Control Schemes for a Device-to-Device Communication Integrated into OFDMA Cellular System,” in Proceeding of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, pp.1775-1780, Sep 2010.

[8] 3GPP TR 25.967 RP-090284, Huawei, “FDD Home NodeB RF Requirements Work Item Technical,” Dec 2008 - Sep 2012.

 ISBN: 978-1-61208-263-9

The MAP in LC Decoding of MTR Codes in Two-Track Magnetic Recording Systems

Nikola Djuric, Vojin Senk
Faculty of Technical Sciences, University of Novi Sad
Trg D. Obradovica 6
21000 Novi Sad, Serbia
e-mail: ndjuric@uns.ac.rs

***Abstract* – The maximum *a posteriori* probability (MAP) algorithm has recently been implemented in Boolean logic circuits (MAP in LC) and presented as an additional method for soft-decision decoding of maximum-transition-run (MTR) codes. Substantial benefits were noticed when the MAP in LC method was used for decoding in an MTR encoded one-track one-head E^2PR4 magnetic recording channel. Those benefits reveal the great potential of possible employment of MTR codes in iterative decoding schemes. Having in mind that MTR codes are able to reduce the overall complexity of trellis channel detection, their utilization in magnetic recording systems with multiple-tracks multiple-heads could be valuable. In this paper, the performance of the MAP in LC decoding was considered in the in-track MTR encoded two-track two-head E^2PR4 channel. The subversion named as *max-log* MAP in LC is presented and compared with *min-max* in LC and *max-log* MAP approach for MTR decoding. In case of the low-level inter-track interference over the recording channel, the *max-log* MAP in LC subversion shows nearly 1 dB of decoding gain, for BER = 10^{-5}, when rate 4/5 (2, 8) MTR was used.**

Keywords-constrained coding; MTR codes; soft-decision decoding; multiple-head recording.

I. INTRODUCTION

Utilization of maximum transition run (MTR) codes [1] as a constrained code for magnetic recording channels, especially in the case of the extended class four partial response channel (E^2PR4) [2], has shown to be quite beneficial [3]. The MTR codes increases the minimum squared Euclidean distance by preventing the ±[+1 –1 +1] error-event, which is dominant at some densities [4]. Consequently, in the case of the two-track two-head E^2PR4 channel model, the MTR employment resulted in 23% of reduction in the number of detection trellis states, and nearly 42% of reduction in the overall channel detection complexity [3].

Even though MTR codes appear to be valuable as constrained codes, it should be emphasized that they possess just a modest error correcting ability. Thus, they have to be used in combination with some proven and powerful error correcting codes, such as the low-density parity-check (LDPC) [5].

Several different approaches had been presented, ranging from the enforcement of MTR constraint into LDPC encoding [6], to the straightforward serial concatenation of LDPC and MTR codes [7], [8]. In all of these instances, the MTR decoding has to be based on a soft-decision approach, so that the corresponding decoder is able to handle soft-values [9], and to produce decision confidence.

In order for them to be implemented in a modern framework of iterative decoding, MTR codes require soft-decision decoding techniques. As a result, this fact encouraged development of several different techniques, such as *min-max* in LC method [7], [8], and the MAP approach [10], [11].

Recently, an additional approach was presented, entitled MAP in LC method, which implements the MAP algorithm into Boolean logic circuits [12], [13]. Utilization of the MAP in LC method offers considerable decoding gain and overall decoding complexity reduction, in case of the MTR encoding over a one-track E^2PR4 magnetic recording channel [7], [8].

In this paper, an extension was made with MAP in LC utilization in magnetic recording systems that use the multiple-track multiple-head approach for data storage [3], [14].

This paper is organized in such a manner that Section II presents an overview of *min-max* in LC and *max-log* MAP algorithm for soft-decision decoding of MTR codes. Section III demonstrates complexity analyses of the MAP decoder, while Section IV explains the MTR encoding over the E^2PR4 two-track channel. Section V offers results of the simulations and performance comparisons, while Section VI gives the conclusion to this paper.

II. MTR SOFT-DECISION DECODING METHODS

This section presents a brief and simple overview of two previously presented methods, the *min-max* in LC [7], [8], and *max-log* MAP approach [10], [11]. In addition, this section is intended to reintroduce notation that will be used in further analyses and explanation of the MAP in LC.

A. Soft-values and Soft-decision Concept

The soft-value of binary variable x is defined as

$$L(x) = \log\left(P(x=1)/P(x=0)\right), \quad (1)$$

where *log*() function is a natural logarithm. The sign of $L(x)$ is the binary decision, the so called hard-decision, while the magnitude of represents the confidence of this decision [9].

The MTR decoder should be able to handle input soft-values and to produce subsequent soft-values on its outputs.

B. min-max in LC Approach

MTR codes can be easily realized using integrated circuit technology and Boolean logic circuits, as it was originally presented by Moon and Brickner [1].

 ISBN: 978-1-61208-263-9

Straightforward and low-cost implementation of the simple and well know rate 4/5 (2, 8) MTR code can be realized, as shown in Fig. 1 [1], [7], [8].

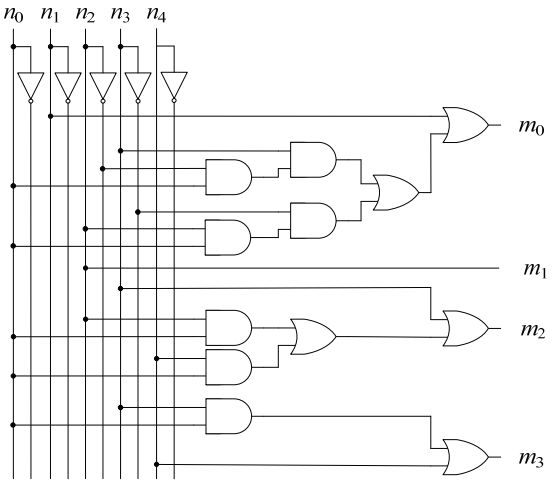


Figure 1. Rate 4/5 (2, 8) MTR decoder.

The idea with the *min-max* in LC method was to use redesigned Boolean logic circuits, which can produce output soft-values according to the following expressions [7], [8]

$$\begin{aligned} L_{out}^{NOT}(x) &= -L_{in}(x), \\ L_{out}^{AND}(x_1,x_2) &= \max[L_{in}(x_1), L_{in}(x_2)], \\ L_{out}^{OR}(x_1,x_2) &= \min[L_{in}(x_1), L_{in}(x_2)]. \end{aligned} \quad (2)$$

where *min*() and *max*() functions return minimal and maximal soft-values, of the input variables.

By implementing such an approach, simple propagation of input soft-values can be realized through the newly created circuits, enabling the *min-max* in LC soft-decision MTR decoding [7], [8].

The *min-max* in LC approach demonstrated excellent performance, when it was used for soft-decision decoding of the MTR code that is combined with LDPC code, over a one-track one-head [7], [8], as well as a multiple-track multiple-head E^{2}PR4 magnetic recording channel [3].

The *min-max* in LC decoding approach is primarily intended for hardware realization of the MTR decoder, using integrated circuits technology.

C. The MAP Approach

MTR codes are simple block codes that basically perform mapping between two sets of sequences. With the defining set

$$N_{set} = \{\underline{n} = (n_0 n_1 \cdots n_{N-1}) \in Z_2^N \mid n_i \in (0,1)\}, \quad (3)$$

$i \in \{1, 2, \ldots N-1\}$, containing MTR codewords, and set

$$M_{set} = \{\underline{m} = (m_0 m_1 \cdots m_{M-1}) \in Z_2^M \mid m_k \in (0,1)\}, \quad (4)$$

$k \in \{1, 2, \ldots M-1\}$, representing output sequences of the MTR decoder, the process of the MTR decoding can be described as "1–1" mapping

$$MTR^{-1} : \underline{n} \in N_{set} \rightarrow \underline{m} \in M_{set}, \quad (5)$$

transforming one sequence from N_{set} to corresponding sequence from M_{set}.

The MAP algorithm is based on subsets

$$N_{bk\,subset} = \{\underline{n} \in N_{set} \mid \underline{m} = MTR^{-1}(\underline{n}) \wedge m_k = b\}, \quad (6)$$

containing those codewords for which MTR^{-1} mapping produces that in a decoder output, at a particular position k, the bit m_k is equal to b, where $b \in (0, 1)$.

These subsets are shown in Table I, for 4/5 (2, 8) MTR code, and for output bit at position $k = 2$ [10], [11], [15],[16].

TABLE I. THE MAP SUBSETS FOR MTR DECODING

$N_{1\,(k=2)\ subset}$		$N_{0\,(k=2)\ subset}$	
$n_0n_1n_2n_3n_4$	$m_0m_1m_2m_3$	$n_0n_1n_2n_3n_4$	$m_0m_1m_2m_3$
0 0 0 1 0	0 0 **1** 0	1 0 0 0 0	0 0 **0** 0
1 0 0 0 1	0 0 **1** 1	0 0 0 0 1	0 0 **0** 1
0 0 1 1 0	0 1 **1** 0	0 0 1 0 0	0 1 **0** 0
1 0 1 1 0	0 1 **1** 1	0 0 1 0 1	0 1 **0** 1
0 1 0 1 0	1 0 **1** 0	0 1 0 0 0	1 0 **0** 0
1 0 0 1 0	1 0 **1** 1	0 1 0 0 1	1 0 **0** 1
1 0 1 0 0	1 1 **1** 0	0 1 1 0 0	1 1 **0** 0
1 0 1 0 1	1 1 **1** 1	0 1 1 0 1	1 1 **0** 1

The $N_{bk\,subsetst}$ subsets are pre-requested for the MAP algorithm and they can be prepared in advance, so that the decoding process can be accelerated.

a) The max-log MAP Subversion

The main subversion of the MAP algorithm operates with probabilities $q_{in\,0}(i)$ and $q_{in\,1}(i)$ of input bit, at sequence position i, which are obtained from (1) as

$$q_{in\,b}(i) = \frac{\exp((-1)^{b+1} L_{in}(i))}{F(i)}, \quad (7)$$

where $F(i) = \exp(L_{in}(i)) + \exp(-L_{in}(i))$.

The output probabilities of bit, at position k, in the decoder output sequence, are calculated as

$$\begin{aligned} q_{out\,b}(k) &= P(m_k = b) = \\ &= \sum_{\underline{n} \in N_{bk\,subset}} \prod_{i=0}^{N-1} P(n_i = b) = \sum_{\underline{n} \in N_{bk\,subset}} \prod_{i=0}^{N-1} q_{in\,b}(i). \end{aligned} \quad (8)$$

It can be observed that expression $F(i)$ depends solely on input soft-values, and does not depend on codewords in subsets $N_{bk\,subset}$. Thus, it can be extracted from the sum, when expression (7) is substituted in (8) [10], [11], [15], [16].

Moreover, the probabilities $q_{out\,0}(k)$ and $q_{out\,1}(k)$ will appear in output soft-value (1), the same as the difference of two *log*() functions, and, thus, it is possible to work with

$$\begin{aligned} r_b(k) &= \sum_{\underline{n} \in N_{bk\,subset}} \prod_{i=0}^{N-1} \exp((-1)^{b+1} L_{in}(i)) \\ &= \sum_{\underline{n} \in N_{bk\,subset}} \exp\left(\sum_{i=0}^{N-1} (-1)^{b+1} L_{in}(i) \right), \end{aligned} \quad (9)$$

without affecting or changing the way in which the output soft-value $L_{out}(k)$ can be calculated [10], [11], [15], [16].

Furthermore, decoder probabilities from (9), can be expressed in logarithmic form as

$$R_b(k) \equiv \log r_b(k) \approx \max_{\underline{n} \in N_{bk\,subset}} \left(\sum_{i=0}^{N-1} (-1)^{b+1} L_{in}(i) \right), \quad (10)$$

using the following approximation [9], [10], [12], [13]

$$\log\left(\sum_{i=1}^{P} \exp(a_i) \right) \approx \max(a_1, a_2, \ldots, a_P), \quad (11)$$

where *max*() returns maximal value among variables.

Using this logarithmic form, the output soft-value $L_{out}(k)$ can be computed as

$$L_{out}(k) = R_1(k) - R_0(k), \quad (12)$$

leading to the new subversion for soft-decision decoding of MTR codes, named as *max-log* MAP subversion.

