International Journal on Advances in Networks and Services

IARIA

2008 vol. 1 nr. 1

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

Sponsored by IARIA
www.iaria.org

International Journal On Advances in Networks and Services
Volume 1, Number 1, 2008

- Christian Emig, University of Karlsruhe, Germany
- Geoffrey Fox, Indiana University, USA
- Mario Freire, University of Beira Interior, Portugal
- Thomas Y Kwok, IBM T.J. Watson Research Center, USA
- Zoubir Mammeri, IRIT – Toulouse, France
- Bertrand Mathieu, Orange-ftgroup, France
- Mihhail Matskin, NTNU, Norway
- Guadalupe Ortiz Bellot, University of Extremadura Spain
- Dumitru Roman, STI, Austria
- Monika Solanki, Imperial College London, UK
- Pierre F. Tiako, Langston University, USA
- Weiliang Zhao, Macquarie University, Australia

Wireless and Mobile Communications

- Habib M. Ammari, Hofstra University - Hempstead, USA
- Thomas Michael Bohnert, SAP Research, Switzerland
- David Boyle, University of Limerick, Ireland
- Xiang Gui, Massey University-Palmerston North, New Zealand
- Qilian Liang, University of Texas at Arlington, USA
- Yves Louet, SUPELEC, France
- David Lozano, Telefonica Investigacion y Desarrollo (R&D), Spain
- D. Manivannan (Mani), University of Kentucky - Lexington, USA
- Jyrki Penttinen, Nokia Siemens Networks - Madrid, Spain / Helsinki University of Technology, Finland
- Radu Stoleru, Texas A&M University, USA
- Jose Villalon, University of Castilla La Mancha, Spain
- Natalija Vlajic, York University, Canada
- Xinbing Wang, Shanghai Jiaotong University, China
- Qishi Wu, University of Memphis, USA
- Ossama Younis, Telcordia Technologies, USA

Sensors

- Saied Abedi, Fujitsu Laboratories of Europe LTD. (FLE)-Middlesex, UK
- Habib M. Ammari, Hofstra University, USA
- Steven Corroy, Philips Research Europe – Eindhoven, The Netherlands
- Zhen Liu, Nokia Research – Palo Alto, USA
- Winston KG Seah, Institute for Infocomm Research (Member of A*STAR), Singapore
- Peter Soreanu, Braude College of Engineering - Karmiel, Israel
- Masashi Sugano, Osaka Prefecture University, Japan
- Athanasios Vasilakos, University of Western Macedonia, Greece
- You-Chiun Wang, National Chiao-Tung University, Taiwan
- Hongyi Wu, University of Louisiana at Lafayette, USA

- Dongfang Yang, National Research Council Canada – London, Canada

Underwater Technologies

- Miguel Ardid Ramirez, Polytechnic University of Valencia, Spain
- Fernando Boronat, Integrated Management Coastal Research Institute, Spain
- Mari Carmen Domingo, Technical University of Catalonia - Barcelona, Spain
- Jens Martin Hovem, Norwegian University of Science and Technology, Norway

Energy Optimization

- Huei-Wen Ferng, National Taiwan University of Science and Technology - Taipei, Taiwan
- Qilian Liang, University of Texas at Arlington, USA
- Weifa Liang, Australian National University-Canberra, Australia
- Min Song, Old Dominion University, USA

Mesh Networks

- Habib M. Ammari, Hofstra University, USA
- Stefano Avallone, University of Napoli, Italy
- Mathilde Benveniste, Wireless Systems Research/En-aerion, USA
- Andreas J Kassler, Karlstad University, Sweden
- Ilker Korkmaz, Izmir University of Economics, Turkey //editor assistant//

Centric Technologies

- Kong Cheng, Telcordia Research, USA
- Vitaly Klyuev, University of Aizu, Japan
- Arun Kumar, IBM, India
- Juong-Sik Lee, Nokia Research Center, USA
- Josef Noll, ConnectedLife@UNIK / UiO- Kjeller, Norway
- Willy Picard, The Poznan University of Economics, Poland
- Roman Y. Shtykh, Waseda University, Japan
- Weilian Su, Naval Postgraduate School - Monterey, USA

Multimedia

- Laszlo Boszormenyi, Klagenfurt University, Austria
- Dumitru Dan Burdescu, University of Craiova, Romania
- Noel Crespi, Institut TELECOM SudParis-Evry, France
- Mislav Grgic, University of Zagreb, Croatia
- Hermann Hellwagner, Klagenfurt University, Austria
- Polychronis Koutsakis, McMaster University, Canada
- Atsushi Koike, KDDI R&D Labs, Japan
- Chung-Sheng Li, IBM Thomas J. Watson Research Center, USA
- Parag S. Mogre, Technische Universitat Darmstadt, Germany
- Eric Pardede, La Trobe University, Australia

- Justin Zhan, Carnegie Mellon University, USA

International Journal On Advances in Networks and Services
Volume 1, Number 1, 2008

Foreword

Finally, we did it! It was a long exercise to have this inaugural number of the journal featuring extended versions of selected papers from the IARIA conferences.

With this 2008, Vol. 1 No.1, we open a long series of hopefully interesting and useful articles on advanced topics covering both industrial tendencies and academic trends. The publication is by-invitation-only and implies a second round of reviews, following the first round of reviews during the paper selection for the conferences.

Starting with 2009, quarterly issues are scheduled, so the outstanding papers presented in IARIA conferences can be enhanced and presented to a large scientific community. Their content is freely distributed from the www.iariajournals.org and will be indefinitely hosted and accessible to everybody from anywhere, with no password, membership, or other restrictive access.

We are grateful to the members of the Editorial Board that will take full responsibility starting with the 2009, Vol 2, No1. We thank all volunteers that contributed to review and validate the contributions for the very first issue, while the Board was getting born. Starting with 2009 issues, the Editor-in Chief will take this editorial role and handle through the Editorial Board the process of publishing the best selected papers.

Some issues may cover specific areas across many IARIA conferences or dedicated to a particular conference. The target is to offer a chance that an extended version of outstanding papers to be published in the journal. Additional efforts are assumed from the authors, as invitation doesn't necessarily imply immediate acceptance.

This particular issue covers papers invited from those presented in 2007 and early 2008 conferences. The papers cover end-to-end QoS and performance, and their prediction, especially in VoIP applications. Additionally, architectural and mechanisms for fault tolerant, ambient and adaptive large-scale systems, as well as content-addressable networks are presented.

We hope in a successful launching and expect your contributions via our events.

First Issue Coordinators,
Jaime Lloret, Universidad Politécnica de Valencia, Spain
Pascal Lorenz, Université de Haute Alsace, France
Petre Dini, Cisco Systems, Inc., USA / Concordia University, Canada

International Journal On Advances in Networks and Services
Volume 1, Number 1, 2008

CONTENTS

A Solution for QoS Support in Wireless Ad hoc Networks

Leila Boukhalfa[1, 3], Pascale Minet[2], Serge Midonnet[1, 3]

[1] *Université Marne la Vallée*
77454 Marne-la-Vallée
France
leila.boukhalfa@univ-mlv.fr

[2] *INRIA, Rocquencourt*
78153 Le Chesnay Cedex
France
pascale.minet@inria.fr

[3] *ESIGETEL*
1 rue du Port de Valvins
77210 Avon, France
serge.midonnet@esigetel.fr

Abstract

Mobile ad hoc networks become more popular as devices and wireless communication technologies are became widespread and ubiquitous. With the expanding range of applications of MANETs, supporting quality of services (QoS) in these networks is becoming a real need. This paper provides a solution for QoS support taking into account radio interferences. We note that, because of the ad hoc networks characteristics, we cannot provide a hard quality of service to QoS flows, but only provide a service differentiation between different flow types. This QoS support is based on the OLSR routing protocol and the CBQ scheduling. Simulation results show that flows with QoS requirements receive the requested bandwidth and Best Effort flows share the remaining bandwidth. Moreover, mobility is supported.

Keywords: MANET, QoS, OLSR, CBQ, routing protocol, quality of service.

1. Introduction

A Mobile Ad hoc NETwork (MANET) is an autonomous system of mobile nodes connected by wireless links. It is self-organizing, rapidly deployable and requires no fixed infrastructure. Ad hoc networks have known a great success and now, they are opening up to civilian applications having requirements of Quality of Service (QoS) [1]. Hence, achieving QoS [4] in MANET corresponds to a real need. The QoS, requested from the network, could be defined in terms of one or a set of parameters such as delay, bandwidth, packet loss, delay and jitter. MANET networks are faced with specific constraints: a) the limited bandwidth because of the reduced available radio resources, b) the highly dynamic topology because of versatile radio propagation and nodes mobility, c) the power constraints because network nodes can rely on battery power for energy. These MANET specificities make it difficult to achieve QoS in these networks.

OLSR [6] is an optimization of the wired link state routing protocol OSPF [7] for MANET. Its innovation lies in the fact that it uses the MultiPoint Relay (MPR) technique. The MPRs of a node are a subset of its one hop neighbors that enables it to reach (in terms of radio range) all its two-hop nodes. The MPRs technique results in the reduction of the control packet size (each node declares only the links with its one hop neighbors which selected it as MPR), and reduces the number of retransmissions when flooding control messages in the network: only the MPRs of the sender forward its packets.

The scheduling policy adopted in our solution is inspired from the one used in wired networks. We recall that our aim is the QoS support [3] in ad hoc networks in order to differentiate services between different traffic classes. One solution is to provide a minimum part of the requested bandwidth to different traffic classes. This means that the medium capacity must be shared between traffic classes. We are then interested in the CBQ scheduler [5] (Class Based Queueing) and we have extended it to the wireless environment. CBQ aims at carrying out two goals. The first one is that each class should be able to receive roughly its allocated bandwidth. The secondary one is that when some class is not using its allocated bandwidth, the distribution of the excess bandwidth among the other classes should not be arbitrary, but should be done according to their relative allocations. Hence, WCBQ leads to good resource utilization.

In this paper, we show how to take into account radio interferences to provide the bandwidth requested by QoS flows. The remainder of this paper is organized

as follows. In section 2, we discuss the impact of interferences on the QoS, and describe the QoS components constituting our solution to support QoS in ad hoc networks. The performance evaluation of our solution is given in section 3, followed by a conclusion in section 4.

2. Proposed solution

In this section we present our solution for QoS support taking into account interferences generated by flows present in the network.

2.1. QoS and interferences

Because of the shared medium access in ad hoc networks, a packet generated by a mobile node is physically received by all nodes in the transmission range of the sender. Consequently, interferences are generated when neighboring nodes are transmitting at the same time. The presence of interferences makes quality of service support much more complex in wireless networks than in wired networks. For example, interferences make bandwidth reservation in a wireless environment an NP-complete problem [9], whereas it is polynomial in wired networks. These interferences can reduce significantly the capacity of the network.

Let us consider a scenario of 6 nodes and one flow f_1. The flow f_1 requests a bandwidth of 100kb/s. f_1 is generated by node N_0 toward node N_3 (Figure 1). To illustrate the interference phenomenon, we measure the consumed bandwidth at the MAC level on each node of the network.