D. The MAP in LC Approach

The conventional MAP approach considers MTR decoding as simple sequence mapping, computing $L_{out}(k)$, of particular bit k, using corresponding $N_{bk\,subsets}$.

The Boolean logic circuits can be seen, also, as sequence translators and thus implementation of the MAP approach imposed in logical circuits seems rational. The idea is to try to embed the *max-log* MAP approach into decision logic of new circuits, and later to design a new MTR decoder with such redesigned logic circuits.

a) The max-log MAP in LC for new AND circuit

In case of the AND circuit, the mapping and the corresponding MAP subsets are shown in Table II [12], [13].

TABLE II. MAP SUBSETS FOR AND LOGIC CIRCUIT

AND			$N_{1\,subset}$		$N_{0\,subset}$	
n_0n_1	m_0		n_0n_1	m_0	n_0n_1	m_0
0 0	0				0 0	0
1 0	0				1 0	0
0 1	0				0 1	0
1 1	1		1 1	1		

It can be observed that the length of the input codeword is $N = 2$, while of the output word $M = 1$. In that sense, the *max-log* MAP in LC works with just a few elements in the corresponding subsets, and, most importantly, with short-length codewords.

According to (10), the *max-log* MAP in LC subversion works with values

$$\begin{aligned} R_0 &= \max[-L_{in}(0) - L_{in}(1), -L_{in}(0) + L_{in}(1), \\ &\quad L_{in}(0) - L_{in}(1)], \\ R_1 &= L_{in}(0) + L_{in}(1), \end{aligned} \quad (13)$$

while, the output soft-value is calculated as

$$L_{out}^{\max-\log MAP\,in\,LC} = R_1 - R_0, \quad (14)$$

allowing for simple creation of the output soft-values of the new redesigned AND circuit.

b) The max-log MAP in LC for new OR circuit

In case of the OR circuit, the corresponding MAP subsets are shown in Table III.

TABLE III. MAP SUBSETS FOR OR LOGIC CIRCUIT

OR			$N_{1\,subset}$		$N_{0\,subset}$	
n_0n_1	m_0		n_0n_1	m_0	n_0n_1	m_0
0 0	0				0 0	0
1 0	1		1 0	1		
0 1	1		0 1	1		
1 1	0				1 1	0

According to (10), the *max-log* MAP in LC for new OR circuits works with values

$$\begin{aligned} R_0 &= \max[-L_{in}(0) - L_{in}(1), L_{in}(0) + L_{in}(1)], \\ R_1 &= \max[-L_{in}(0) + L_{in}(1), L_{in}(0) - L_{in}(1)], \end{aligned} \quad (15)$$

while the output soft-value is obtained similarly to (14).

Via utilization of such redesigned circuits into hardware realization, as depicted in Fig. 1, the new soft-decision decoder for MTR decoding can be realized and, what is more, implemented in iterative decoding schemes.

III. COMPLEXITY OF THE MAP DECODER

The main problem with the MAP algorithm implementation lies in the potentially high number of codewords in the corresponding $N_{bk\,subsets}$ [10], [11], [15], [16].

The MAP decoding approach primarily leans towards implementation at the software level. Unfortunately, depending on the code rate of the used MTR code, the $N_{bk\,subsets}$ can contain a considerable number of codewords that can unnecessarily slow down the decoder and the decoding process.

A. Complexity of the max-log MAP Approach

Considering general MTR code, with code rate $R = M/N$, it can be easily shown that *max-log* MAP subversion requires

$$\begin{aligned} N_{oper}^{b}(k) &= N \cdot (1add \cdot 1multip) \cdot (N-1)add \cdot \frac{2^M}{2} words \\ &= 2^{M-1} \cdot N \cdot (N-1), \end{aligned} \quad (16)$$

operations to produce the $R_b(k)$, according to (10), leading to the total number of

$$N_{oper}^{total}(k) = 2 \cdot N_{oper}^{b}(k) = 2^M \cdot N \cdot (N-1) \quad (17)$$

operations, necessary to produce the output soft-value, for bit at position k, according to (12). During this analysis it was assumed that addition has the same complexity as multiplication.

Given that the both $N_{bk\,subsets}$ contain the same number of codewords, each output of the *max-log* MAP decoder requires the same number of operations. In case of the rate $R = 4/5$ MTR code this number is

$$N_{oper}^{total} = 2^4 \cdot 5 \cdot (5-1) = 320, \quad (18)$$

operations for each decoder output.

B. Complexity of the max-log MAP in LC Approach

Considering the *max-log* MAP in LC implementation for AND and OR logic circuit, it can be seen that the required number of operations, in order to produce probabilities R_0 and R_1, according to (13) and (15), is

$$N_{oper}^{AND\ or\ OR} = 4 \cdot (2multip + 1add) = 4 \cdot 3 = 12. \quad (19)$$

This is the number of operations for the output of one re-designed circuit (AND or OR circuit). Although, it should be kept in mind that if an MTR decoder is realized, as presented in Fig. 1, then the actual number of required operations for the decoder outputs, in case of the *max-log* MAP in LC, is

$$\begin{aligned} N_{oper}^{m_0} &= 4AND \cdot 12 + 2OR \cdot 12 = 4 \cdot 12 + 2 \cdot 12 = 72. \\ N_{oper}^{m_1} &= 0, \\ N_{oper}^{m_2} &= 2AND \cdot 12 + 2OR \cdot 12 = 2 \cdot 12 + 2 \cdot 12 = 48, \\ N_{oper}^{m_3} &= 1AND \cdot 12 + 1OR \cdot 12 = 1 \cdot 12 + 1 \cdot 12 = 24. \end{aligned} \quad (20)$$

It can be observed that the *max-log* MAP in LC approach considerably decreases the required number of operations per output, comparing with conventional *max-log* MAP, as presented in Fig. 2.

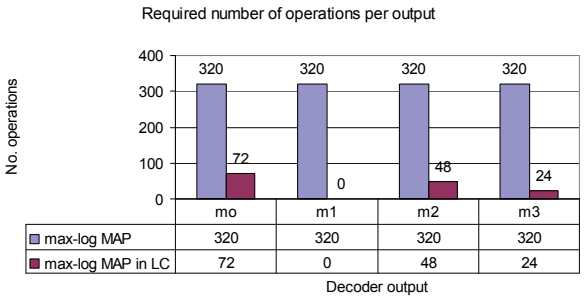


Figure 2. *max-log* MAP in LC versus *max-log* MAP.

Such a result is quite valuable in the case of extensive decoder utilization and demanding signal processing. However, as the realization of the MTR decoder using logic circuits implies that some optimization techniques will be used to decrease the overall number of logic circuits, then the real percent of reduction in a number of operations can differ between realizations.

However, an important result of analyses is that merging the MAP algorithm into Boolean logic circuits and working with the *max-log* MAP in LC will overcome the complexity of the original *max-log* MAP method.

IV. MTR ENCODING OVER A TWO-TRACK CHANNEL

Multiple-head arrays had been proposed to enable reading and writing data at the same time on multiple tracks [14]. Such heads provide both high density and high speed [17], but they suffer from inter-track interference (ITI) [18]. The ITI is a result of a signal induced in reading heads as a superposition of magnetic transitions in neighboring tracks.

A. Channel Model

This paper considers a simple two-track two-head E^2PR4 recording channel, where two independent tracks exist and the reading heads simultaneously detect signals from both tracks [18]. It is assumed that linear and symmetrical ITI is present and modeled with the following matrix

$$\mathrm{A} = \begin{bmatrix} 1 & \varepsilon \\ \varepsilon & 1 \end{bmatrix}, \quad (21)$$

where $\varepsilon \in [0,1]$ represents the ITI level between tracks [18].

A coding scheme employs rate 4/5 (2, 8) MTR code [1], as an in-track constrained code, so that each track is independently encoded, as shown in Fig. 3.

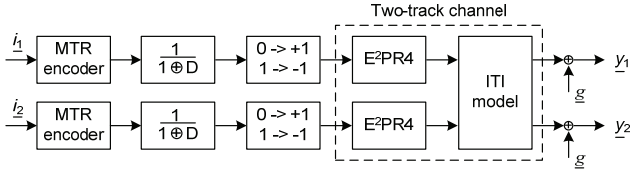


Figure 3. In-track MTR encoding over two-track channel.

where $\underline{i}_1$, $\underline{i}_2$, $\underline{y}_1$ and $\underline{y}_2$ are in-track input and output sequences.

Furthermore, it is assumed that read-back signals are distorted with additive, white and zero-mean, Gaussian noise g and that signal-to-noise ratio (SNR) is defined as

$$SNR = 10\log\left(\frac{E_b}{N_o}\right) = 10\log\left(\frac{E_b}{2\sigma^2}\right) = 10\log\left(\frac{E_c}{2R\sigma^2}\right), \quad (22)$$

where $E_c = RE_b$ is symbol bit energy at channel output, N_o is one-sided power spectral density and σ^2 is noise variance.

Channel detection was performed with the optimum two-head soft-output Viterbi detector (2H-SOVA) that uses ideal ITI estimation and the twenty symbols detection window [3], as shown in Fig. 4.

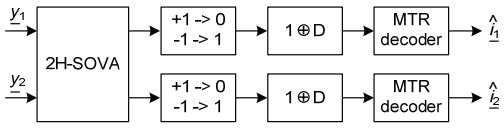


Figure 4. MTR decoding over two-track two-head.

Assuming that y_{ki} is a received symbol at instant i, in track k, the 2H-SOVA calculates branch distance between (y_{1i}, y_{2i}) and noiseless trellis transition label (v_{1i}, v_{2i}), as

$$[y_{1i} - (v_{1i} + \varepsilon v_{2i})]^2 + [y_{2i} - (\varepsilon v_{1i} + v_{2i})]^2, \quad (23)$$

where (u_{1i}, u_{2i}) is corresponding information bits label for the ITI-based trellis and ε represents ITI level [3].

B. Two-track Squared Euclidian Distance

The ITI presence in the two-track channel model can be used to partially improve channel detector performance [18].

It can be shown that knowing and incorporating ITI level into 2H-SOVA branch metric (23) considerably enhances the square Euclidian distance of two-track detector, regarding to independent in-track detection approach, as shown in Fig. 5 [4], [18], [19], [20].

Utilization of the two-head detector that simultaneously reads data from both tracks can mitigate detector performance degradation encountered by ITI presence.

The depicted square Euclidian distance demonstrates the advantage of two-head detector employment over an interfering channel. Over the range of ITI values

$$0 < \varepsilon < \varepsilon_d = 0.293 \quad (24)$$

Euclidian distance gradually increases and a growth of about 7.18% can be noticed, even though track interference is present [18], [19], [20].

This feature will help the two-head detector to successfully combat the low-level ITI in the interfering channel.

 ISBN: 978-1-61208-263-9

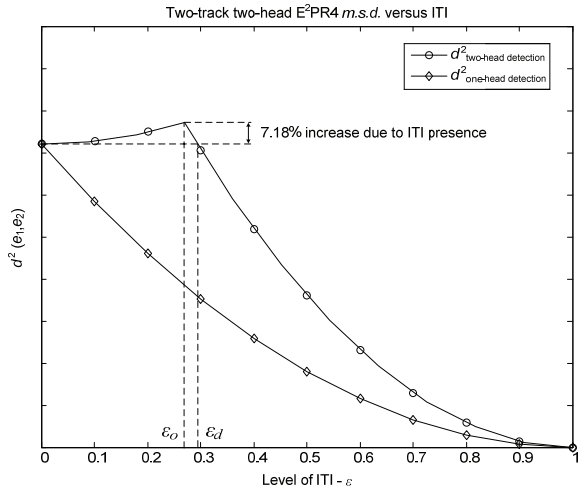


Figure 5. Two-track two-head *m.s.d.* versus ITI.

Regrettably, independent in-track detection is hindered by the ITI presence, degrading the performance of the one-head detector [19], [20]. In such a detection approach, the one-head detector is unable to combat against ITI. This fact additionally suggests that a two-head detector is highly desirable within interfering magnetic recording systems.

V. SIMULATION RESULTS

This paper is intended to analyze only the performance of the proposed algorithms. Thus, the simulation scheme implements only the MTR code, even though they possess a modest error correcting ability [1]. The paper focus is soft-value propagation through the MTR decoder and the complexity of the proposed algorithms.

A. MTR Decoding Using min-max in LC Approach

The *min-max* in LC was the first presented method for the MTR soft-decision decoding [7], [8]. In order to maintain consistency and to easily evaluate the MAP in LC method, the performances of soft-decision decoding using the *min* and *max* functions in logic circuits, are repeated in Fig. 6 [7], [8].

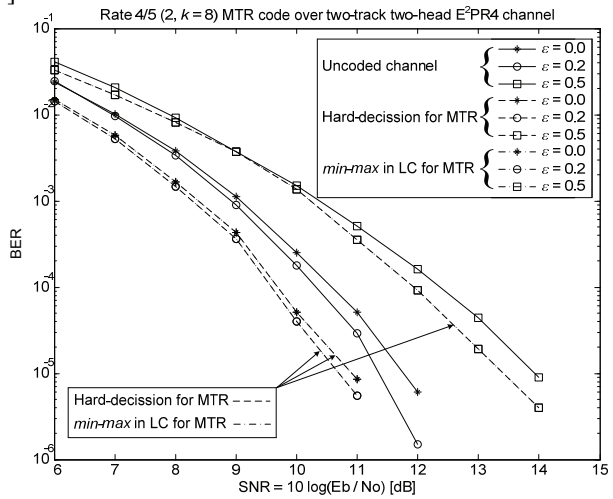


Figure 6. MTR decoding using *min-max* in LC.

The MTR is a single code implemented in an analyzed simulation scheme. Therefore, when the appropriate calculations are finished, the final decision is made in a binary way, even the MTR decoder internally operates with soft-values.

It can be observed that utilizing the *min-max* in LC, the decoding gain is about 0.5 dB for BER = 10^{-5} and ITI level $\varepsilon = 0.0$. Moreover, an additional gain is present for ITI level of $\varepsilon = 0.2$, because of the enhancement of the squared Euclidian distance [19], [20].

The *min-max* in LC method application outcome is identical to the performance achieved with the classical hard-decision approach, but the reason behind this is the solitary role of the MTR decoder, and its inability to fully exploit the soft-values and the corresponding confidences.

However, its advantage is that the MTR decoder is now able to handle soft-values, having performances not weaker than those obtained with conventional hard-decision [1].

The capability of the MTR decoder to manage the soft-values will become overt in an encoding scheme that uses combination of MTR and error-correcting codes [15], [16], e.g. in some of the iterative decoding schemes.

B. MTR Decoding Using the Conventional MAP Approach

The performance of the *max-log* MAP soft-decision decoding of MTR codes is presented in Fig. 7.

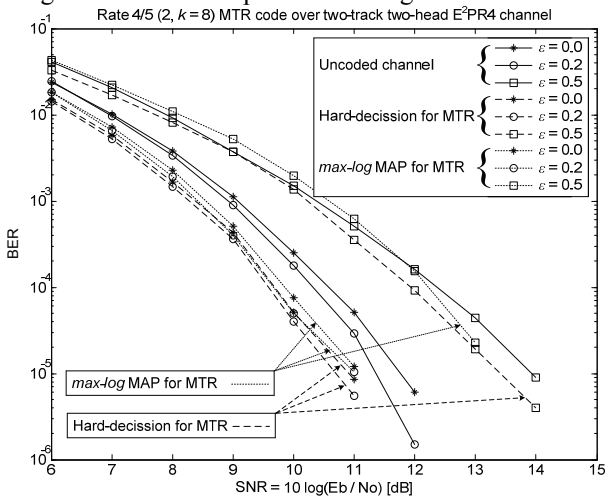


Figure 7. MTR soft-decision decoding using *max-log* MAP.

The *max-log* MAP approach shows a slightly inferior performance to that of hard-decision, but this slight difference is overshadowed by the fact that the hard-decision approach cannot produce output soft-values.

The decoding gain is again around 0.5 dB for BER = 10^{-5} and $\varepsilon = 0.0$, but for ITI level of $\varepsilon = 0.2$, the gain of nearly 1 dB is obtained, for the same BER level.

Such a result indicates that the approximation applied for output probabilities $R_b(k)$ (10), does not hinder the performance of the *max-log* MAP method [10], [11], [15], [16].

C. Comparison of the MTR Decoding Approaches

Finally, a comparison was made between the approaches for MTR soft-decision decoding over the E²PR4 two-track system. The performances are summarized in Fig. 8.

 ISBN: 978-1-61208-263-9

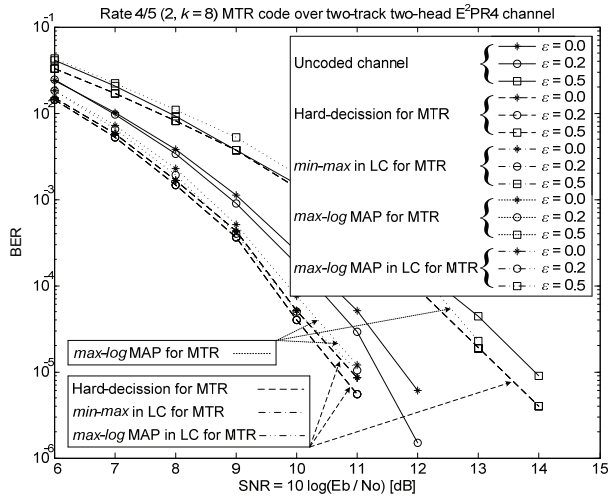


Figure 8. Approaches for MTR soft-decision decoding.

It can be observed that logic circuits utilization, both with *min-max* functions or *log-max* MAP in LC, resulted in a decoding gain of around 0.5 dB for BER = 10^{-5} and $\varepsilon = 0.0$ and nearly 1 dB for of $\varepsilon = 0.2$ and the same BER level.

Considering all presented simulation results, it can be observed that the soft-decision approach has performances that are similar to or somewhat worse than the classical MTR hard-decision [1]. Unfortunately, the solitary position of the MTR decoder, in this simulation scheme, fails to offer exploitation of both decision confidence, and, consequently, the full benefits of soft-decision decoding. Therefore, it is to be expected that the real power of the analyzed approaches will be in some iterative simulation scheme and in combination with some of the error-correcting codes [15], [16].

VI. CONCLUSION

This paper considers the concept of the *max-log* MAP in LC, as an additional method for MTR soft-decision decoding, over the two-track E^2PR4 magnetic recording channel.

This method was compared to already presented MTR soft-decision approaches, suggesting that *min-max* in LC and *max-log* MAP in LC are simpler and more effective than regular *max-log* MAP, which works on the set of MTR codewords.

The low complexity of *max-log* MAP in LC decoder, as well as the fact that MTR codes can reduce the overall two-track channel detection, suggests that their utilization and its soft-decision approaches would be quite valuable, especially in the case of interfering channels.

In addition, simulation results suggest that the *max-log* MAP in LC method will additionally enable MTR code utilization in the iterative decoding framework, and that will result in considerable gain when MTR is to be used in combination with some powerful error-correcting codes over channels for high magnetic recording densities.

ACKNOWLEDGMENT

This paper has been supported by the Provincial Secretariat for Science and Technological Development of the Autonomous Province of Vojvodina, the Republic of Serbia, through the grant for project 114-451-2061/2011-01.

REFERENCES

[1] J. Moon and B. Brickner, "Maximum transition run codes for data storage systems," IEEE Trans. Magn., vol. 32, no. 5, Sep. 1996, pp. 3992-3992.

[2] H. K. Thapar and A. M. Patel, "A class of partial-response systems for increasing storage density in magnetic recording", IEEE Trans. Magn., vol. MAG-25, Sep. 1987, pp. 3666-3668.

[3] N. Djuric and M. Despotovic: "Soft-output decoding in multiple-head MTR encoded magnetic recording Systems," IEEE International Conference on Communications – ICC 2006, vol. 3, Istanbul, Jun 11 – 15, 2006, pp. 1255-1258.

[4] S. A. Altekar, M. Berggren, B. M. Moision, P. H. Siegel, and J. K. Wolf, "Error-event characterization on partial-response channels," IEEE Trans. Inform. Theory, vol. 45, No. 1, Jan. 1999, pp. 241-247.

[5] D. J. C. MacKay and R. Neal, "Near Shannon limit performance of low density parity check codes," IEE Electron. Letters, vol. 33, March 1997, pp. 457-458.

[6] R. M. Todd and R. Cruz, "Enforcing maximum-transition-run code constraints and low-density parity-check decoding," IEEE Trans. Magn., vol. 40, no. 6, Nov. 2004, pp. 3566-3571.

[7] N. Djuric and M. Despotovic, "Soft-output decoding approach of maximum transition run codes", The International Conference on "Computer as a tool" – EUROCON 2005, Nov. 22 – 24, Belgrade, Serbia, pp 490-493.

[8] N. Djuric and M. Despotovic, "Application of MTR soft-decision decoding in multiple-head magnetic recording systems," Indian Academy of Sciences, Sadhana – Academy Proceedings in Engineering Science, vol. 34, Part 3, June 2009, pp. 381–392.

[9] J. Hagenauer, "Source-controlled channel decoding," *IEEE Trans. Comm.*, vol. 43, No. 9, Sep. 1995, pp. 2449-2457.

[10] N. Djuric, "A MAP algorithm for soft-decision decoding of MTR codes," 4th International Conference on Engineering – ICET 2009, Novi Sad, Serbia, April 28 – 30, 2009, pp. 1-3.

[11] N. Djuric: "MAP decoding of MTR codes in LDPC-MTR encoded magnetic recording systems," 9th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services – TELSIKS 2009, vol. 34, no. 3, Nis, Serbia, Oct. 7 – 9, 2009, pp. 381- 392.

[12] N. Djuric and V. Senk, "The MAP implementation in logic circuits for soft-decision decoding of MTR codes," *UKSim-AMSS 6th European Modelling Symposium* – EMS 2012, Malta, Nov. 14 – 16, 2012, pp 201-206.

[13] N. Djuric and V. Senk, "MTR decoding employing MAP algorithm in Boolean logic circuits," IEEE 20th Telecommunications forum – TELFOR 2012, Belgrade, Nov. 20-23, 2012, pp. 803-806.

[14] L. Barbosa, "Simultaneous detection of readback signals from interfering magnetic recording tracks using array heads," IEEE Trans. Magn., vol. 26, no. 5, Sep. 1990, pp. 2163-2165.

[15] N. Djuric and V. Senk, "Methods for the soft-decision decoding of MTR codes in multiple-head magnetic recording systems," 10th IEEE International Conference on Communications – ICC 2010, Cape Town, May 23-27, 2010, pp. 1-5.

[16] N. Djuric, V. Senk, and B. Vasic, "MAP decoding of MTR codes in multiple-head magnetic recording systems," 10th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services – TELSIKS 2011, Nis, 5-8 Oktobar, 2011, pp. 164-167.

[17] P. A. Voois and J. M. Cioffi, "Achivable radial information densities in magnetic recording systems," in Proc. 1992 IEEE Global Telecommunications Conf. – GLOBECOM 1992, Orlando, FL, Dec. 1992, pp. 1067-1071.

[18] E. Soljanin and C. N. Georghiades, "Multihead detection for multitrack recordig channels," IEEE Trans. Inform. Theory, vol. 44, No. 7, Nov. 1998, pp. 2988-2997.

[19] N. Djuric and M. Despotovic, "Distance analysis for E^2PR4 two-track two-head magnetic recording channel", Proceedings of 11th Telecommunications forum – TELFOR 2003, November 25-27, 2003, Belgrade, pp. 1-4.

[20] N. Djuric, "In-track iterative decoding for two-track partial response magnetic recording channels", FACTA UNIVERSITATIS, series: Electronics and Energetics vol.17, pp. 341-351, Dec. 2004.