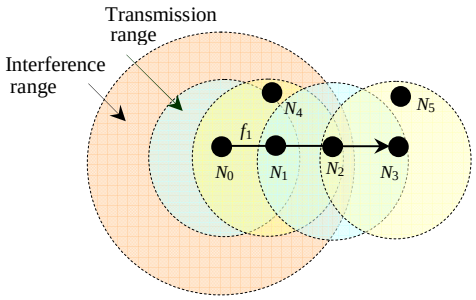


Figure 1. Interference phenomenon

We note that (see Figure 2) flow f_1 has consumed 281kb/s on N_2. It represents nearly three times the bandwidth requested by f_1. Indeed, node N_2 is disrupted by any packet of flow f_1, once when N_0 transmits, because N_2 is in the interference area of N_0, a second time when N_1 transmits because $N2$ is in the transmission range of N_1, and a third time when the node itself transmits. As for node N_5, it does not belong

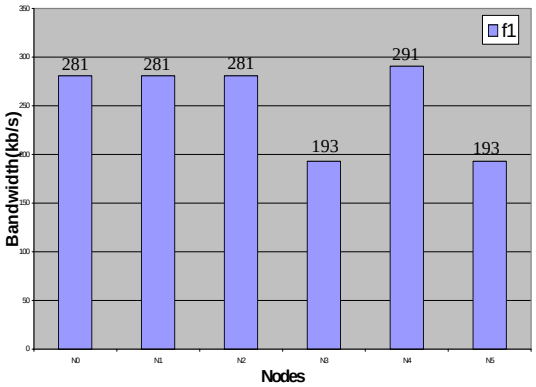


Figure 2. Measured consumed Bandwidth

to the route, the bandwidth consumed by f_1 on this node is nearly twice the bandwidth requested by f_1. Indeed, N_5 is in the interference area of N_1 and N_2. These two nodes belong to the route of flow f_1. Consequently, N_5 is disrupted each time one of these nodes transmits. We conclude that because of the interferences, a flow consumes more bandwidth than it requests. This illustrates the necessity to take into account the interferences in all solutions managing quality of service with bandwidth requirements. In the following, we assume that interferences caused by a transmitting node are limited to two hops.

Providing quality of service in ad hoc networks therefore should be interference aware [10]. For this goal, we consider an admission control which takes into account interferences induced by flows present on the network. Also, the routing protocol considered in our solution takes into account interferences to provide routes with the requested quality of service. QoS routing needs QoS signaling to collect information related to QoS. Besides these components, other QoS components can be used to provide the quality of service requested by QoS flows. Hence, the QoS architecture we propose in the next section.

2.2. QoS components

In [1] we have presented a general QoS architecture and defined its different components illustrated in Figure 3. Among these components we are interested in the five following components:

- *QoS model* specifies the architecture in which services can be provided as well as the necessary mechanisms such as classification. The QoS model directly influences the functionality of the other components.
- *Admission control* is the mechanism that results in the acceptance or rejection of a new flow according to *(i)* the available resources on the path taken by this flow and *(ii)* the QoS requirements of this flow.

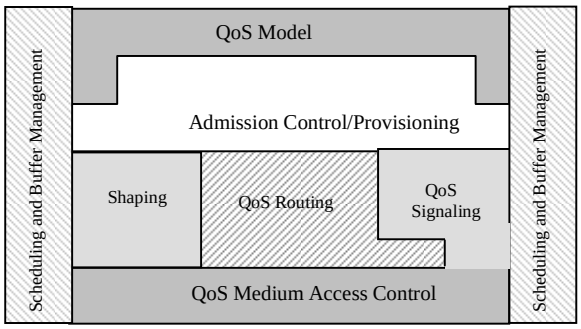


Figure 3. QoS Components

- *Admission control* is the mechanism that results in the acceptance or rejection of a new flow according to *(i)* the available resources on the path taken by this flow and *(ii)* the QoS requirements of this flow.
- *QoS routing* aims to find routes with sufficient resources to meet the application requirements but does not reserve resources. In our solution, QoS routing is based on an extension of OLSR.
- *QoS signaling* is used to propagate QoS control information in the network, as well as to generate the QoS reports that indicate the effectively measured QoS.
- *The scheduler* determines the message transmission order according to the priorities given to QoS classes. In our solution, it is based on CBQ.

2.3. Adopted assumptions

2.3.1. QoS MAC. In the ideal case, a medium access protocol managing QoS makes it possible to guarantee, to the non-pre-emptive effect close, that the transmitted packet is the packet having the highest priority among all packets waiting for transmission. Let us notice that the IEEE802.11e protocol does not satisfy this property. It guarantees only that the average delay of flows with higher priority is weaker than that of flows with lower priority.

The MAC layer must be also able to provide information allowing to calculate the available bandwidth on a node. The QoS management on the MAC level allows to obtain a better services differentiation as it shown in [11] for the IEEE 802.11 protocol where flows with higher priority obtain weaker average delays.

The solution we propose does not require a QoS MAC to behave properly. In the performance evaluation reported in section 3, we use the IEEE 802.11b MAC protocol which is more currently used for the MANET networks but not yet offering QoS management. However, the performances of our solution will be improved by a QoS MAC.

2.3.2. Interference model. The proposed solution takes in consideration the interferences to *h* hops:
- at the sender node: the receptions of the nodes located at less than *h* hops of the sender are disturbed when this sender transmits.
- at the receiver node: the simultaneous transmissions of nodes located at less than *h* hops of the receiver prevent the good reception.

Consequently, two senders located at less than 2*h* hops can interfere with each other. The interference range is said to be 2*h*. In this paper we take an interference range 2. This assumption is generally adopted in ad hoc networks literature.

2.3.3. Computation of the needed bandwidth. We note that, because of the interferences, a flow *f* requiring a bandwidth $B(f)$ at the application level, really consumes a bandwidth $B_{real}(f)$ at the MAC level, higher than $B(f)$. This is true on any route node and on any neighbor node of a route node. That is due to the interferences. We show below, how to evaluate the bandwidth really consumed by a flow.

In our solution, the route is supposed to be such that a route node belongs to the interference zone of, at most, its two predecessors and, at most, its two successors; hence the value of 5 in formula (1).

$$B_{real}(f) \leq coef.\min(5, hop).B(f) \qquad (1)$$

Where:
hop is the number of hops from the source to the destination.
coef is a coefficient allowing to take into account the overhead induced by the MAC acknowledgement and the headings of the protocols: physical, MAC, IP and UDP. The *coef* also depends on the packet size. For example, for a QoS flow whose packet size is equal to 500 bytes, and with a medium of 2Mb/s, the value of *coef* is equal to 1.144.

We note that the value $coef.\min(5, hop).B(f)$ corresponds to the maximum bandwidth which a flow can consume on a node *i.e.*, the bandwidth really consumed by a flow on any node is never higher than $coef.5.B(f)$ with our assumptions.

The formula can appear too simple but any more sophisticated method wanting to take into account all the exact interferences requires a transmission overhead without allowing an exact evaluation as shown in the following example:

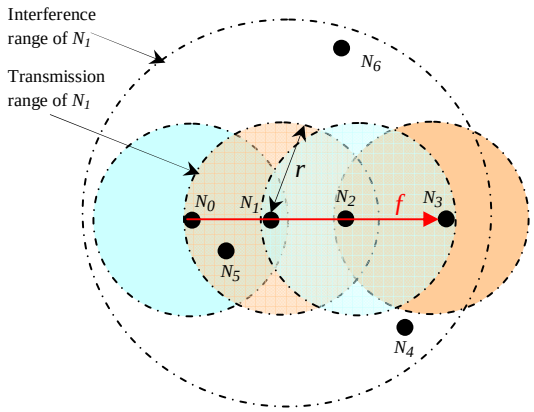


Figure 4. Scenario of 7 nodes

Node N_6 is located in the interference range of node N_1 but not in its transmission range *i.e.*, $r < d(N_1, N_6) < 2r$ where $d(N_1, N_6)$ is the distance between the two nodes N_1 and N_6 and r is the transmission range. In no case, the node N_1 can detect the presence of node N_6, because there is no intermediate node belonging to the transmission range of N_1 and of N_6 making it possible each of the two nodes to detect the presence of the other. Consequently, the evaluation of the bandwidth really consumed by flow f does not consider the disturbances induced by node N_1 on node N_6.

2.4. QoS model

2.4.1. Considered flow types. We consider three flow types:
- QoS flows having QoS requirements expressed in terms of bandwidth,
- QoS flows having QoS requirements expressed in terms of delay,
- Best Effort (BE) flows having no specific QoS requirements.

In our solution, we adopt the following decreasing priority order of flows:
Control flows > QoS flows with delay constraints > QoS flows with bandwidth constraints > Best Effort flows.

2.4.2. Bandwidth provisioning. To share the medium bandwidth between QoS flows and BE flows, we will use provisioning. The provisioning consists in reserving a percentage of the nominal bandwidth to each flow type. We consider then:

- $ProvQoS^N$: provisioning of QoS flows on node N.
- $ProvBE^N$: provisioning of BE flows on node N.

We assume that $ProvQoS^N$ and $ProvBE^N$ are global parameters of the network and they are identical on all network nodes. For an effective use of the network resources, we allow each flow type to exceed its provisioning. In this case, the bandwidth not used by one flow type can be used by the other, and when necessary, each flow type can recover its share of bandwidth used by the other one. Moreover, QoS flows can requisition the bandwidth used by BE flows. The reverse is not true.

In our solution, only QoS flows can recover their available bandwidth used by BE flows. BE flows must not recover their available bandwidth used by QoS flows, to avoid the deterioration of the quality of service of QoS flows already admitted. However, if a new QoS flow arrives when the QoS available bandwidth it needs entirely or partially, is used by BE flows, this flow can recover the bandwidth it needs from BE flows.

2.5. Admission control

Let us recall that, the admission control decides to accept a new flow f if and only if:
- the QoS of already accepted flows is not compromised;
- the QoS required by the flow f can be satisfied.

We present below the rules of the admission control. The admission control is performed for the two flow types QoS and BE:

- In our solution, the admission control of QoS flows having bandwidth requirements takes into account the interferences *i.e.*, a flow will be accepted only if the interferences that it generates are acceptable for already accepted flows and the QoS it will receive is compatible with that required taking into account the interferences generated by other flows.
- BE flows do not require any constraint, but an admission control is necessary to verify that they do not exceed their available bandwidth.

Let us consider the following notations:
$BQoS_a^N$: available QoS bandwidth on node N.
BBE_a^N: available BE bandwidth on node N.
$BQoS_u^N$: QoS bandwidth used on node N.
BBE_u^N: BE bandwidth used on node N.
$ProvQoS^N$: provisioning granted to QoS flows on node N.
$ProvBE^N$: provisioning granted to BE flows on node N.