 ISBN: 978-1-61208-263-9

Evaluation Study of Self-Stabilizing Cluster-Head Election Criteria in WSNs

Mandicou Ba, Olivier Flauzac, Rafik Makhloufi and Florent Nolot
Université de Reims Champagne-Ardenne, France
CReSTIC - SysCom EA 3804
{mandicou.ba, olivier.flauzac, rafik.makhloufi, florent.nolot}@univ-reims.fr

Ibrahima Niang
Université Cheikh Anta Diop, Sénégal
Laboratoire d'Informatique de Dakar (LID)
iniang@ucad.sn

***Abstract*—In the context of Wireless Sensor Networks (WSNs), where sensors have limited energy power, it is necessary to carefully manage this scarce resource by saving communications. Clustering is considered as an effective scheme in increasing the scalability and lifetime of wireless sensor networks. We propose an energy-aware distributed self-stabilizing clustering protocol based on message-passing for heterogeneous wireless sensor networks. This protocol optimizes energy consumption and prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters and by minimizing stabilization time. Our generic clustering protocol can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads, such as nodes' identity, residual energy or degree. We propose to validate our approach under the different election metrics by evaluating its communication cost in terms of messages, stabilization time, energy consumption and number of clusters. Simulation results show that in terms of number of messages and energy consumption, it is better to use the Highest-ID metric for electing CHs. However, the criterion of energy provides a better distribution of clusters.**

***Keywords-Self-stabilizing clustering*; *Wireless Sensor Networks*; *Energy-aware*; *OMNeT++ simulator*.**

I. INTRODUCTION

Due to their properties and to their wide applications, Wireless Sensor Networks (WSNs) have been gaining growing interest in the last decades. These networks are used in various domains like: medical, scientific, environmental, military, security, agricultural, smart homes, etc. [1].

In a WSN, sensors have very limited energy resources due to their small size. This battery power is consumed by three operations: data sensing, communication, and processing. Communication by messages is the activity which needs the most important quantity of energy, while power required by CPU is minimal. For example, Pottie and Kaiser [2] shows that the energy cost of transmitting a 1KB message over a distance of 100 meters is approximately equivalent to the execution of 3 million CPU instructions by a 100 MIPS/W processor. Thus, saving communication power is more urgent in WSNs than optimizing processing. Consequently, to extend the sensor network lifetime, it is very important to carefully manage the very scarce battery power of sensors by limiting communications. This can be done through notably efficient routing protocols that optimize energy consumption. Many previous studies (e.g., Yu et *al.* [3] and Younis and Fahmy [4]) proved that clustering is an effective scheme in increasing the scalability and lifetime of wireless sensor networks. Clustering consists in partitioning the network into groups called clusters, thus giving a hierarchical structure [5].

Several clustering approaches are proposed in the literature and used, for example, in the case of a WSN for routing collected information to a base station. However, most of them are based on state model, so they are not realistic compared to message-passing based clustering ones. Moreover, approaches in the last category are not self-stabilizing and they are generally highly costly in terms of messages, while in the case of WSNs clustering aims at optimizing communications and energy consumption.

In this paper, we propose an energy-aware distributed self-stabilizing clustering protocol based on message-passing for heterogeneous wireless sensor networks. This protocol optimizes energy consumption and then prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters and by minimizing stabilization time. It also offers an optimized structure for routing. Our clustering protocol is generic and complete. It can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads such as: nodes' identity, residual energy, degree or a combination of these criteria. We propose to validate our approach by evaluating its communication cost in terms of messages, stabilization time, energy consumption and number of clusters. Thus, we compare its performance in the case of using different cluster-heads election methods under the same clustering approach and testing framework.

The remainder of the paper is organized as follows. Section II illustrates the related work on clustering approaches. Section III describes the proposed clustering approach, cluster-head election methods and the models used for representing both energy consumption and network structure. Section IV presents the validation of the proposed approach through simulation. Finally, Section V concludes this paper and presents our working perspectives.

II. RELATED WORK

Several self-stabilizing k-hops algorithms have been done in the literature [6], [7], [8].

Mitton et *al.* [6] applied self-stabilization principles over a clusterization protocol they proposed in [9] and presents properties of robustness. Each node calculates its density and

 ISBN: 978-1-61208-263-9

broadcasts it to its neighbors located at k-hops. This robustness is an issue related to the dynamicity of ad hoc networks, to reduce the time stabilization and to improve network stability.

Datta et *al.* [7], by using the criterion of minimal identity, have proposed a self-stabilizing distributed algorithm designed for the state model that computes a subset D is a minimal *k-dominating* set of graph G. By using D as the set of *clusterheads*, G is partitioned into clusters, each of radius k. This algorithm converges in $O(n)$ rounds and $O(n^2)$ steps and requires $\log(n)$ memory space per process, where n is the size of the network.

Caron et *al.* [8], by using an arbitrary metric, have proposed a self-stabilizing *k-clustering* algorithm based on a state model. Note that *k-clustering* of a graph is a partition of nodes into disjoint clusters, in which every node is at a distance of at most k from the clusterhead. This algorithm executes in $O(nk)$ rounds and requires $O(log(n) + log(k))$ memory space per process, where n is the network size.

These approaches are based in state model [7], [8] and are not realistic in the context of sensor networks. Furthermore, they have extremely high stabilization time.

The approach proposed by Mitton et *al.* [6], [9] generates a lot messages. The main reason is due to the fact that each node must know $\{k+1\}$-*Neighboring*, computes its *k-density* value and locally broadcasts it to all its *k-neighbors*. This is very expensive in terms of exchanged messages.

III. Proposed Clustering approach

A. Basic idea

To simplify the description of our approach, we consider the case where the selection criterion to become clusterhead is the node's identity. We will present later the proposed approach when using other CH election criteria.

Our proposed algorithm is self-stabilizing and does not require any initialization. Starting from any arbitrary configuration, with only one type of message exchanged, the nodes are structured in non-overlapping clusters in a finite number of steps. This message is called *hello message* and it is periodically exchanged between each neighbor nodes. It contains the following four information: node identity, cluster identity, node status and the distance to cluster-head. Note that cluster identity is also the identity of the cluster-head. Thus, the hello message structure is $hello(id_u, cl_u, status_u, dist_{(u,CH_u)})$. Furthermore, each node maintains a neighbor table $StateNeigh_u$ that contains the set of its neighboring nodes states. Whence, $StateNeigh_u[v]$ contains the states of nodes v neighbor of u.

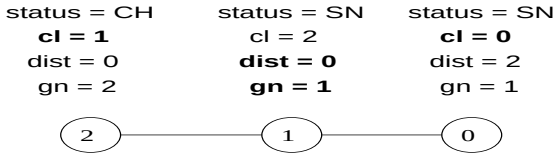
The solution that we propose proceeds as follows:

As soon as a node u receives a hello message, it executes three steps consecutively (see Algorithm 1). The first step is to update neighborhood, the next step is to manage the coherence and the last step is to build the clusters. During the last step each node u chosen as cluster-head the node which optimizes the criterion and located at most a distance k. After this three steps, u sends a hello message to its neighbors. The details of Algorithm 1 and mathematical proof are describe in Ba et *al.* [10].

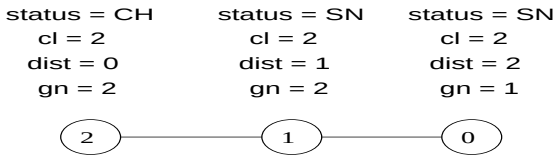
Algorithm 1: k-hops clustering algorithm

```
 /* Upon receiving message from a
    neighbor                                  */
1 UpdateNeighborhood();
2 CoherenceManagement();
3 Clustering();
```

After updating the neighborhood, nodes check their coherency. For example, as a cluster-head has the highest identity, if a node u has CH status, its cluster identity must be equal to its identity. In Fig. 1(a), node 2 is cluster-head. Its identity is 2 and its cluster identity is 1, so node 2 is not a coherent node. Similarly for nodes 1 and 0. Each node detects its incoherence and corrects it during the coherence management step. Fig. 1(b) shows nodes that are coherent.



(a) Incherence nodes



(b) Coherence nodes

Figure 1. Coherent and incoherent nodes

B. Cluster-heads election

Existing clustering approaches use one or more criteria for electing cluster-heads, for example: nodes' ID, degree, density, mobility, distance between nodes, service time as a CH, security, information features or a combination of multiple criteria. However, to the best of our knowledge, there is no paper in the literature where the same proposed approach is compared in the case of different CH election methods. It is important to study the influence of each criterion under the same test conditions and, ideally, under the same clustering approach. To this end, we propose a generic distributed self-stabilizing clustering approach that can be used with any CH election criterion and then we compare its cost and performance when considering important election criteria in the case of a WSN, namely: Highest-ID, Highest-degree and residual energy of nodes.

1) Highest ID:

Lowest-Identifier based clustering is originally proposed by Baker et *al.* [11]. It has proven one of the most performant clustering approaches in ad hoc networks [12], [13], [14], [15].

In our approach, each node compares its identity with those of its neighbors a distance 1. A node u elects itself as a cluster-head if it has the highest identity among all nodes of its cluster

(in Fig. 2, example of node 9 in cluster V_9). If a node u discovers a neighbor v with a highest identity then it becomes a node of the same cluster as v with SN status (in Fig. 2, example of nodes $1, 3, 4$ and 7 in cluster V_9). If u receives again a hello message from another neighbor which is into another cluster than v, the node u becomes gateway node with GN status (in Fig. 2, example of nodes 5 and 8 in cluster V_{10} and node 2 in cluster V_9). As the hello message contains the distance between each node u and its clusterhead, u knows if the diameter of cluster is reached. So it can choose another cluster as illustrated in Fig. 2.

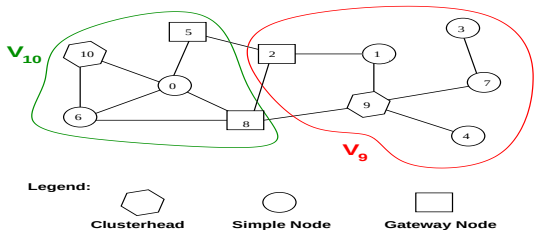


Figure 2. Clusters organization

2) Highest or Ideal Degree: In this approach, we determine how well suited a node is for becoming CH according to its degree D (i.e., the number of neighbors). There are two categories of approaches based on nodes' degrees. Some of them propose to limit communications by electing the node having the highest degree as CH. This is an original proposal of Gerla and Tsai [16]. However, each CH can ideally support only ρ (a pre-defined threshold) nodes to ensure an efficient functioning regarding delay and energy consumption. Indeed, at each step of the routing process, when a node has many neighbors it receives as many messages as its degree. This leads to a rapid draining of sensors' battery power. To ensure that a CH handles upto a certain number of nodes in its cluster, some approaches [14], [17], [18] propose to elect as CH the node having the nearest degree to an ideal value ρ. Thus, the best candidate is the one minimizing its distance to this ideal degree $\triangle_d = |D - \rho|$.

For the two cases described above, when more than one node has the maximum (respectively ideal) degree and is candidate to become a CH, the election is done according to a secondary criterion which is the highest ID. As each node of the network has a unique ID, this criterion is discriminating.

3) Residual Energy: In this approach, decision-making concerning the most suitable node to become CH is done according to the residual energy (i.e., remaining battery power level) of each sensor. Indeed, CHs are generally much more solicited during the routing process. So, in order to preserve their energy and to avoid the frequently reconstruction of the clusters, CHs need more important battery levels compared to the others normal nodes.

During the clustering procedure, network nodes progressively consume their energy due to the messages exchanges. Thus, after some rounds a node i with initially the maximum battery power level and candidate to become a CH can have later less energy than an another neighbor node j. This can lead to more iterations aiming at electing the other node j with the maximum residual energy. In order to limit the frequently changes of CH candidates for a negligible energy difference, we propose to use an energy gain threshold E_T. Thus, while $\triangle_e = |E_i - E_j|$ is less than E_T, the node i preserves its leadership position. This guarantees more stability of the clustering process and extends the network lifetime by minimizing the energy consumption involved in the clustering procedure.

C. Models

In order to implement our clustering approach in a realistic way, we use standard models for representing both the energy consumption and the network structure.

1) Energetic model: To model the energy consumption for a node when it sends/receives a message, we use the first order radio model proposed by Heinzelman et *al.* [19] and used in many other studies [3], [20], [21]. A sensor node consumes E_{Tx} amount of energy to transmit one l-bits message over a distance d (in meters). As shown in equation 1, when the distance is higher than a certain threshold d_0, a node consumes more energy according to a different energetic consumption model.

$$E_{Tx}(l,d) = \begin{cases} l * E_{elec} + l * \varepsilon_{fs} * d^2, & \text{if d} < d_0; \\ l * E_{elec} + l * \varepsilon_{mp} * d^4, & \text{if d} \geq d_0. \end{cases} \quad (1)$$

Each sensor node will consume E_{Rx} amount energy when receiving a message, as shown in equation 2.

$$E_{Rx}(l) = l * E_{elec} \quad (2)$$

The values of the parameters used in equations 1 and 2 to model energy are summarized in Table I:

TABLE I
RADIO MODELING PARAMETERS

Parameter	definition	Value
E_{elec}	Energy dissipation rate to run radio	$50nJ/bit$
ε_{fs}	Free space model of transmitter amplifier	$10pJ/bit/m^2$
ε_{mp}	Multi-path model of transmitter amplifier	$0.0013pJ/bit/m^4$
d_0	Distance threshold	$\sqrt{\varepsilon_{fs}/\varepsilon_{mp}}$

2) Network model: We consider a network represented by an arbitrary random graph based on Erdos Renyi model [22] with probability $p = 0,1$ for all network sizes. Our system can be modeled by an undirected graph $G = (V, E)$. $V = n$ is the set of network nodes and E represents all existing connections between nodes. An edge exists if and only if the distance between two nodes is less or equal than a fixed radius $r \leq d_0$. This r represents the radio transmission range which depends on wireless channel characteristics including transmission power. Accordingly, the neighborhood of a node u is defined by the set of nodes that are inside a circle with center at u and radius r and it is denoted by $N_r(u) = N_u = \{\forall v \in V \setminus \{u\} \mid d_{(u,v)} \leq r\}$. The degree of a node u in G is the number of edges which are connected to u, and it is equal to $deg(u) = |N_r(u)|$.

 ISBN: 978-1-61208-263-9

TABLE II
THEORETICAL COMPARISON OF STABILIZING TIME AND MEMORY SPACE

	Stabilizing Time	Memory space per node	Neighbourhood
Our approach	$n+2$	$\log(2n+k+3)$	1 hop
Datta et al. [7]	$O(n)$, $O(n^2)$	$\log(n)$	k hops
Caron et al. [8]	$O(n*k)$	$O(log(n)+log(k))$	k+1 hops

IV. VALIDATION FRAMEWORK

In this section, we present the evaluation study that we carried out using *ONMeT++* [23] simulator to compare the performance of the previously described clustering approach when utilizing different CH election methods. For generating random graphs, we have used the SNAP [24] library. All simulations were carried out using *Grid'5000* [25] platform.

A. Theoretical validation

In [10], we have provided a formal proof of our clustering approach. Table II illustrates a comparison of stabilizing time and memory space between our proposal algorithm and other approach designed for the state model. We note that our stabilization time does not depend on the parameter k contrary to approach proposed by Caron et *al.* [8]. We have a unique phase to discover the neighborhood and build *k-hops* clusters and an unique stabilizing time contrary to approach describes in [7]. Furthermore, we consider a 1-hop neighborhood at opposed to Datta et *al.* [7] and Caron et *al.* [8].

B. Testbed

The parameters we used in our simulations are summarized in Table III. In all our simulations, a 99% confidence interval I_c is computed for each average value represented in the curves. These intervals are plotted as error bars and computed according to this equation: $I_c = [\overline{x} - t_\alpha \frac{\delta}{\sqrt{n}}; \overline{x} + t_\alpha \frac{\delta}{\sqrt{n}}]$, where n is the population length, $\overline{x}$ is the average value, δ is the standard deviation, and finally, t_α has a fixed value of 2.58 in the case of 99% interval.

TABLE III
SIMULATION PARAMETERS

Parameter	Value
Message size	2000 bits
distance between 2 nodes	100 meters
Ideal degree	{5,20,50}
Energy threshold	{0.1%,0.01%}
Number of nodes	[100,1000]
Random graph model	Erdos Renyi
Network density	0.1
Number of simulations for each network size	100

C. Simulation results

1) Communication cost (messages): In order to evaluate the validity of our clustering approach, we first measure the necessary cost in terms of messages to achieve the clustering procedure.

Based on the same network topology, the clustering based on the criterion of ID generates less messages as shown in Figs 3 and 4. The main reason is that the ID criterion brings greater stability during the clustering phase. In addition, the ID criterion is simpler and deterministic compared to the criteria of degree or energy. Indeed, for the criterion of degree, it is necessary for nodes to receive a message from their neighbors to calculate their degree. Then, the degree is broadcasted and the clustering phase begins. This is expensive in terms of messages. Also, the criterion of energy ration generates more messages than the criteria of ID and degree. As energy is a parameter which decreases during the clustering phase, it provides less stability and requires more messages to reach a stable state in the entire network.

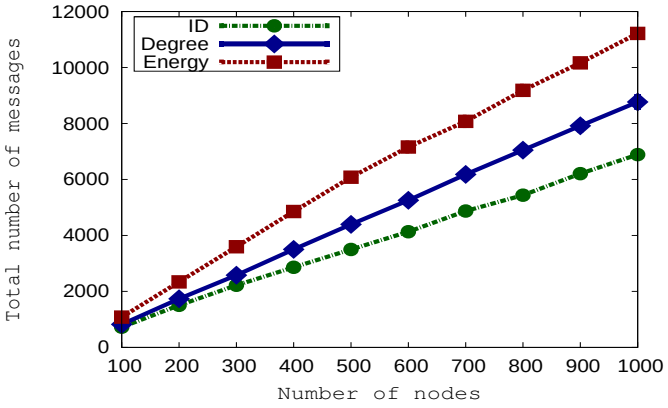


Figure 3. Total number of messages

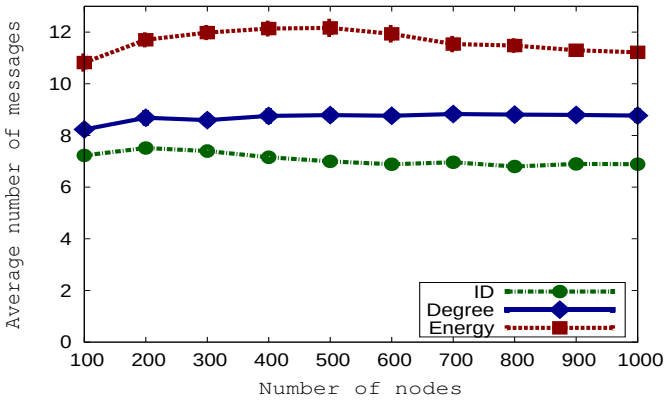


Figure 4. Average number of messages

2) Energy consumption: We have also measured the energy consumption required for building clusters in the entire

network. As illustrated in Figs. 5 and 6, the ID criterion consumes less energy during the clustering phase. Indeed, as illustrated in Figs. 3 and 4, both degree and energy generate more messages than ID during the construction of clusters. However, in sensor networks communications are the major source of energy consumption.

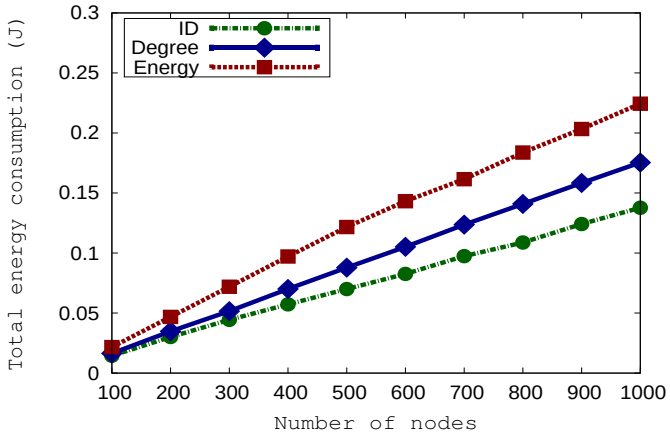


Figure 5. Total energy consumption

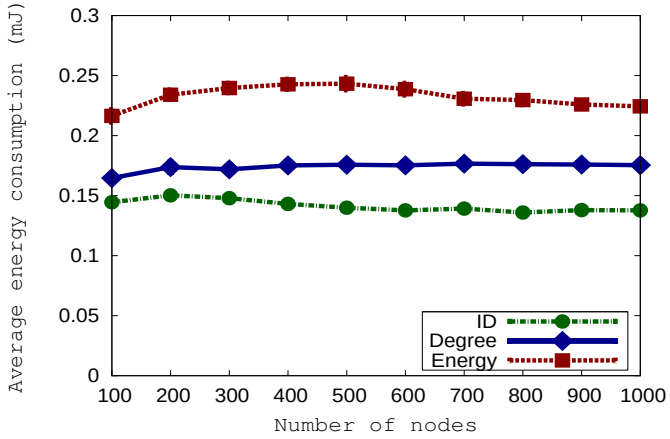


Figure 6. Average energy consumption

3) Number of clusters : The evaluation of the number of clusters as illustrated in Fig. 7 shows that the criterion of energy, even if it generates more messages and greater energy consumption, provides a better distribution of clusters in the network. The main reason is that the criterion of energy does not depend on the network topology contrary to for example the criterion of degree. In fact, in the latter, the node having the highest degree constructs large clusters.

4) Impact of highest and Ideal degree: To evaluate the impact of highest and Ideal degree, we fix $\triangle_d$ to 5, 20 and 40 and then we evaluate energy consumption and clusters distribution. We observe a slight increase in the energy consumption for ideal degree 5, 20 and 40 as illustrated in Fig. 8. Nevertheless, as illustrated in Fig. 9, the ideal degree offers a better distribution of the clusters. Note that the main problem

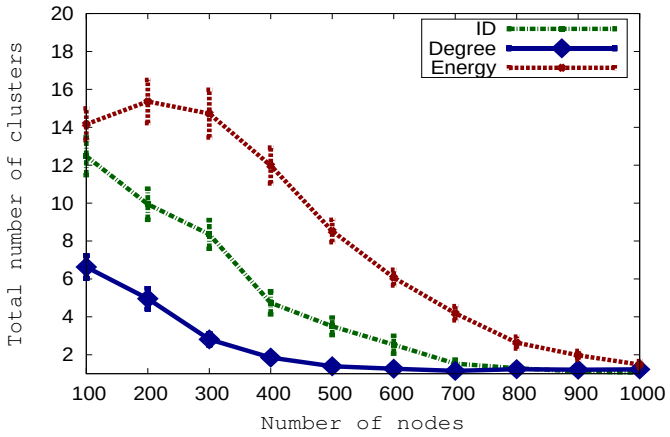


Figure 7. Number of clusters according to network size

with the highest degree is the distribution of clusters.

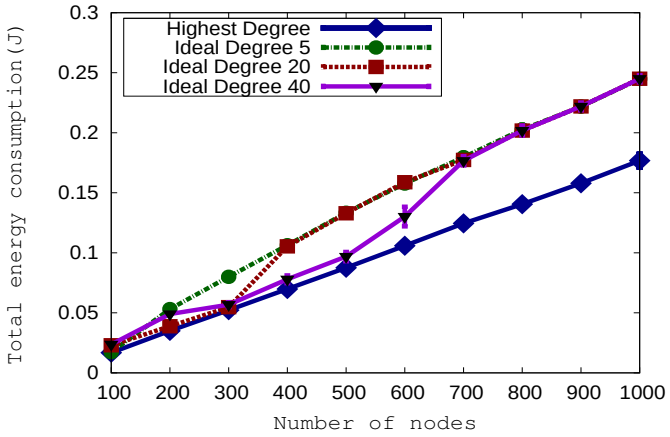


Figure 8. Energy consumption under highest and ideal degrees

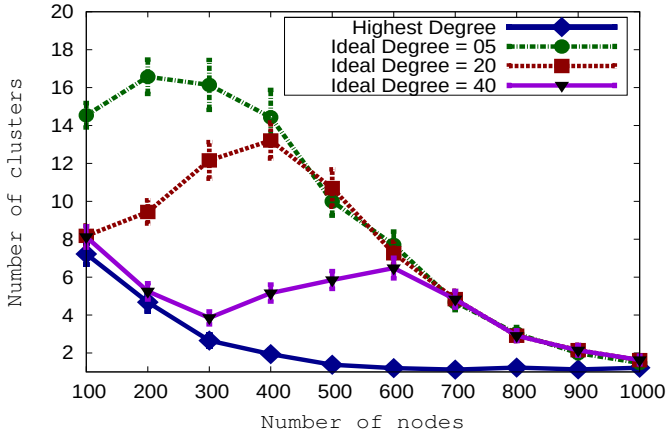


Figure 9. Number of clusters under highest and ideal degrees

5) Impact of residual energy or energy threshold: As the main problem with the criterion of energy is its volatility,

we fix energy threshold to limit abrupt changes of nodes when their energy CHs decreases substantially. We fixed the energy threshold to 0.1% and 0.01% and we evaluate both energy consumption and clusters distribution. Fig. 10 shows that energy threshold reduces energy consumption during the clustering phase. Indeed, the nodes no longer change after slight decrease of their energy CHs. This entails less messages exchanged and less energy consumption. Moreover, energy threshold offers a more balanced distribution of the clusters, as shows in Fig. 11.