More particularly, the admission control consists in checking:

- For each route node N (except the destination) and for each node M at a distance lower than or equal to two hops of N :

  - *For a QoS flow f:*

    ❶ $B^{N}_{real}(f) \leq BQoS_a^{N}$

    ❷ $B^{M}_{real}(f) \leq BQoS_a^{M}$

  - *For a BE flow f*

    ❶ $B^{N}_{real}(f) \leq BBE_a^{N}$

    ❷ $B^{M}_{real}(f) \leq BBE_a^{M}$

- For the destination node D

  - *For a QoS flow f:*

    $B^{D}_{real}(f) \leq BQoS_a^{D}$

  - *For a BE flow f*

    $B^{D}_{real}(f) \leq BBE_a^{D}$

Where:

$BQoS_a^{N} = \max (ProvQoS^{N} - BQoS_u^{N}, available^{N})$

$BBE_a^{N} = \max (ProvBE^{N} - BBE_u^{N}, available^{N})$

$Available^{N} = (ProvQoS^{N} - BQoS_u^{N}) + (ProvBE^{N} - BBE_u^{N})$

2.6. QoS routing

Routing protocol OLSR with QoS aims at finding:

- for QoS flows, the shortest route satisfying the requested bandwidth.
- for BE flows, the shortest route.

The OLSR extension which we propose consists in: (*i*) modifying the choice of the multipoint relay and (*ii*) adding information in control messages Hello and TC, information necessary to the admission control and the QoS routing. We also, present the rules of admission control adapted to this extension.

2.6.1. Selection of MPRs according to the available bandwidth. In an ad hoc network, the native OLSR protocol provides an optimal route to any destination in the network. This route is optimal in terms of number of hops but does not take into account the requirements of QoS flows. For a QoS flow, we need to find a route which satisfies the required quality of service. However, the intermediate nodes of a requested route found by OLSR are MPR nodes. This is why we perform the MPR selection according to the QoS local available bandwidth denoted $BQoS_a$.

In the extension that we propose, multipoint relays are selected so as to reach the two hop neighbors through a one-hop neighbor with the maximum QoS available bandwidth ($BQoS_a$) *i.e.*, if a two-hop neighbor can be reached by several one-hop neighbors then the one having the larger $BQoS_a$ is selected. Because we have taken into account the bandwidth to select the MPR nodes, the MPRs are called MPRBs.

2.6.2. Evaluation of the bandwidth used by QoS and BE. The QoS used bandwidth (or the BE used bandwidth) on a given node N is equal to the QoS (or BE) load on N plus the sum of QoS (or BE) loads on the one or two hop neighbor nodes of N:

$$BQoS_u^{N} = \left(QoS\_ch^{N} + \sum_{V} QoS\_ch \right) \cdot coef \cdot MC$$

$$BBE_u^{N} = \left(BE\_ch^{N} + \sum_{V} BE\_ch \right) \cdot coef \cdot MC$$

Where:

V: the one and two hop neighbor set of node N

MC: medium capacity

$coef$: a coefficient depending on packet size. It takes into account the overhead generated by MAC acknowledgement and protocol headers: physical, MAC, IP and UDP. The $coef$ value is identical to that used for the evaluation of the bandwidth really consumed by a flow.

2.6.3. Route selection. From its neighbor and topology tables, each node builds its routing table using the Dijkstra algorithm. The intermediate nodes of routes toward each destination are MPRB nodes.

A. Route selection for QoS flows

When a new QoS flow f is generated on a source node, this source node selects the shortest route offering the demanded QoS by applying the Dijkstra algorithm on a copy of the topology and the neighbor tables in which only nodes offering the demanded QoS are present.

The admission control of a new QoS flow is performed on the source node. According to the information it maintains from Hello and TC messages, the source cannot verify correctly the second condition of admission control seen in section 2.5 because it does not know the $BQoS_a$ of all neighbors at one and two hops of each node belonging to the route.

In our solution, a QoS flow f is accepted if and only if for each node N on the route, f is supported by (*i*) the node N and (*ii*) by any node up to two hops from N, if N is not the destination.

If the flow is not accepted on one of the route nodes or on one of the neighbors of one of the route nodes, the flow is rejected. Otherwise, when the route satisfying the requested QoS is found, it will be fixed

in order to perform source routing *i.e.*, the list of node route addresses will be included in the header of flow packets. In this way, all packets of this flow will follow the same route to reach the destination. This route is recalculated periodically to verify if there exists either a shorter route satisfying the QoS or a broken link.

B. Route selection for BE flows

Best effort flows are routed hop by hop and the admission control of these flows is performed locally on each route node and for each packet. Hence, when a new BE flow *f* is generated on a source node, this source node checks for each packet, if the destination node exists in its routing table. If the destination does not exist, the packet is rejected. Otherwise the node performs a local control admission for this packet to verify if the flow is supported by this node and by all its one and two hop neighbors. If so, the flow is transmitted toward the next node according to the routing table. We note that, for each packet of a new BE flow *f*, the admission control consists of verifying on each route node *N* that flow *f* is supported by (*i*) the node *N* and (*ii*) by any node up to two hops from *N*, if *N* is not the destination. This computation is done using BBE_{min}, the minimum available bandwidth for BE flow in the one and two hop neighborhood of *N*. It is computed from BBE_a values received in the Hello messages.

2.7. QoS signaling

We have extended the Hello and TC messages in order to convey the necessary information for QoS routing and admission control.

A Hello message, sent by a node, contains the following information:

- its address, its *QoS_ch*, its *BE_ch*, its $BQoS_a$ and its BBE_a.
- the address, the *QoS_ch*, the *BE_ch*, the $BQoS_a$ and the BBE_a of any one hop neighbor with the link status.

From the Hello messages, each node in the network can know the $BQoS_a$ of all its one and two hop neighbors. Thus, each node can select its MPRB set.

A TC message contains the following information:
- address of the TC sender,
- $BQoS_a$ of the TC sender,
- $BQoS_{min}$ which correspond to the minimum $BQoS_a$ of all the one and two hop neighbor of the TC sender,
- Address of the MPRB selectors,
- $BQoS_a$ of the MPRB selectors.

From the received TC messages, each node builds its topology table.

2.8. WCBQ scheduling

In a network, packet scheduling policy refers to the decision process used to select the next packet that will be transmitted. At present, many schedulers are used in wired networks such as First In First Out (FIFO), Stochastic Fair Queueing (SFQ), Fair Queueing (FQ), and CBQ. Whereas in wireless networks, only FIFO and PriQueue schedulers are used.

The scheduling policy adopted in our solution is inspired from the one used in wired networks. We recall that our aim is the QoS support in ad hoc networks in order to differentiate services between different traffic classes. One solution is to provide a minimum part of the requested bandwidth to different traffic classes. This means that the medium capacity must be shared between traffic classes. We are then interested in the CBQ scheduler [5] (Class Based Queueing), we have extended it to the wireless environment and we have called it WCBQ (Wireless CBQ). WCBQ inherits the three modules of CBQ which are:

- *Classifier*: it inserts packets ready to be sent by the node in the appropriate class queue.
- *Estimator*: it estimates the bandwidth used by each class in the appropriate time interval. This information is used to determine whether or not each class has received its allocated bandwidth.
- *Selector*: using the information from the estimator, it has to decide which class queue is allowed to send a packet. According to [4, 5], a selector should implement two mechanisms which are the general scheduler and the link sharing scheduler. The general scheduler is to be used to schedule the class queues if the allocated bandwidth for each class can meet the requirement. Otherwise, the link-sharing scheduler is used to adjust the transmission rates.

In [2], we have shown by means of simulations that WCBQ provides the following properties:

P1: it shares the node bandwidth between flows present on the node proportionally to their weight.

P2: it minimizes the standard deviation of the average bandwidth except for forwarded flows with low throughput.

P3: it minimizes the end-to-end delay except for forwarded flows with low throughput.

P4: it minimizes the standard deviation of the end-to-end delay for all flows.

Let us recall that, in our QoS model, we have considered three user flow types which are QoS flows having QoS requirements expressed in terms of bandwidth, QoS flows having QoS requirements expressed in terms of delay and Best Effort (BE) flows having no specific QoS requirements. To schedule

these three user flow types, we have combined the use of two schedulers which are WCBQ and Priority Queueing (PQ) in the following way (see Figure 6):

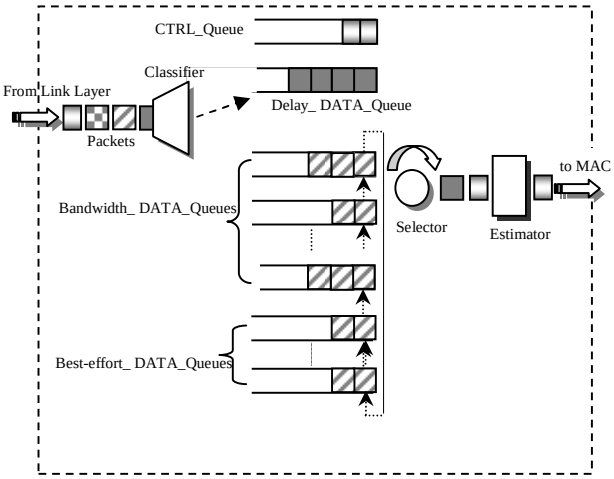


Figure 6. Coexistence of three user flow types

- CTRL_Queue is dedicated to control packets (e.g. routing packets). This queue has the highest priority, thus, it is served first *i.e.*, when a control packet arrives at the CTRL_Queue, either it is transmitted immediately, if there is no packet being transmitted or it is transmitted after the packet in the course of transmission.
- We have also attributed a Delay_DATA_Queue for traffic having delay requirements. This Queue has a lower priority than CTRL_Queue *i.e.*, a packet from Delay_DATA_Queue is transmitted if and only if there is no packet to transmit in CTRL_Queue.
- Bandwidth_DATA_Queues are reserved for traffic having bandwidth constraints.
- Best-Effort_DATA_Queues are reserved for BE flows.

Bandwidth_DATA_Queues and Best-Effort_DATA_Queues are managed according to WRR (Weighted Round Robin). Also, these queues, are served only if there is no packet in CTRL_Queue and no packet in Delay_DATA_Queue for transmission. Thus, with the combination of the two schedulers PQ and WCBQ, we can enable the three user flow types to coexist.

For WCBQ, we have calculated the weight $\phi(f_j)$ associated with each flow f_j present on the node N and requesting bandwidth $B(f_j)$, as follows:

$$\phi(f_j) = \frac{B(f_j)}{\sum_{i=1....n} B(f_i)}$$

Where n is the number of flows present on the node N having the same priority as f_j.

3. Performance evaluation

We now report performance evaluation of the QoS support described in the previous section.