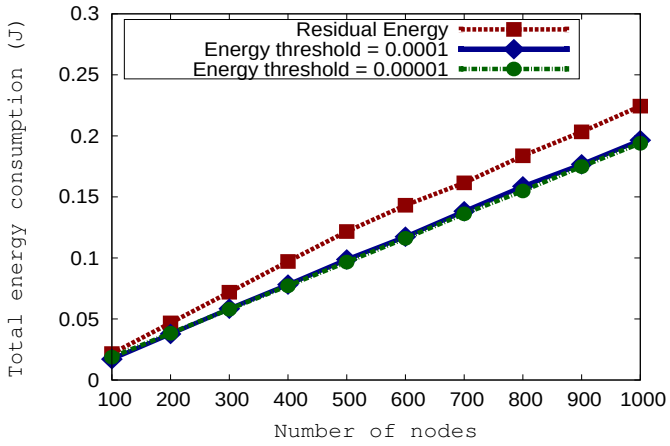


Figure 10. Residual energy *vs* Energy threshold

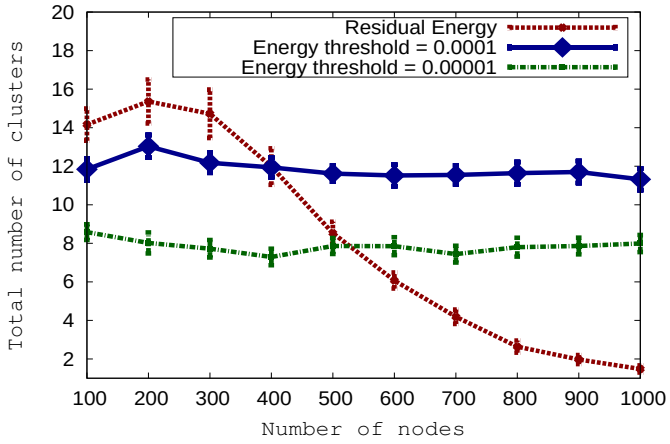


Figure 11. Number of clusters according to energy variation

V. Conclusion and Perspectives

In this paper, we proposed an efficient self-stabilizing distributed energy-aware clustering protocol for heterogeneous wireless sensor networks. This protocol prolongs the network lifetime by minimizing the energy consumption involved in the exchanged of messages. It can be used under different CHs election methods like those investigated in this work.

Simulation results show that in terms of number of messages and energy consumption, it is better to use the Highest-ID metric for electing CHs. However, the criterion of energy provides a better distribution of clusters.

As future work, we plan to propose routing process based on our clustering approach.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," Computer Networks, pp. 393 – 422, 2002.

[2] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," Communications of the ACM, pp. 51–58, 2000.

[3] J. Yu, Y. Qi, G. Wang, and X. Gu, "A cluster-based routing protocol for wireless sensor networks with nonuniform node distribution," AEU - International Journal of Electronics and Communications, pp. 54 – 61, 2012.

[4] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," IEEE Transactions on Mobile Computing, pp. 366–379, 2004.

[5] C. Johnen and L. Nguyen, "Self-stabilizing weight-based clustering algorithm for ad hoc sensor networks," in ALGOSENSORS, pp. 83–94, 2006.

[6] N. Mitton, E. Fleury, I. Guerin Lassous, and S. Tixeuil, "Self-stabilization in self-organized multihop wireless networks," in ICDCSW, pp. 909–915, 2005.

[7] A. K. Datta, S. Devismes, and L. L. Larmore, "A self-stabilizing O(n)-round k-clustering algorithm," in SRDS, 2009, pp. 147–155.

[8] E. Caron, A. K. Datta, B. Depardon, and L. L. Larmore, "A self-stabilizing k-clustering algorithm for weighted graphs," JPDC., pp. 1159–1173, 2010.

[9] N. Mitton, A. Busson, and E. Fleury, "Self-organization in large scale ad hoc networks," in MED-HOC-NET, 2004.

[10] M. Ba, O. Flauzac, B. S. Haggar, F. Nolot, and I. Niang, "Self-stabilizing k-hops clustering algorithm for wireless ad hoc networks," in 7th ACM IMCOM (ICUIMC), pp. 38:1–38:10, 2013.

[11] D. J. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," IEEE Transactions on Communications, pp. 1694–1701, 1981.

[12] Y.-F. Wen, T. A. F. Anderson, and D. M. W. Powers, "On energy-efficient aggregation routing and scheduling in IEEE 802.15.4-based wireless sensor networks," WCMC, 2012.

[13] I. G. Shayeb, A. H. Hussein, and A. B. Nasoura, "A survey of clustering schemes for mobile ad-hoc network (MANET)," American Journal of Scientific Research, pp. 135–151, 2011.

[14] S. K. D. Mainak CHATTERJEE and D. TURGUT, "WCA: A weighted clustering algorithm (WCA) for mobile ad hoc networks," in Cluster Computing, pp. 193–204, 2002.

[15] C.-C. Chiang, M. Gerla, and L. Zhang, "Forwarding group multicast protocol (FGMP) for multihop, mobile wireless networks," Cluster Computing, pp. 187–196, 1998.

[16] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network," Wireless Networks, pp. 255–265, 1995.

[17] W. Choi and M. Woo, "A distributed weighted clustering algorithm for mobile ad hoc networks," in AICT-ICIW, 2006.

[18] M. R. Brust, A. Andronache, and S. Rothkugel, "WACA: A hierarchical weighted clustering algorithm optimized for mobile hybrid networks," in ICWMC, 2007.

[19] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in HICSS, 2000.

[20] J. Wang, J.-U. Kim, L. Shu, Y. Niu, and S. Lee, "A distance-based energy aware routing algorithm for wireless sensor networks," Sensors, pp. 9493–9511, 2010.

[21] J. WANG, J. CHO, S. LEE, K.-C. CHEN, and Y.-K. LEE, "Hop-based energy aware routing algorithm for wireless sensor networks," IEICE Transactions on Communications, pp. 305–316, 2010.

[22] P. Erdos and A. Renyi, "On the evolution of random graphs," Publ. Math. Inst. Hung. Acad. Sci, pp. 17–61, 1960.

[23] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in Simutools, pp. 60:1–60:10, 2008.

[24] SNAP: Stanford Network Analysis Platform. [Online]. Available: http://snap.stanford.edu

[25] F. Cappello and et *al.*, "Grid'5000: A large scale and highly reconfigurable grid experimental testbed," in GRID, pp. 99–106, 2005.

 ISBN: 978-1-61208-263-9

SOA Model for High Availability of Services

Tayyaba Anees and Heimo Zeilinger
Institute of Computer Technology
Vienna University of Technology
Gußhausstraße 27-29, 1040 Vienna, Austria
e-mail: {anees, zeilinger}@ict.tuwien.ac.at

Abstract— **Service-oriented Architecture (SOA) provides reusability and enables easy functionality integration. Service availability in SOA is important as it is used by safety critical systems, telecommunication systems and business systems. Service unavailability can result in reduced profits, reputation damage and reduced safety. Machine virtualization, clusters and group communication systems are used to increase availability, but they are not very much applied to SOA-based systems. This paper focuses on service-orientation and a model for increasing service availability in SOA is proposed. The proposed model improves failure detection process using monitoring. Use of heartbeat mechanism is proposed for failure detection instead of timeout mechanism as it can provide more accuracy and also it can reduce failure detection time. Model is emulated in LAN and WAN environments to investigate impact of different network configurations on service availability. Results indicate that service availability is increased and failure detection process is improved by monitoring.**

Keywords- service-oriented architecture; availability; high availability; monitoring; failover; safety critical systems; business systems; telecommunication systems

I. INTRODUCTION

Service-oriented architecture [1] is for flexibility and reuse, and enables organizations to easily integrate systems [2]. The term SOA followed in the paper is defined by the organization for advancement of structured information standards (OASIS) [1] as "…a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains" [1]. Services are reusable, which can work autonomously as well as in service compositions. SOA follows a standard-based development approach [3]. Service availability is seen in the paper from service consumer's point of view. In our opinion, standards based development makes SOA-based systems more acceptable to service consumers and they can be trusted for quality.

According to authors', availability of services in SOA needs attention as unavailability of services can result in dissatisfaction among service consumers, lost revenues, damaged reputation for service providers and loss of human lives. One of the most important issues for SOA is to assure availability [4]. Business services today are not only doing more work but also have more users, often spread out across the globe and require 24/7 availability and availability is one of the important factors to be considered for business-driven IT service management [5]. The fundamental characteristic of SOA, loose coupling and on-demand integration, enable organizations to seek more flexibility and responsiveness from their business IT systems, but this brings challenges to assure QOS, especially availability, which should be considered in an integrated way in SOA [6].

SOA adoption is increasingly seen in the latest trends where safety critical systems, telecommunication systems and business systems are using it ([7], [8] and [9]). This tendency is due to reduced expected costs due to reusability, which is achievable by using SOA. Service availability is becoming a requisite for such systems as profits can only be earned if service functionality is available to service consumers. Many of these systems require not only availability, but instead high availability for safety as unavailability of service can cause information loss, which can put system into hazardous state, thus reducing system safety. In the paper, the term availability describes the probability that a service in a given time is available.

Availability is dependent on mean time to failure (MTTF) and mean time to repair (MTTR). In the paper, the qualitative description of high availability [11] is followed as highly available systems are those systems which are expected to operate correctly in the presence of multiple failures, using a subset of original components, with reduced capacity and system should be able to self-heal and reconstitute itself, without the loss of data or application services. The system must detect failures and reconfigure system operations dynamically. In our opinion, high availability is expensive and it is not required for all applications or services. Requirements for availability and high availability are dependent on area of application and also on a specific solution. In SOA, requirements for availability and high availability are generally specified in service level agreements (SLA).

This work focuses on increasing service availability by reducing failover time and failure detection time through monitoring. Failover means a backup module taking the workload, when the primary module has failed [12]. Failover time includes the failure detection time and the recovery time [12].

Clusters, group communication systems (GCS) and machine virtualization are solutions, which are used to increase availability. These solutions also use monitoring but they differ in concept, and they are not very much applied within the domains of SOA. In machine virtualization, several operating systems share the resources of same physical machine. Shared physical machine can increase performance overhead and it can

 ISBN: 978-1-61208-263-9

become a single point of failure for virtualized solutions. Clusters use GCS and GCS based solutions require coordination activities between group members, which can impose performance overhead. SOA-based solutions for increasing availability are mostly focusing on service compositions. Middleware based solutions also exist, which use an enterprise service bus. These solutions can also add performance overhead, which can increase failover time.

The proposed approach focuses on the use of monitoring and heartbeat mechanism for failure detection instead of using timeouts for failure detection and retrying in case of failures. The proposed approach simplifies the management of failures by focusing only on the necessary participants of SOA-based systems including service consumers, service providers and service registry. Additionally, the focus is on atomic services instead of service compositions for simplification. In the proposed approach, service provider's availability is seen from service consumer's perspective because they are the ones who ultimately use the service and in this context failover time becomes important, which is the time when service is unavailable. Failures are covered in the approach through redundancy and service provider's availability is determined through failover time. The proposed approach improves the process of failure detection and failover by using heartbeat mechanism and service provider availability is improved by reducing failure detection time through monitoring service provider's failures and by selection of an optimal heartbeat interval.

The remainder of the paper is organized as follows: Section II describes the state of the art research work. In Section III we present and propose a SOA model for improving service availability. It also includes a discussion about the availability parameters considered in the proposed solution and describes the approach for analyzing service availability in SOA. Section IV presents experimental results and discussion. Section V contains concluding remarks.

II. STATE OF THE ART

SOA eases development efforts due to reusability and standards based development approach [3]. As stated by Li [4], SOA is an emerging approach addressing the requirements of loosely coupled, standards-based, and protocol independent distributed systems. Costs are reduced by reusability of components which turns out to be an advantage orchestrating large scale distributed applications [13]. These cost reductions lead to upcoming adoptions of SOA in business computing environments [8].

Recently, a shift in trends is seen and there is a move towards SOA adoption by safety critical systems. SOA is being adopted by military organizations such as the United States Department of Defense, The North Alliance Treaty Organization and the UK's Ministry of Defense [9]. Telecommunication networks are service centric and use service composition techniques in accordance to SOA principles [14].

Platforms that are supposed to form the core of mission critical service-oriented applications need mechanisms that can regulate the reliability and availability of the core services in changing conditions [15].

For increasing availability of services different solutions are proposed. In [16], a solution for improving availability of service compositions or complex services is proposed. Another solution is to pool multiple services that provide the same functionality by different service providers [4]. If a service fails, another service in the pool is selected to process the request again. In this approach, an appropriate size of service pool has to be selected otherwise resources can be underutilized. In the proposed approach, the focus is on reducing failure detection time and failover time by monitoring for increasing service provider availability. If failure detection time is reduced, the time for which the service is unavailable or MTTR is reduced and consequently availability is increased as it is dependent on MTTR. The proposed approach focuses on monitoring failures of individual services and not composite services. In service compositions, wrong execution order of services or failure of one service in a service composition can reduce service availability of all services in a service composition but mostly the focus of service compositions is on a single solution and how services are invoked in that solution. In our opinion, the probability of reuse of individual services is higher than the reuse of service compositions. A single service can be reused in many different business solutions so availability of individual services can be more beneficial as it can increase service availability in different solutions.

The current solutions for increasing availability include machine virtualization [17], clusters [4] and group communication systems [18]. The emergence of machine virtualization has significantly reduced system setup time, coupled with the ability to migrate services and the flexibility to consolidate multiple underutilized servers into a smaller number of machines [19]. These solutions aim at reducing MTTR by using redundancy and failover mechanism. In any reliability work in general, a decrease in MTTR contributes a proportional reduction in unavailability [20]. In most of the high availability distributed systems, redundancy is used to increase availability and redundant servers appear to reduce MTTR [19]. Failover can be used to ensure availability [4]. Failover is realized by heartbeat detection and automatically takeover of functions [21].

Failover requires failure detection and service migration to a redundant service provider. For failure detection, heartbeat mechanism can be used and timeouts can also be used. For failure identification at application level keep-alive probing can be used and applications can set own timeouts [22]. Another common method to detect failures is the error prone approach of timeouts in order to overcome inaccurate failure detections in software [23]. Another possibility to introduce fault tolerance to applications are self checking mechanisms in the code. The application verifies that it is healthy on its own. Through such tactics, most failures can be detect accurately [23].

In the proposed approach, heartbeat mechanism is used for failure detection and for better accuracy in comparison to the timeout approach. Monitoring service constantly checks for service failures and in timeout approach failure is only detected when service consumer sends a request to the service provider. As monitoring is done on a constant basis failures can be detected earlier than the timeout approach where the process of failure detection begins

 ISBN: 978-1-61208-263-9

after consumer sends a service request. In case of monitoring, failed service can be restored using failover before a service consumer sends a request. In failover process, after failure detection, recovery is done by switchover process in which a redundant service provider takes over the work of failed service provider. Runtime monitoring can be used for analyzing and recovering from detected faults [24]. Monitoring is used in the proposed solution for failure detection and recovery.

III. PROPOSED MODEL

A. Availability Parameters

The tendency of SOA adoption raises the need for investigation of availability issues related to SOA. Nowadays, there is a tough competition in every field of life. Business systems and telecommunications systems need to increase customer satisfaction for acquiring higher profits. Safety critical systems need to provide more confidence to service consumers for getting more profits and in worst case for retaining profits. As these systems are using SOA, this can be achieved by improving quality of service attributes, such as by improving service availability in SOA.

This paper proposes a SOA-based model, which can be used by such systems for increasing service provider availability. Basic SOA model includes service providers, service consumers and service registry. In modified SOA model we have added a monitoring service to basic SOA model as shown in Figure 1. In Figure 1, service providers publish service descriptions (1.publish) in registry. Service consumers find (2.find) required service from registry. Service provider has service implementation and service registry holds service description. Service consumers bind (3.bind) to service provider. Service consumers and service provider start interacting (4.interact) with each other. Next section explains the role of monitoring service.

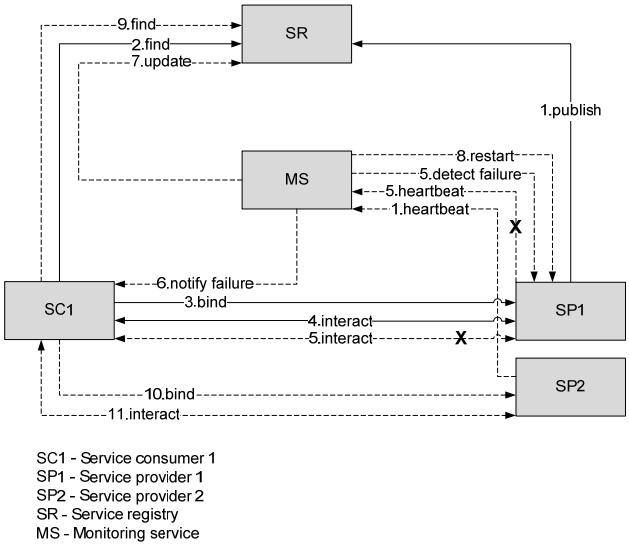


Figure 1. Modified SOA model.

Availability of service provider is essential as they provide functionality to service consumers. In general, availability decreases due to failures. Availability can be increased by increasing MTTF or by reducing MTTR. MTTF is the time until a failure happens and it can be increased by reducing failures from the system. For increasing MTTF statistical data is needed. MTTR is the time to repair the system. As the model proposed in the paper is based on a newly developed system, statistical data is not available for it and statistical data is not always accurate and complete as well. The focus in the proposed approach is on reducing MTTR by reducing failover time through monitoring for increasing service availability. Failover time is the downtime of service. In our opinion, by reducing failover time, availability can be increased.

The requirements for availability and high availability vary for different systems. High availability can be analyzed quantitatively as well as qualitatively. In the proposed solution, availability and high availability are qualitatively analyzed. Qualitative descriptions are used for analyses because SOA is not specific to an area of application and different areas of application can have different requirements for availability and high availability. Quantitative requirements differ for different areas of application whereas qualitative description is applicable in a generic way such as highly available systems should be able to detect failures and restore operations dynamically. Failures of some services can be tolerable and some are not depending on requirements and cost of high availability. Reliable communication is an essential service for many distributed applications, some of which require very fast recovery from failures, while others can tolerate slower failure recovery [25].

In our opinion, in SOA, several services work together to perform a business task and not all services used in a service-centric solution require high availability. For instance, customer interaction services may require high availability because their unavailability can result in monetary losses but services which are used for internal communication between employees may not need high availability as in this case monetary losses are not expected. In our opinion, high availability is expensive and it should not be added without considerable thought.

In SOA, service level agreements are used to describe quality of service parameters [26]. In our opinion, for SOA-based systems requirements of availability and high availability should be specified in SLA. Authors believe that highly available systems have certain features such as redundancy, use of automated means for recovery, minor failures and the ability of failure detection. A system can be considered highly available on the basis of these features. Redundancy, failure detection and automated recovery using monitoring are used for high availability in the proposed approach.

B. Service Availability in SOA

In the proposed solution, for increasing service provider availability in SOA, monitoring service is added to the basic SOA model as shown in Figure 1. Monitoring service detects failures, notifies service consumers about failures and initiates the failover process. Monitoring service deletes the information of failed service from service registry to reduce failures by reducing chances of requests being sent to failed services. Monitoring service restarts the failed service provider as well. UDDI based service registry is used in the test environment as it fulfils the requirements. In proposed solution, availability of

 ISBN: 978-1-61208-263-9

service provider is improved by reducing failure detection time and failover time through monitoring. Failover time includes failure detection time and switchover time. Switchover time is the time that the primary and the backup are switching over the roles [12]. In the model, switchover time is the time for finding the backup service provider from service registry. In the proposed model, redundant service providers are used who send heartbeat messages to monitoring service at periodic intervals. Monitoring service detects failures on the basis of 3 consecutive missed heartbeat messages from the service provider.

In the proposed approach, heartbeat mechanism is used as failure detection time can be reduced with it and it provides greater accuracy in comparison to timeout approach. In case of heartbeat approach, failures can be detected earlier than timeout approach as failures are constantly monitored and service an be restored before a service consumer sends a request to service whereas in timeout approach service failure is detected only after the service consumer sends request to the service. In the approach of timeouts, service consumers are blocked for a certain time to get a response from service. If they do not get a response within a specified time interval, they retry the service and wait for ensuring service failure. In heartbeat mechanism service consumers are not blocked for a certain time and another service can be used as soon as the failure is detected.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The following experiment is conducted to evaluate the proposed approach to increase service availability in service-oriented systems. Test environment for evaluating the proposed model is shown in Figure 2. WANem [27], an emulator is used for evaluating the model in test environment under different network conditions. Four nodes are used in the test setup. All traffic between service consumers, service providers and monitoring service passes through the emulator. Service consumers send requests in parallel to service providers and they are placed on one node. Service registry and service providers are placed on another node. Monitoring service is placed on a separate node. Service consumers and service providers are deployed on Glassfish server [28]. Service registry which is used is jUDDI [29] and it is deployed on Jakarta-Tomcat server [30]. Service registry uses mySQL [31] database for storing information. Synchronous web services are used in the implementation.

In experiments in Figure 3, different number of service consumers and different number of service providers are used with no packet loss or delay. In experiments shown in Figure 4, different number of service consumers and 10 service providers are used with different rates of packet loss 0 %, 1 % and 5 %. Different rates of packet loss are used for analyzing the impact of packet loss on service provider's availability. Packet loss is chosen to analyze the applicability of the proposed model to all kind of services as for some services such as VOIP services high rate of packet loss is expected whereas for other services rate of packet loss can be low. In the experiment, different heartbeat intervals are used to analyze optimal failure detection time and to reduce failover time to increase service provider availability. Model is emulated with 100 ms, 500 ms and 1000 ms heartbeat interval for sending heartbeats. Different heartbeat intervals are used to analyze how fast monitoring service can detect failures accurately. In the experiment, measurements are taken to see the impact of different redundancies and different number of service consumers on failover time.

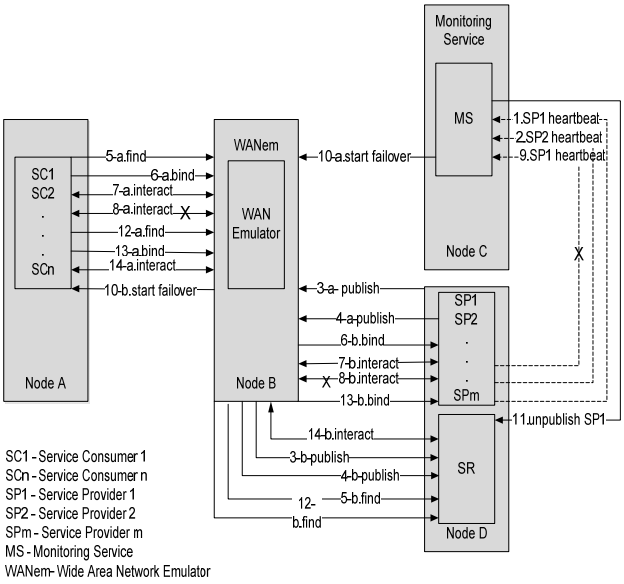


Figure 2. Test environment of modified SOA model.

Failover time is chosen in measurements as it is the time when service is unavailable. Different redundancies are used because overall performance can decrease or failover time can increase by adding more redundancy. In the proposed approach, performance is determined with respect to failover time. Failover time or performance should be acceptable to service consumers. Performance can also decrease with more service consumers as failover time can increase due to more number of service requests with higher number of service consumers.

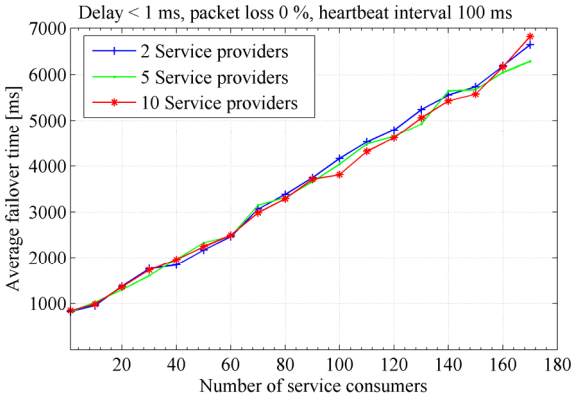


Figure 3. Average failover time with different redundancies.