3.1. Simulation parameters

The solution performance evaluation is carried out under the NS2 simulator [8]. The network simulator NS2 is an object-oriented, discrete event-driven network simulator. First, we consider an ad hoc network made up of 50 static nodes. The simulation parameters are summarized in the following table:

Table 1. Simulation parameters

Simulation	- simulation duration: 300s - Number of nodes: 50 - Flat area: 1000mx1000m - Traffic type: CBR - Packet size: 500 bytes
Routing protocol (OLSR)	- Source routing for QoS flows - Hop by hop routing for BE flows - Periodic routing table calculation for QoS flows (period 2s) - Hello period: 2s - TC period: 5s - Use of MPRB
MAC	- MAC protocol: IEEE802.11b - Throughput: 2Mb/s - No RTS/CTS messages
Radio	- Radio propagation model : TwoRayGround - Transmission range: 250m - Interference range: 500m

3.2. Fair sharing of bandwidth for BE flows and routes stability for QoS flows

In this section we show that, on a node, BE flows share the available bandwidth proportionally to their weight. In order to do this, we consider six QoS flows (f_1….,f_6) which obtain their requested bandwidth. Afterwards, we gradually introduce ten identical BE flows (f_7....., f_{16}) *i.e.*, same rate, same source and same destination. Each time we measure the bandwidth received by each flow present in the network. We provide also, the number of routes taken by each flow as well as the number of route changes during the simulation. Simulation results are given in Table 3. The source, the destination and the requested bandwidth of each flow are given in Table2.

Table 2. Flows parameters

Flow	Type	Requested bandwidth (kb/s)	Source	Destination
f_1	QoS	50	43	10
f_2	QoS	40	27	48
f_3	QoS	60	18	7
f_4	QoS	30	1	32
f_5	QoS	50	19	28
f_6	QoS	40	41	15
f_7	BE	20	38	12
f_8	BE	20	38	12
f_9	BE	20	38	12
f_{10}	BE	20	38	12
f_{11}	BE	20	38	12
f_{12}	BE	20	38	12
f_{13}	BE	20	38	12
f_{14}	BE	20	38	12
f_{15}	BE	20	38	12
f_{16}	BE	20	38	12

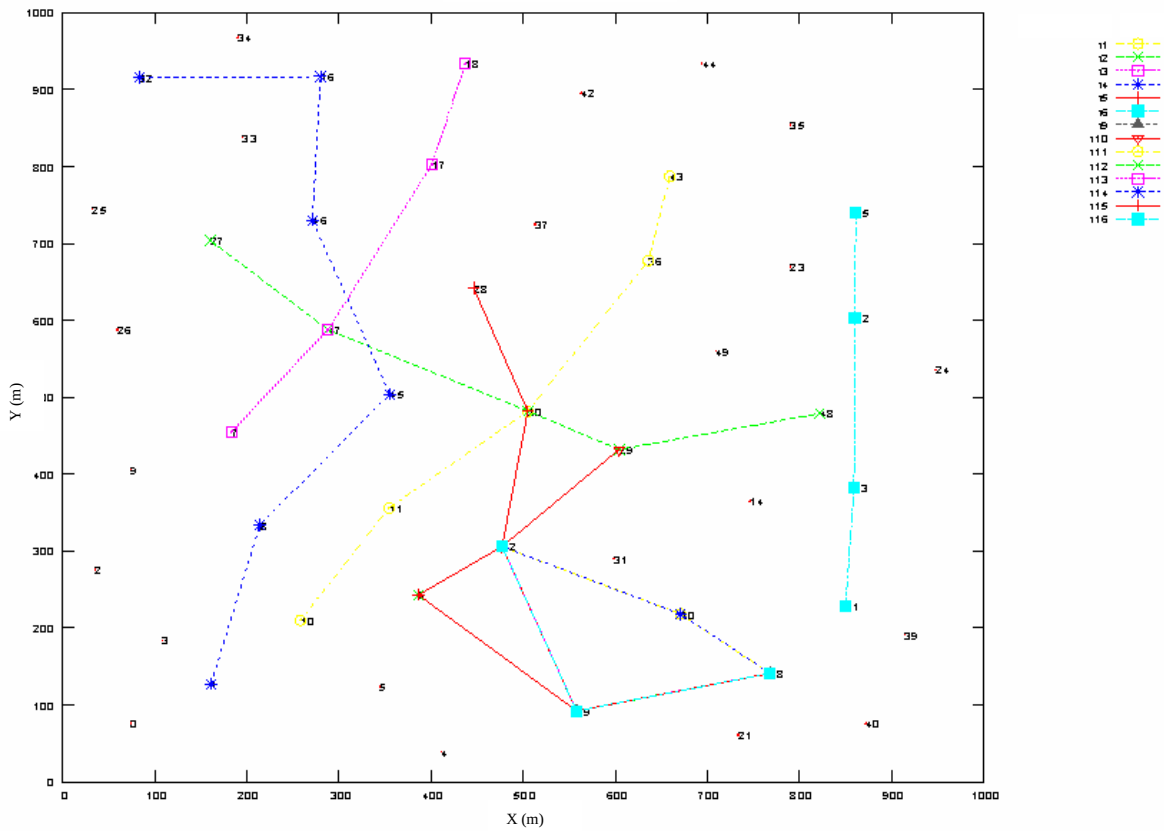


Figure 7. An ad-hoc network of 50 nodes

Table 3. Simulation results

Flows	Type	Requested bandwidth (kb/s)	Measured bandwidth (kb/s)	Route numbers	Number of route changes
f_1	QoS	50	49	1	0
f_2	QoS	40	38	1	0
f_3	QoS	60	60	1	0
f_4	QoS	30	29	1	0
f_5	QoS	50	48	1	0
f_6	QoS	40	40	1	0
f_7	BE	20	4	3	12
f_8	BE	20	4	3	11
f_9	BE	20	4	3	11
f_{10}	BE	20	4	3	11
f_{11}	BE	20	4	3	11
f_{12}	BE	20	4	3	10
f_{13}	BE	20	4	2	8
f_{14}	BE	20	4	3	12
f_{15}	BE	20	4	3	10
f_{16}	BE	20	4	3	10