Results shown in Figure 3, with 0 % packet loss show that by increasing number of service consumers, failover time is increased irrespective of redundancies. Results indicate that failover time with 130 service consumers is 5 s. In our opinion, failover time up to 5 s should be acceptable for most of the service consumers unless there

 ISBN: 978-1-61208-263-9

are critical services which have higher requirements for failover time. In our opinion, for business systems and telecommunication systems 5 s failover time can be tolerable in most of the cases as loss of human lives is not expected with the services provided by business systems. Results indicate that service provider can tolerate failover requests from 130 service consumers at the same time which is quite acceptable according to the capacity of one system. However, by improving the capacity of system using better hardware the limitation of 130 service consumers can be removed. 5 s's failover time is high for critical services used by safety critical systems as they have high requirements for availability and high availability but for non-critical services they can also consider 5 s's failover time.

Results indicate that by increasing redundancy and due to sending of more heartbeat messages, bandwidth consumption is not changed considerably, due to which performance or failover time stays similar with different redundancies. In our opinion, redundancy can be added in the system to increase availability of service according to requirement as performance is not deteriorated with it. We recommend that, for systems where frequency of failures is high or reputation damage is important for a certain business, redundancy can be added for increasing availability because results indicate that performance is not decreased with redundancy. However, if frequency of failures in a system is low, adding more redundancy to the system is not recommended as it will be expensive and it can result in underutilized resources.

Results in Figure 4, indicate that monitoring service can detect failures in less time when a small heartbeat interval is used such as 100 ms. Results show that with a small heartbeat interval, failure detection time is reduced. Failure detection time with 1000 ms heartbeat interval is 3000 ms, with 500 ms heartbeat interval failure detection time is 1500 ms whereas with 100 ms heartbeat interval failure detection time is reduced to 300 ms.

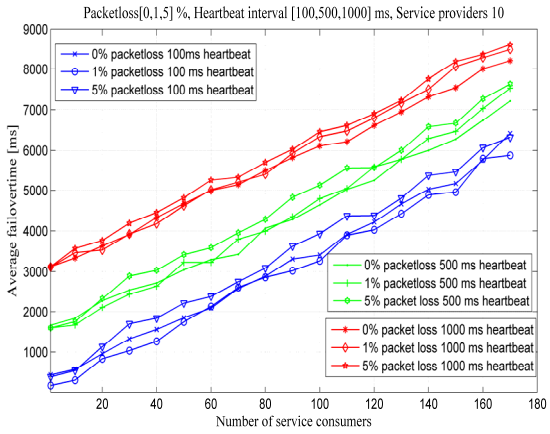


Figure 4. Average failover time with different rates of packet loss and with different redundancies.

Results in Figure 4 indicate that by selecting an appropriate heartbeat interval monitoring service can detect failures quickly which reduces failure detection time. As failure detection time is reduced, failover time is reduced as well and service availability is increased. The results indicate that service provider can handle 130 service consumers with 5 s's failover time which can be acceptable for non-critical services and if the requirements of availability and high availability are not high for a specific system.

We have also analyzed the impact of packet loss on service provider availability in these measurements. Results in Figure 4, indicate that 1 % packet loss has insignificant impact on service provider availability and 5 % packet loss can reduce service provider availability, but the difference is not very high. The results indicate that services which can tolerate packet loss to some extent can be accommodated by using the model. Results indicate that sustainable work load can be identified by using the model. By avoiding peak load on the system, service provider availability can be increased.

V. CONCLUSION

Applicability of service-oriented architecture is increasing in safety critical systems, telecommunication systems and business systems. Requirements of availability and high availability for every application area are different. Even the best practices cannot be utilized properly to fulfil the requirements. The solutions for increasing availability, such as machine virtualization, clusters and group communications systems are not very much applied within the domains of SOA. In this paper, a SOA-based model has been proposed and monitoring is used for increasing service availability. Clusters, machine virtualization, group communication systems and middleware based solutions can increase availability but they can also increase complexity, performance overhead, installation requirements and maintenance costs due to which they are not chosen in the proposed approach. In the paper, it is investigated that how different network conditions can impact or reduce service availability. Proposed SOA model focuses on reducing failure detection time by using heartbeat mechanism. Heartbeat mechanism is chosen for more accuracy in failure detections in comparison to the timeouts approach. In case of timeouts, failure is detected once and with heartbeat mechanism failure is ensured repeatedly. Experimental results show the effectiveness of the approach and indicate that by using heartbeat mechanism failures can be detected earlier than the timeout approach. Results indicate that a small heartbeat interval can reduce failure detection time and failover time and by selecting an optimal heartbeat interval service availability can be increased. Availability is also increased by adding redundancy as a redundant system can cover more failures than a non-redundant system. Results indicate that redundancy does not reduce performance and it can be used according to requirement. The next step in the research work is to extend the model with redundant monitoring services as a single monitoring service can become a single point of failure for the system. Diverse monitoring services can be introduced in model to avoid failures of same kind. Model can be extended by analyzing availability of service compositions or by analyzing availability of asynchronous services. Also, a middleware can be added to the model and availability can be analyzed for middleware based service-oriented systems.

 ISBN: 978-1-61208-263-9

REFERENCES

[1] Reference Model for Service Oriented Architecture 1.0, OASIS Committee Specification 1, Aug 2006.

[2] OASIS, "Advancing open standards for information society", [retrieved: Feb, 2013] from http://www.oasis-open.org/.

[3] W. D. Yu and C. H. Ong, "A SOA-based Software Engineering Design Approach in Service Engineering," Proc. IEEE International Conf. on e-Business Engineering (ICEBE), China, Oct. 2009, pp. 409 - 416.

[4] B. Li, "Research and application of SOA standards in the integration on web services," Proc. 2nd International Workshop on Education Technology and Computer Science (ETCS), China, vol. 2, Mar. 2010, pp. 492–495.

[4] M. Chen, C. Wu, and k. Wu, "Staging adjust service pool to assure availability in SOA title," Proc. International Conf. on Complex, Intelligent, and Software Intensive Systems (CISIS), Korea, Jun. 2011,pp. 409-413.

[5] J. Qiu, J. A. Pershing, Y. Li, L. Xie, J. Luo, and Y. Chen, "Availability weak point analysis over an SOA deployment framework," IEEE Symposium on Network Operations and Management (NOMS), Brazil, Apr. 2008, pp. 473 - 480.

[6] J. A Pershing, L. Xie, J. Luo, Y. Li, and Y. Chen, "A methodology for analyzing availability weak points in SOA deployment frameworks," IEEE Transactions on Network And Service Management (TNSM), vol. 6, Mar. 2009, pp. 31-44.

[7] J. Niemöller, I. Fikouras, K. Vandikas, R. Levenshteyn, R. Quinet, and E. Freiter, "Blending the telecommunication domain with Web 2.0 services," Proc. 14th International Conf. on Intelligence in Next Generation Networks (ICIN), Berlin, Oct. 2010, pp. 1-6.

[8] J. He, E. Castro-Leon, and M. Chang, "Scaling down SOA to small businesses," Proc. IEEE International Conf. on Service-Oriented Computing and Applications (SOCA), Jun. 2007, pp. 99 - 106.

[9] J. Fenn, A. Brown, and C. Menon, "Issues and considerations for a modular safety certification approach in a service-oriented architecture," Proc. 5th IET International Conf. on System Safety, United kingdom, Oct. 2010, pp.1-6.

[10] M. Haberkorn, and K. Trivedi, "Availability monitor for a software based system," Proc. 10th IEEE High Assurance Systems Engineering Symposium (HASE), USA, Nov. 2007, pp. 321–328.

[11] A. Apon and L. Wilbur,"Ampnet - a highly available cluster interconnection network," 17th International Symposium on Parallel and Distributed Processing (IPDPS), France, Apr. 2003, pp. 201.2.

[12] M. Yin, "Assessing availability impact caused by switchover in database failover," Proc. Annual Reliability and Maintainability Symposium (RAMS), USA, Jan. 2009, pp. 401 - 406.

[13] T. G. Papaioannou, N. Bonvin, and K. Aberer, "An economic approach for scalable and highly-available distributed applications," Proc. IEEE 3rd International Conf. on Cloud Computing, USA, Jul. 2010, pp. 498 - 505.

[14] R. Levenshteyn, R. Quinet, J. Niemöller, I. Fikouras, K. Vandikas, and E. Freiter, "Blending the telecommunication domain with web 2.0 services," Proc. 14th International Conf. on Intelligence in Next Generation Networks (ICIN), Berlin, Oct. 2010, pp.1-6.

[15] S. Ahlfeld, S. Schulte, J. Eckert, A. Papageorgiou, T. Krop, and R. Steinmetz, "Enhancing availability with self-organization extensions in a SOA platform," Proc. 5th International Conf. on Internet and Web Applications and Services (ICIW), Barcelona, May. 2010, pp. 161 - 166.

[16] A. Grzech and P. Swiatek, "Complex services availability in service oriented systems," Proc. 21st International Conf. on Systems Engineering (ICSEng), USA, Aug. 2011, pp. 227 - 232.

[17] C. Lin, X. Zhang, and X. Kong, "Model-driven dependability analysis of virtualization systems," Proc. 8th IEEE/ACIS International Conf. on Computer and Information Science, China, Jun. 2009, pp. 199 - 204.

[18] Y. Krasny, A. Krits, E. Farchi, G. Kliot, and R. Vitenberg, "Effective testing and debugging techniques for a Group Communication System," Proc. International Conf. on Dependable Systems and Networks (DSN), Japan, Jun. 2005, pp. 80-85.

[19] H. Y. Chan, B. Y. Ooi, and Y. Cheah, "Dynamic service placement and redundancy to ensure service availability during resource failures," Proc. International Symposium in Information Technology (ITSim), Kuala Lumpur, Jun. 2010, pp. 715 - 720.

[20] W. D. Grover and A. Sack, "High availability survivable networks: When is reducing MTTR better than adding protection capacity?," Proc. 6th International Workshop on Design and Reliable Communication Networks (DRCN), France, Oct. 2007, pp. 1-7.

[21] L.Yue, Y. Shengsheng, G. Hui, and Z. Jingli, "Design of a dual-computer cluster system and availability evaluation," Proc. IEEE International Conf. on Networking, Sensing and Control (ICNSC), Taiwan, Mar. 2004, pp. 355 - 360.

[22] R. Wolski, J. S. Plank, and M. Allen, "The effect of timeout prediction and selection on wide area collective operations," Proc. IEEE International Symposium on Network Computing and Applications (NCA), USA, Oct. 2001,pp. 320 – 329 .

[23] K. P. Birman, "Reliable Distributed Systems Technologies, Web Services and Applications," Springer, 2005.

[24] A. Q. Gates, N. Delgado, and S. Roach, "A taxonomy and catalog of runtime software-fault monitoring tools," Proc. IEEE Transactions on Software Engineering, vol. 30, Dec. 2004, pp. 859 - 872.

[25] S. Han and K. G. Shin, "Experimental evaluation of failure-detection schemes in real-time communication networks," Proc. 27th Annual International Symposium on Fault-Tolerant Computing, USA, Jun 1997, pp. 122 - 131.

[26] P. Merson, L. O'Brien Lero, and L. Bass, "Quality attributes for service-oriented architectures," Proc. Int. Workshop Systems Development in SOA Environments (SDSOA), USA, May 2007, p. 3.

[27] WANem, "Wide Area Network Emulator," [retrieved: Feb 14, 2013] from http://wanem.sourceforge.net/.

[28] Glassfish homepage, [retrieved: Feb 16, 2013] from http://glassfish.java.net/.

[29] jUDDI, "An Apache Project Homepage," [retrieved: Feb 11, 2013] from http://juddi.apache.org/.

[30] Apache Tomcat, [retrieved: Feb 14, 2013] from http://tomcat.apache.org/.

[31] MySQL, [retrieved: Feb 22, 2013] from http://www.mysql.com/.

 ISBN: 978-1-61208-263-9