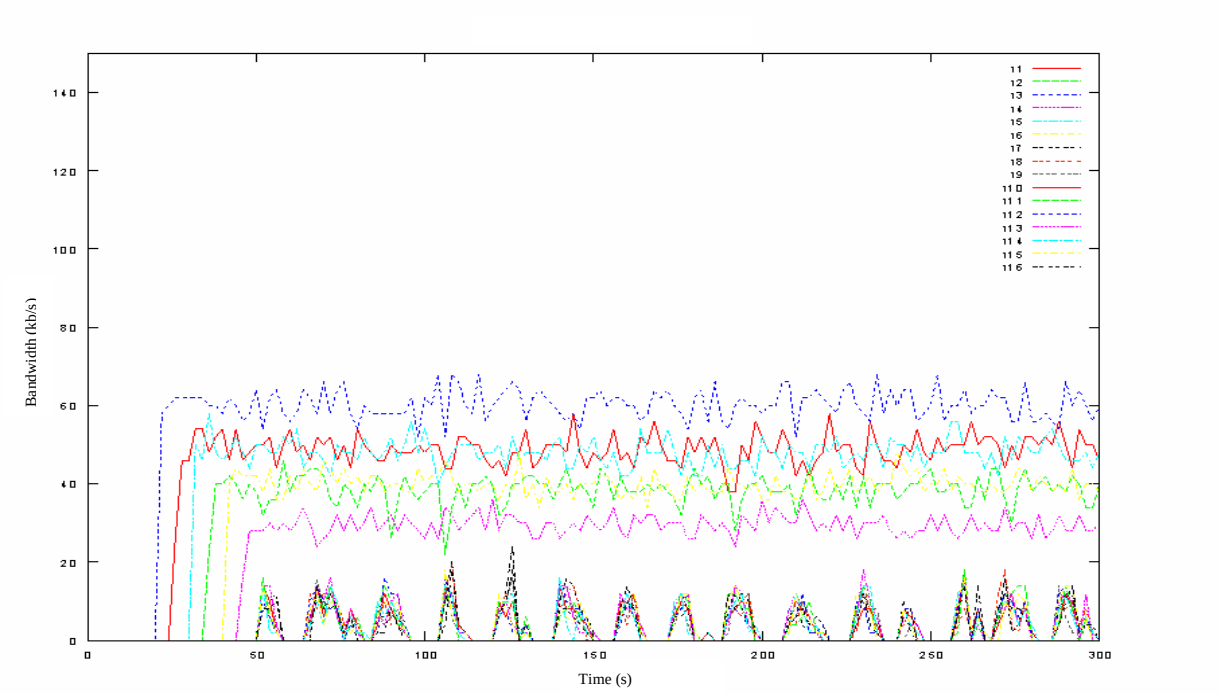


Figure 8. Measured instantaneous bandwidth for 6 QoS flows and 10 BE flows

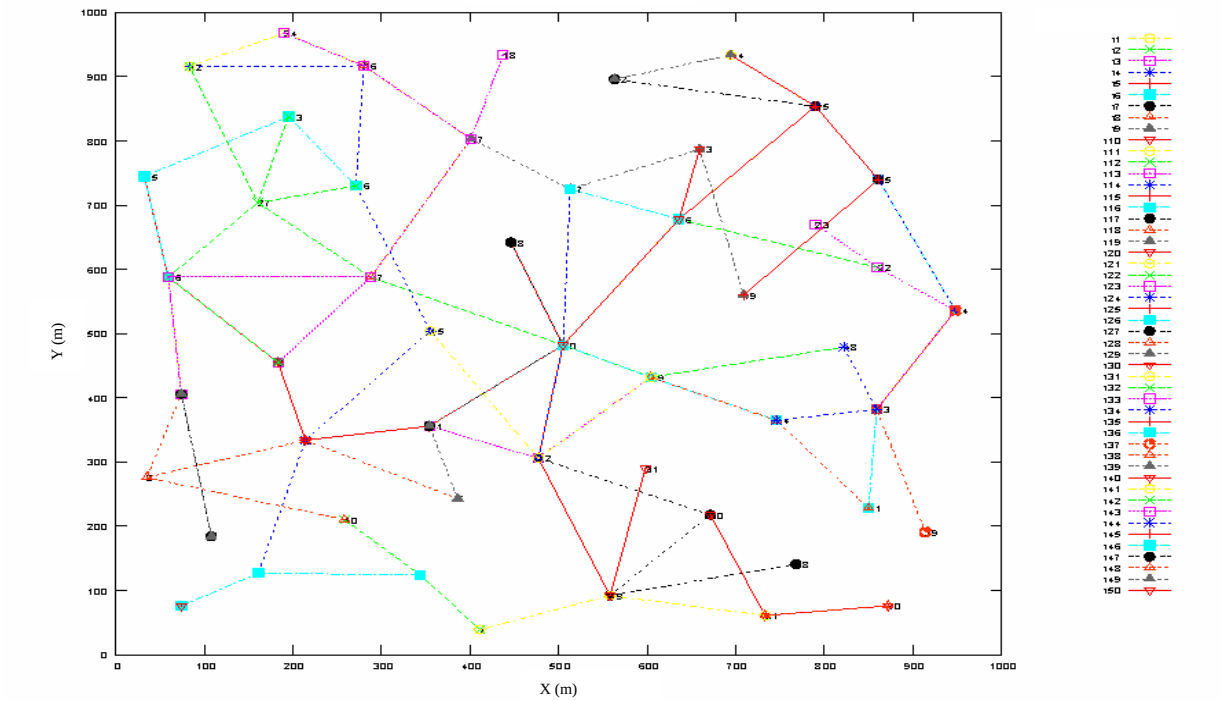


Figure 9. Routes representation of 50 flows

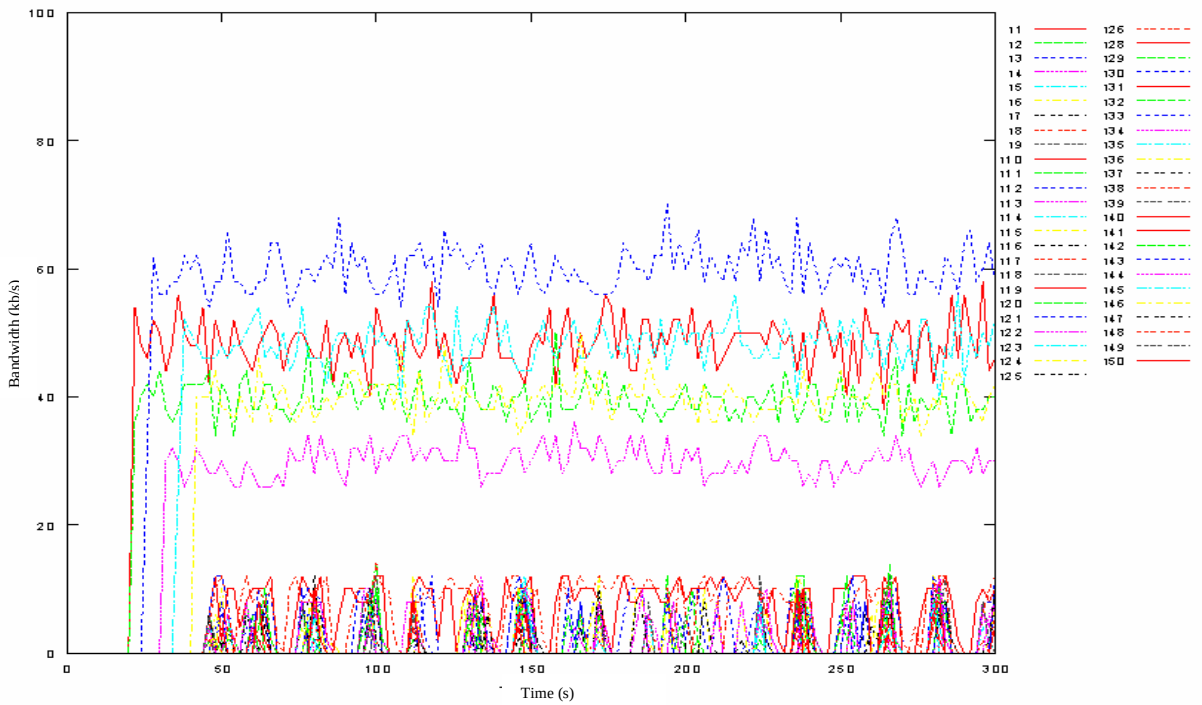


Figure 10. Measured instantaneous bandwidth
for 6 QoS and 44 BE flows

According to the simulation results, we can conclude that BE flows share the available bandwidth proportionally to their weight. In the considered scenarios, all BE flows request the same bandwidth and obtain the same weights for WCBQ. They also obtain the same measured bandwidth (Figure 8).

Concerning QoS flows, once admitted, they receive their requested bandwidth whatever the number of BE flows introduced in the network. We now consider another scenario with a higher number of flows where each node in the network generates at least one flow. We consider 50 flows: 6 QoS flows with the parameters given in the above Table 2 and 44 BE flows. Each BE flow requests a bandwidth of 10kb/s. In Figures 9 and 10, we present respectively routes taken by each flow, and the instantaneous bandwidth received by each flow in the network.

The results of the second example confirms that, in spite of the significant number of BE flows present in the network, QoS flows, once admitted, obtain their requested bandwidth. We also notice that the route of QoS flows is much more stable than the route of BE flows. Thus, QoS flows (f_1…,f_6) always use the same route whatever the number of BE flows present. The number of route changes of BE flows is very large, it can reach 12 during a simulation (300s).

3.3. Requisition of bandwidth by QoS flows

In our solution, the routing table of QoS flows is periodically computed in order to provide the shortest route satisfying the requested QoS. To study this characteristic, we consider the network of 50 nodes with at the beginning an overloaded zone with 44 BE flows. The bandwidth requested by each BE flow is equal to 10kb/s. We introduce QoS flows (f_1…, f_6) at times t_1 = 100s, t_2 = 106s, t_3 = 112s, t_4 = 118s, t_5= 124s and t_6 = 130s respectively. Flows f_1 and f_5 request a bandwidth of 60kb/s, flows f_2 and f_4 request a bandwidth of 30kb/s and flows f_3 and f_6 request a bandwidth of 40 kb/s. We stop the transmission of BE flows at time t = 200s, and then, we study the behavior of QoS flows in the absence of BE flows.

Figure 11 represents routes taken by each flow, and Figure 12 represents the instantaneous bandwidth received by each flow.

In conclusion, this configuration shows that the six QoS flows did not circumvent the overloaded zone by BE flows. Indeed, when QoS flows arrive, they requisition the bandwidth used by BE flows. Moreover, each QoS flow takes only one route. This route does not change even in the absence of BE flows, because it is the shortest route satisfying the requested bandwidth. Consequently each QoS flow receives its requested bandwidth.

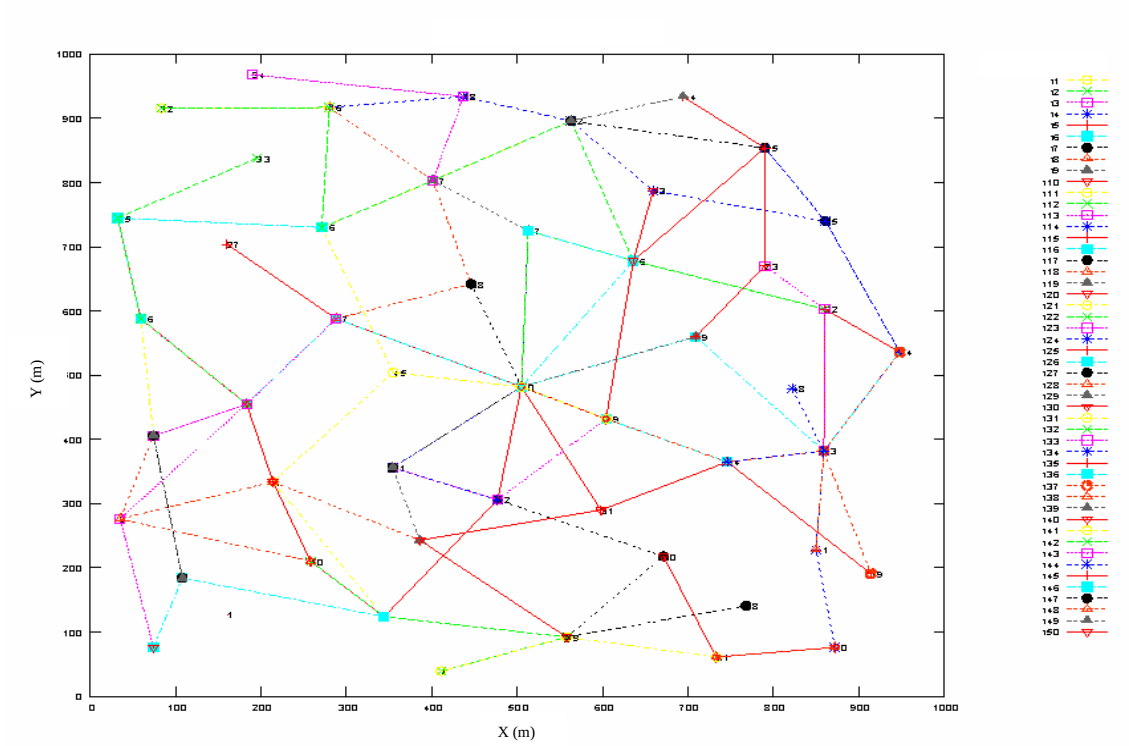


Figure 11. Routes representation

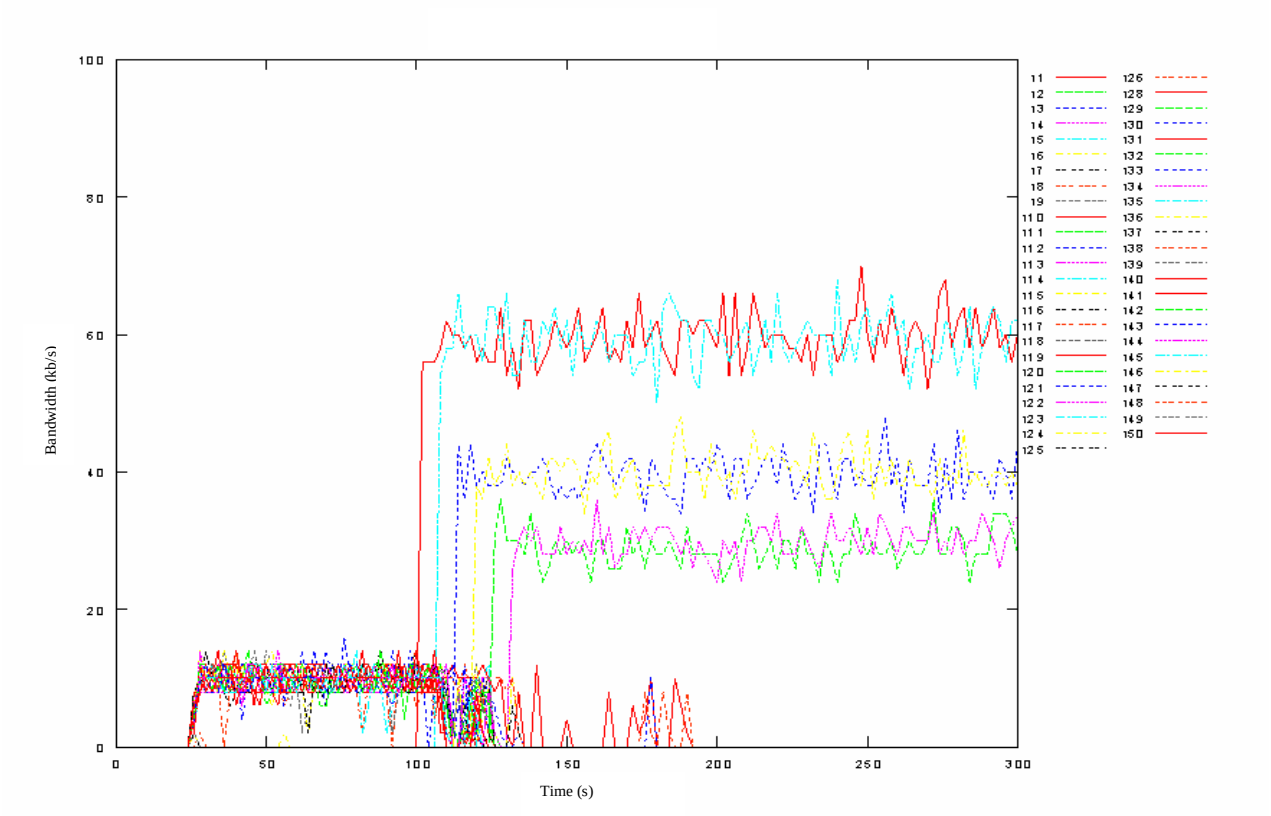


Figure 12. Measured instantaneous bandwidth

3.4. Benefits brought by QoS support

In this section, we evaluate the benefits brought by the QoS support in terms of bandwidth obtained by QoS flows. We consider again the scenario described in the section 3.2, including 6 QoS flows and 10 BE flows. We compare the performances obtained by our solution and those obtained by native OLSR. Figure 13 shows that with the QoS support, each QoS flow obtains the required band-width while with native OLSR, QoS flows f_1, f_2, f_3, f_4, f_5 and f_6 obtain only 28kb/s, 26kb/s, 40kb/s, 24kb/s, 33kb/s and 32kb/s respectively.

Figures 14 and 15 represent the instantaneous bandwidth obtained by each QoS flow with respectively QoS support and native OLSR. With native OLSR, the instantaneous bandwidth obtained by each QoS flow is very chaotic, while with the QoS support, it has only light oscillations around the requested bandwidth.

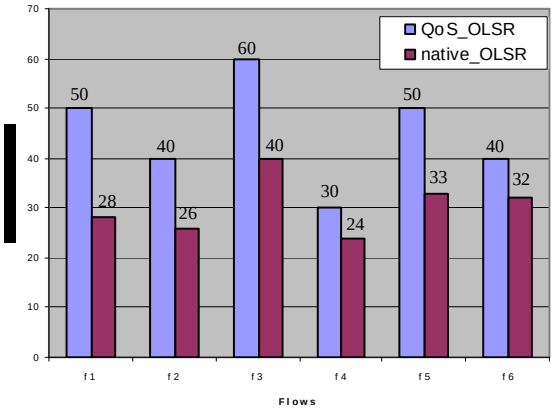


Figure 13. Average measured bandwidth of QoS flows with QoS OLSR and native OLSR

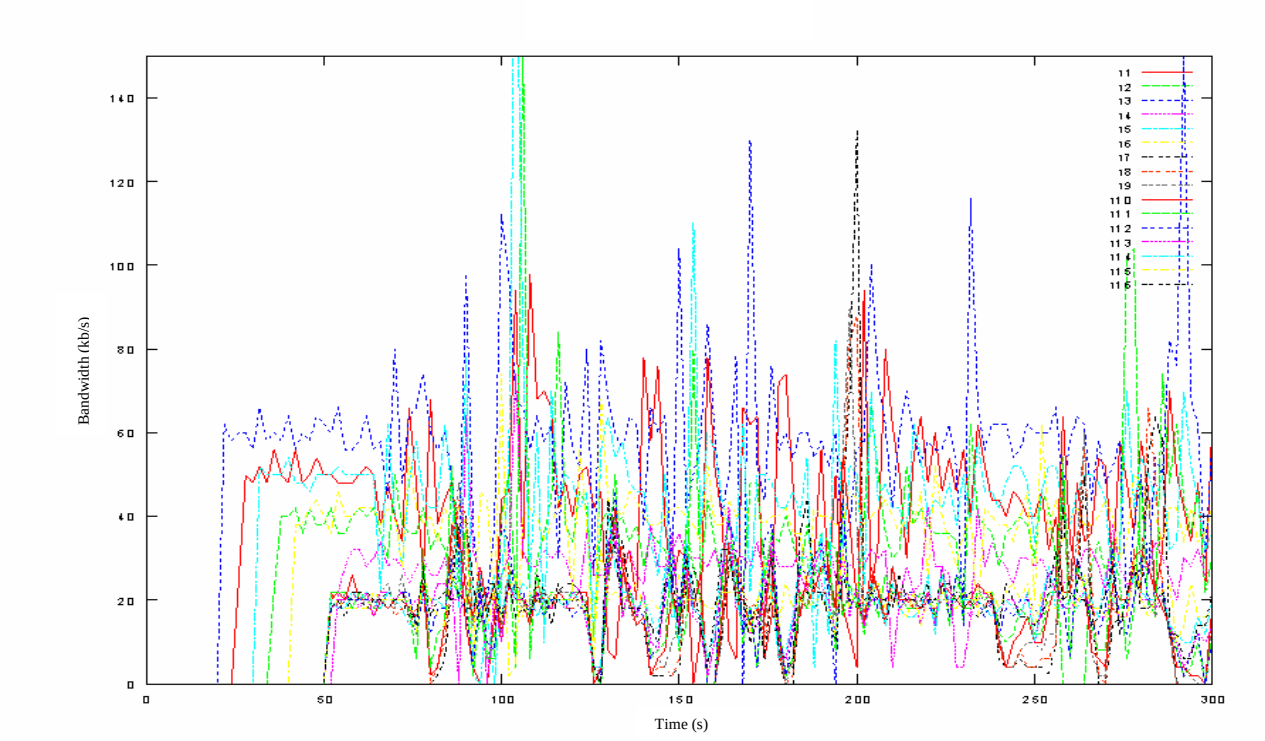


Figure 14. Measured instantaneous bandwidth with native OLSR

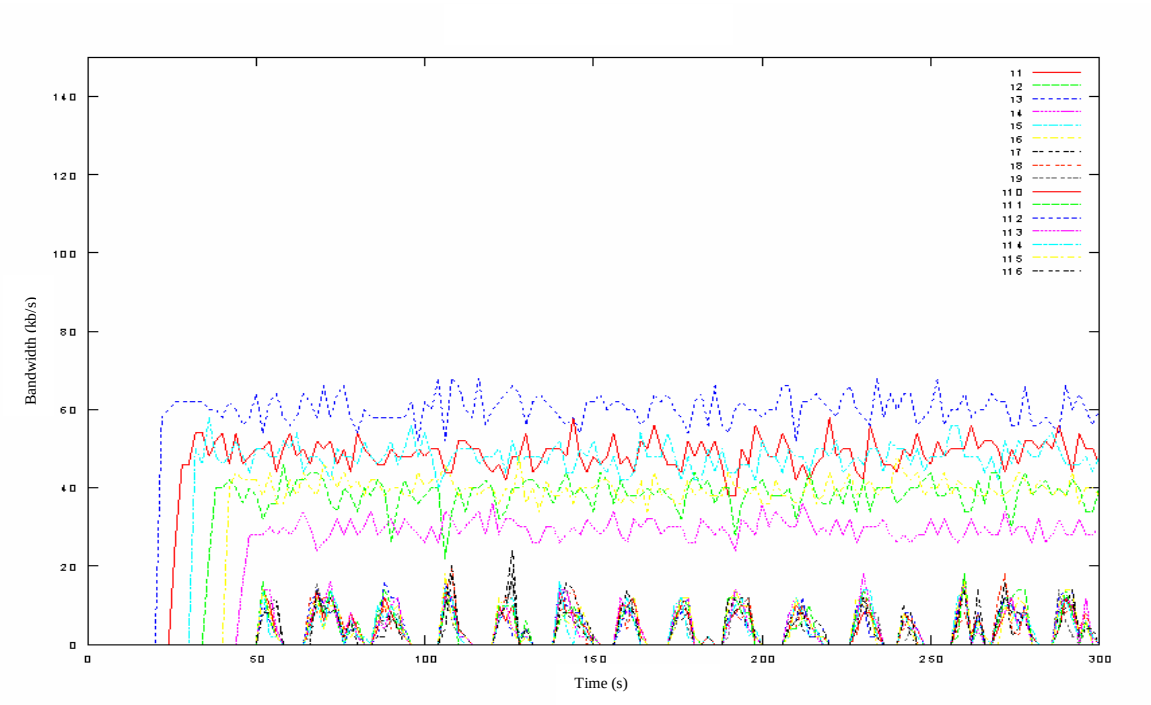


Figure 15. Measured instantaneous bandwidth with QoS support

3.5. QoS flows with delay constraints

In our solution, flows having delay constraints are prioritized compared to flows with bandwidth constraints and BE flows. In this section, we study the interest of the delay flows class and show the interest of having several priorities in this class.

In an ad hoc network of 50 nodes, we consider three flows: a first one with delay constraint (DL), a second one with bandwidth constraint (BW) and a third one with no constraint (BE). The three flows have the same rate (100kb/s), the same source (N_6) and the same destination. According to the number of hops and for each flow, we measure the received bandwidth and the end-to-end delay. These measurements are presented in figures 16 and 17:

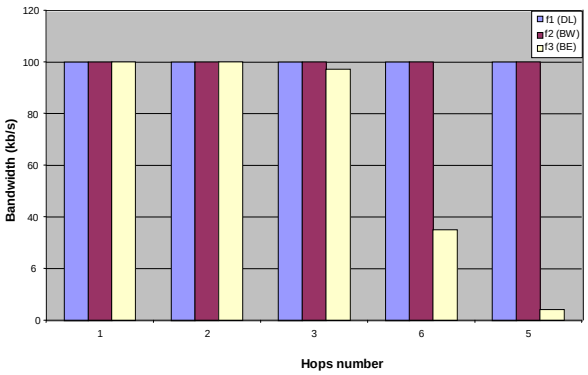


Figure 16. Measured average bandwidth according to hops number

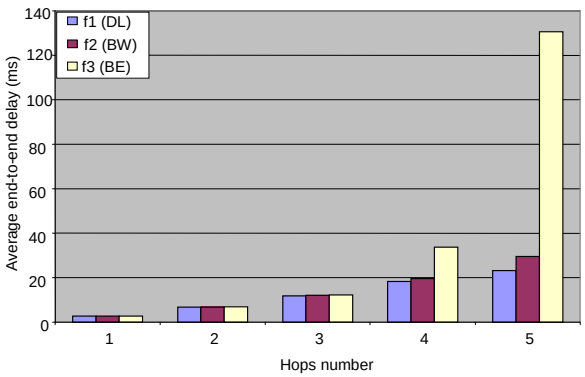


Figure 17. Measured average end-to-end delay according to hops number

We can conclude that once admitted, QoS flows with delay constraint and flows with bandwidth constraint obtain their requested bandwidth. For a number of hops higher than three, the end-to-end delay of flows having delay constraint is smaller than that obtained by flows having bandwidth constraint and that obtained by BE flows. This method thus makes it possible to privilege flows with delay constraint. Consequently, they obtain shorter delays.

Now, let us analyse how to manage Delay_DATA_Queue in the presence of several flows having different delay constraints. For that, we carry out a comparative study between the two schedulers: Priority Queueing (PQ) and Earliest Deadline First (EDF).

- With PQ, the flow having the smallest end-to-end delay receives the highest priority. Packets are inserted in the Delay_DATA_Queue according to their priority *i.e.*, the packet with the highest priority is inserted ahead of the Queue. If there at least one packet with the same priority, the new packet is inserted after the last one. Packets are then transmitted in a FIFO manner.
- With EDF, a local deadline is associated to each packet. This local deadline is calculated according to (*i*) the end-to-end deadline of the flow to which it belongs and (*ii*) the number of hops towards the destination. On a node, packets are inserted in the Delay_DATA_Queue according to their local deadline *i.e.*, the packet with the smallest local deadline is inserted ahead of the Queue. Packets are then transmitted in a FIFO manner. In the following we demonstrate how to calculate deadlines of packets. For that let us define the following notation:

ete_rel_dead: end-to-end relative deadline
ete_abs_dead: end-to-end absolute deadline
loc_abs_ded: local absolute deadline
t: packet generation time on the source node
t_a: packet arrival time on a route node
r: number of hops from the current node towards the destination.

- To each packet generated at time *t* on the source node are associated:

$$ete\_abs\_dead = t + ete\_rel\_dead$$
$$loc\_abs\_ded = t + ete\_rel\_dead\ /r.$$

- On each route node (except the destination), the local absolute deadline of each packet is recalculated:

$$loc\_abs\_ded = (ete\_abs\_dead - t_a)/r + t_a$$

To compare the two schedulers, we consider five QoS flows having delay constraints. They have also the same source, the same destination and follow the same route made up of four hops. The delay constraints of flows are expressed in term of end-to-end relative deadline. The bandwidth of the 5 flows f_1, f_2, f_3, f_4 and f_5 is 70kb/s, 60kb/s, 60kb/s, 50kb/s and 50kb/s respectively. And their end-to-end relative deadlines are 0.8s, 0.9s, 1s, 1.1s and 1.2s respectively.

In our simulations, we measure, with the two schedulers PQ and EDF and for each flow:(*i*) the average end-to-end deadline (see Figure 18), (*ii*) the maximum end-to-end deadline (see Figure 19), and (*iii*) the rate of packets respecting their deadline (see Figure 20).

According to the simulation results, we can notice that:

- PQ tends to transmit in priority flows with the highest priority. That is why the measured average end-to-end delay and measured maximum end-to-end delay (see Figures 18 and 19) increase when the priority associated to the flow decreases. This strongly impacts the rate of packets respecting their deadline for the flows having the weakest priority (flow f5 in our case).
- EDF, on the other hand, tends to transmit all flows. Indeed, EDF is a dynamic scheduler based on the absolute deadlines. Priority of flows changes according to these absolute deadlines *i.e.*, at a given time, the flow having the smallest absolute deadline is scheduled for transmission. Consequently, EDF, which is known for its scheduling optimality [12] in the single processor context, provides a better rate of packets respecting their deadlines.

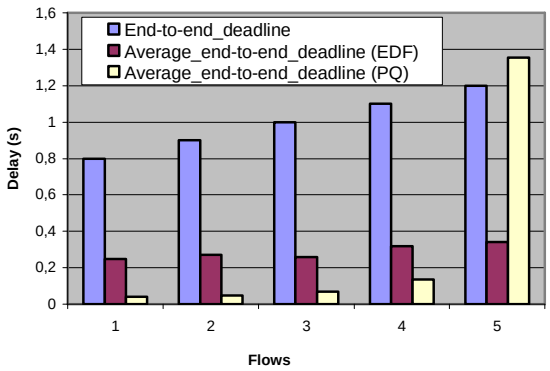


Figure 18. Measured average end-to-end deadline with EDF and PQ

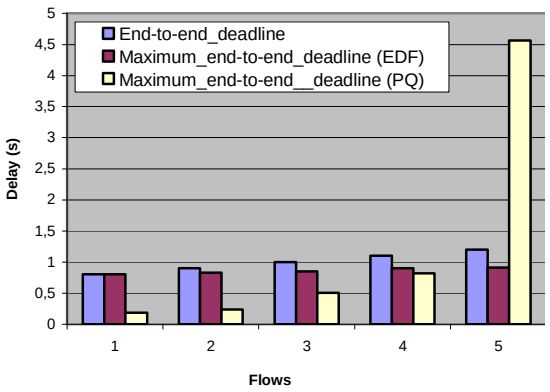


Figure 19. Measured maximum end-to-end deadline with EDF and PQ

According to this simulation results, we recommend to use a dedicated queue for flows having end-to-end delay constraints. We have evaluated the performance of EDF scheduling when the local deadline is computed has the difference between the end-to-end deadline and the time already spent in the network divided by the remaining number of hops towards the destination. Simulation results show that this EDF allows a higher rate of packets meeting their deadline. Therefore, we recommend its use for the Delay_Data_Queue.

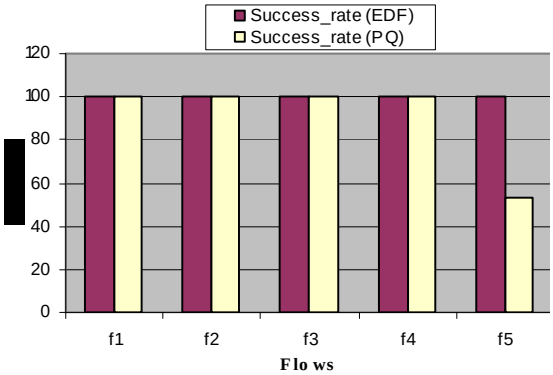


Figure 20. Rate of packets respecting their deadline

3.6. Overhead

Let us evaluate the overhead induced by the QoS support on each node. Thus, for each node, we calculate the number of OLSR messages sent per second. This number takes into account the OLSR messages generated by a node as well as the OLSR messages forwarded by this node. Figure 21 illustrates the overhead calculated for the scenario described in section 3.2 including 10 BE flows and 6 QoS flows.

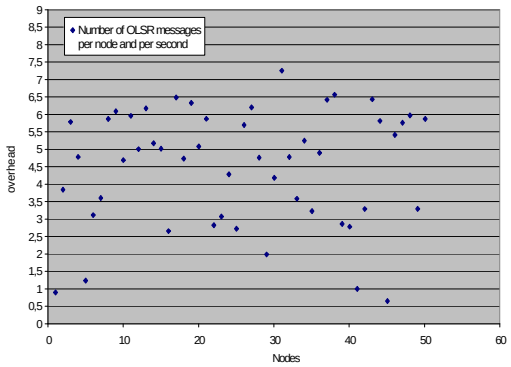


Figure 21. Overhead of QoS support

The average value of the overhead is equal to 4.504. It is larger than the average of the overhead obtained with native OLSR which is equal to 3.445 (see Figure 21). That is due to the fact that with the support of QoS, more nodes are selected as MPRBs and consequently generate TC messages in addition to Hello messages.

3.7. Mobility support

Now, we study the impact of the mobility on the QoS support performances. For the same network, we compare the performances obtained in the presence and the absence of QoS support. This evaluation is carried out on a network of 100 nodes (see Figure 22). The mobility model considered in the simulations is Random Waypoint Model (RWM) where the maximum speed for each node is limited to 5m/s. We define in Table4 the remaining simulation parameters.

In the considered network, 10 flows are present including 2 QoS flows with bandwidth constraint, and 8 BE flows. In table 5, we indicate the requested bandwidth, the source and the destination of each flow.

Table 4. Simulation parameters for an ad hoc network of 100 mobile

Simulation	- simulation duration: 300s - Number of nodes: 100 - Flat area: 1000mx1000m - Traffic type: CBR - Packet size: 500kb
Routing protocol (OLSR)	- Source routing for QoS flows - Hop by hop routing for BE flows - Periodic routing table calculation for QoS flows (period 1s) - Hello period: 1s - TC period: 5s - Use of MPRB
MAC	- MAC protocol: IEEE802.11b - Throughput: 2Mb/s - No RTS/CTS messages
Radio	- Radio propagation model : TwoRayGround - Transmission range: 250m - Interference range: 500m

For each flow present in the network, we measure the received bandwidth, we also provide the number of routes taken as well as the number of route changes during the simulation (see Table 6).

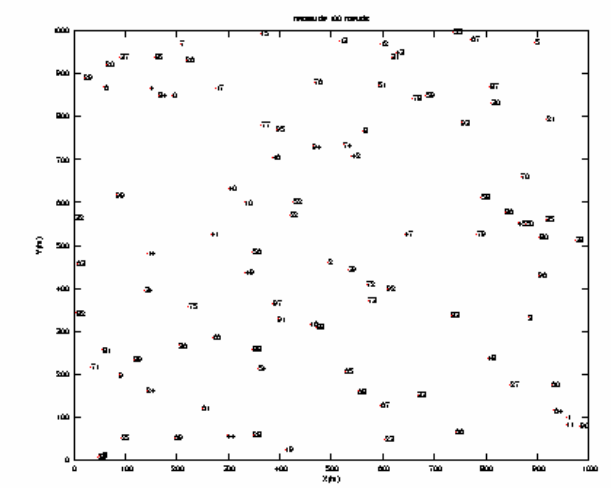


Figure 22. Ad hoc network of 100 mobile nodes

Table 5. Flows parameters

Flow	Type	Requested bandwidth (kb/s)	Source	Destination
f1	QoS	120	67	38
f2	QoS	140	71	8
f3	BE	50	49	58
f4	BE	100	67	90
f5	BE	80	22	0
f6	BE	80	71	19
f7	BE	80	94	61
f8	BE	50	5	66
f9	BE	80	60	72
f10	BE	50	76	30

Table 6. Simulation results

Flows	Type	Requested bandwidth (kb/s)	Measured bandwidth (kb/s)	Routes number	Number of route changes
f1	QoS	120	118	9	11
f2	QoS	140	122	47	51
f3	BE	50	16	7	8
f4	BE	100	10	12	17
f5	BE	80	13	13	15
f6	BE	80	24	3	3
f7	BE	80	18	7	10
f8	BE	50	4	14	13
f9	BE	80	30	2	1
f10	BE	50	9	12	14

Now, we consider the same previous scenario without QoS support (native OLSR). In this case, the two QoS flows respectively obtain 94 kb/s and 70 kb/s (Figure 23).

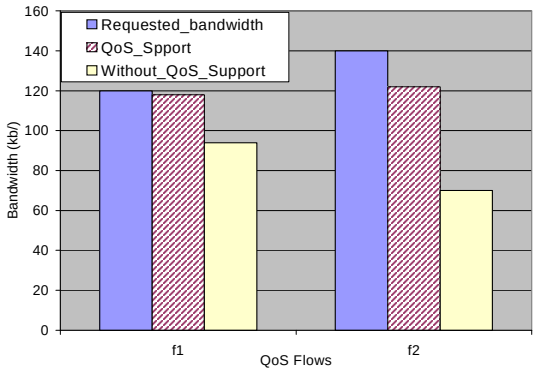


Figure 23. Measured average bandwidth for QoS flows in the two cases: in presence and in absence of QoS support

According to the simulation results, we notice that, with the QoS support, QoS flows obtain more bandwidth. For example, in the scenario above, the first QoS flow *f*1 received a bandwidth (118/ kb/s) close to its required bandwidth (120 kb/s). The second QoS flow *f*2, has received 122 kb/s whereas it requested 140 kb/s. If QoS flows did not obtain the exact requested bandwidth, it is because of the random mobility of the nodes *i.e.*, the random mobility of nodes induces a link failure and consequently a loss of packets. Let us note for example that flow *f*2 changed its route 51 times. A change of route can be due to the one of the three reasons below:

- a link of the current route becomes invalid,
- the current route does not satisfy the required QoS,
- a shorter route satisfying the required bandwidth is found.

Without QoS support, QoS flows see their QoS being degraded. In our example, QoS flows *f*1 and *f*2 obtained only 94 kb/s and 70 kb/s respectively. That is due, on one hand, to the mobility of nodes, and on the other hand to the interferences induced by BE flows which are not taken into account.

4. Conclusion

In this paper, we have proposed a QoS support for mobile ad hoc networks, based on the OLSR routing protocol and the CBQ scheduling. This QoS support takes into account radio interferences and is based on six QoS components: QoS MAC, QoS model, admission control, QoS routing, QoS signaling and scheduling.

User flows are classified according to three types:

- QoS flows with delay constraints
- QoS flows with bandwidth constraints
- BE flows with no specific QoS requirements.

To schedule these three user flow types, we have combined the use of two schedulers which are WCBQ and Priority Queueing (PQ). The CTRL_Queue, dedicated to control traffic, has the highest priority. The Delay_DATA_Queue, dedicated to QoS flows with delay requirements, has lower priority. The Bandwidth_DATA_Queues are dedicated to QoS flows with bandwidth requirement. Best-Effort_DATA_Queues are reserved to BE flows. Bandwidth_DATA_Queues and Best-Effort_DATA_Queues are managed according to CBQ where the weight associated with each flow depends on its bandwidth request. The Delay_Data_Queue is scheduled according to EDF where the local deadline of a packet on a visited node is computed from (*i*) the end-to-end deadline of the flow this packet belongs to, (*ii*) the time already spent by this packet in the network, (*iii*) the number of hops remaining to the destination. This EDF scheduling increases the rate of packets meeting their end-to-end deadline.

We have shown by means of NS2 simulations that this solution provides a fair sharing of bandwidth for best effort flows and ensures route stability for QoS flows. As a consequence, these flows have shorter delays and jitters. As QoS flows are allowed to requisition the bandwidth used by BE flows, they use the shortest route providing the requested QoS. We have also, pointed out the benefits brought by this solution with regard to native OLSR. The overhead of this solution is kept reasonable. Finally, we have shown that our solution supports node mobility.

5. References

[1] L. Boukhalfa, P. Minet, L. George, S. Midonnet, “Mobile ad hoc networks and QoS demanding applications”, *in: 5th IEEE int. conf. on Mobile and Wireless Communications Networks, MWCN'03*, Singapore, October 2003.

[2] L. Boukhalfa, P. Minet, S. Midonnet, L. George, “Comparative evaluation of CBQ and PriQueue in a MANET”, *in IEEE int. Workshop on Heterogeneous Multi-hop Wireless and Mobile Networks, IEEE MHWMN’05*, Washington, November 2005.

[3] L. Boukhalfa, P. Minet, S. Midonnet, “QoS support in a MANET based on OLSR and CBQ”, *IEEE International Conference on Networking, ICN'07*, Sainte-Luce, Martinique, April 2007.

[4] Z.Y. Demetrios, “A glance at Quality of Service in Mobile Ad-Hoc Networks”, Research report for CS260-seminar in Mobile Ad-Hoc Networks, 2001.

[5] S. Floyd and V. Jacobson. “Link-sharing and Resource Management Models for Packet Networks”, *IEE/ACM Transactions on Networking*, August 1995, pp. 365-386.

[6] T. Clausen, P. Jacquet , “Optimized Link State Routing Protocol (OLSR) ”, IETF RFC 3626, Project Hipercom, INRIA, October 2003.

[7] J. Moy , “OSPF Version 2”, IETF RFC 2328, April 1998.

[8] The VINT Project. The network simulator - ns-2 v.2.1b7a. http://www.isi.edu/nsnam/ns/, November 2000.

[9] L. Georgiadis, P. Jacquet, B. Mans, “Bandwidth Reservation in Multihop Wireless Networks: Complexity and Mechanisms”, *24th International Conference on Distributed Computing Systems Workshops W6: WWAN (ICDCSW’04)*, 2004, pp. 762-767.

[10] R. Gupta, Z. Jia, T. Tung, J. Walrand, “Interference-aware QoS Routing (IQRouting) for Ad-Hoc Networks”, *Globecom*, St. Louis, MO, November 2006.

[11] A. Vers, A. Campbell, M. Barry, L. Sun, “Supporting Service Differentiation in Wireless Packet Networks using Distributed Control”, *IEEE Journal on Selected Areas in Communications*, vol.19, N°10, October 2001.

[12] K. Jeffay, D. F. Stanat, C.U. Martel, “On Non-Preemptive Scheduling of Periodic and Sporadic Tasks”, *IEEE Real Time Systems Symposium*, December 1991, pp. 129-139.

End-to-End Prediction Model of Video Quality and Decodable Frame Rate for MPEG Broadcasting Services

Harilaos Koumaras, Anastasios Kourtis
Institute of Informatics and Telecommunications
NCSR Demokritos
Athens, Greece
{koumaras, kourtis}@iit.demokritos.gr

Cheng-Han Lin, Ce-Kuen Shieh
High Performance Parallel & Distributed Syst. Laboratory
National Cheng Kung University
Tainan, Taiwan
{jhlin5,shieh}@hpds.ee.ncku.edu.tw

***Abstract*—This paper proposes, describes and evaluates a novel theoretical framework for end-to-end video quality assessment of MPEG-based video services in hand-held and mobile wireless broadcast systems. The proposed framework consists two discrete models: A model for predicting the video quality of an encoded signal at a pre-encoding state by specifying the bit rate that satisfies a specific level of user satisfaction and a model that maps the packet loss ratio of the transmission channel to the quality degradation percentage of the broadcasting service. The accuracy of the proposed framework is experimentally validated, demonstrating its efficiency.**

***Keywords*— Video Quality, MPEG, Packet Loss, DVB.**

I. INTRODUCTION

RECENTLY, MPEG-based applications that are specialized and adapted in broadcasting digitally encoded audiovisual content have known an explosive growth in terms of development, deployment and provision. The new era of digital video broadcasting for hand-held terminals has arrived and the beyond MPEG-2 based transmission for terrestrial or satellite receivers is a fact, setting new research challenges for the assessment of Perceived Quality of Service (PQoS) under the latest MPEG-4/H.264 and the DVB-H standard.

MPEG standards exploit in their compression algorithms the high similarity of the depicted data in the spatial, temporal and frequency domain among subsequent frames of a video sequence. Removing the redundancy in these three domains, it is achieved data compression against a certain amount of visual data loss, which from the one hand it cannot be retrieved but on the other hand it is not perceived and conceived by the mechanisms of the Human Visual System.

Therefore, MPEG-based coding standards are characterized as lossy techniques, since they provide efficient video compression with cost a partial loss of the data and subsequently the video quality degradation of the initial signal. Due to the fact that the parameters with strong influence on the video quality are normally those, set at the encoder (with most important the bit rate), the issue of the user satisfaction in correlation with the encoding parameters has been raised.

A content/service provider, depending on the content dynamics, must decide for the configuration of the appropriate encoding parameters that satisfy a specific level of user satisfaction.

Currently, the determination of the encoding parameters that satisfy a specific level of video quality is a matter of recurring subjective or objective video quality assessments, each time taking place after the encoding process (repetitive post-encoding evaluations). Subjective quality evaluation processes of video streams require large amount of human resources, establishing it as an impractical procedure for a service provider. Similarly, the repetitive use of objective metrics on already encoded sequences may require numerous test encodings for identifying the specified encoding parameters, which is also time consuming and financially

unaffordable from a business perspective.

Once the broadcaster has encoded appropriately the offered content at the preferred quality level, then the provision of the service follows. Digitally video encoded services, due to their interdependent nature, are highly sensitive to transmission errors (e.g. packet loss, network delay) and require high transmission reliability in order to maintain between sender and receiver devices their stream synchronization and initial quality level. Especially, in video broadcasting, which is performed over wireless environments, each transmitted from one end video packet can be received at the other end, either correctly or with errors or get totally lost. In the last two cases, the perceptual outcome is similar, since the decoder at the end-user usually discards the packet with errors, causing visual artifact on the decoded frame and therefore quality degradation.

In this context, the paper aims at proposing, describing and evaluating a theoretical framework for end-to-end video quality assessment of MPEG-based broadcasting services, focusing on:

i. The prediction of the encoding parameters that satisfy a specific video quality level in terms of encoding bit rate and content dynamics.
ii. The mapping of the packet loss ratio during the transmission to the respective quality degradation percentage.

Through the proposed end-to-end video quality assessment framework, the content provider (i.e. the broadcaster) will be able to estimate the finally delivered video quality level, considering specific encoding parameters and transmission conditions. Such an end-to-end perceived QoS framework will not only play an essential role in performance analysis, control and optimization of broadcasting systems, but it will also contribute towards a more efficient resource allocation, utilization and management.

The rest of the paper is organized as follows: Section II performs a literature review on the relative research works, focusing both on the video quality assessment and the estimation of the degradation due to the conditions of the transmission channel. Also, in this section are described the fundamental concepts of a MPEG encoded signal, which will be later used for the description of the proposed framework. Section III describes and evaluates the proposed model for the prediction and determination of the encoding bit rate value that satisfies a specific level of user satisfaction. Similarly, Section IV discusses the consequence of a packet loss on the transmitted broadcasting signal, focusing on the decoding performance of the service. In this context, it is described the proposed model for the video quality degradation over error-prone transmission channel. In section V, the concept of the end-to-end video quality assessment framework is introduced, described and explained. Finally, Section VI concludes the paper.

II. BACKGROUND

A. Video Quality Assessment Methods

The advent in the field of video quality assessment is the application of pure error-sensitive functions between the encoded and the original/uncompressed video sequence. These primitive methods, although they initially provided a quantitative approach of the degradation caused by the encoding procedure, practically they do not reflect reliably the video quality as it is observed and perceived by human viewers.

Beyond these primitive models, currently the evaluation of the video quality is a matter of objective and subjective procedures, which are applied after the encoding process (post-encoding evaluation).

The subjective test methods, which have mainly been proposed by International Telecommunications Union (ITU) and Video Quality Experts Group (VQEG), involve an audience, who watch a video sequence and score its quality as perceived by the participants, under specific and controlled watching conditions. Afterwards, usually the Mean Opinion Score (MOS) is exploited for the statistical analysis and processing of the collected data.

Subjective video quality evaluation processes require large amount of human resources, making it a time-consuming process (e.g. large audiences evaluating test sequences). On the other hand, objective evaluation methods provide faster quality assessment, exploiting multiple metrics related to the encoding artifacts (e.g. tilling, blurriness, error blocks, etc).

The majority of the objective methods require the undistorted video source as a reference entity in the quality evaluation process. Due to this requirement, they are characterized as Full Reference (FR) Methods [1-3]. However, it has been reported that these complicated FR methods do not provide more accurate results than the simple mathematical measures (such as PSNR). Towards this, lately some novel full reference metrics have been proposed based on the video structural distortion and content entropy [5-8].

On the other hand, the fact that these methods require the original video signal as reference deprives their use in broadcasting services, where the initial undistorted clips are not accessible at user side. Moreover, even if the reference clip becomes somehow available, then synchronization predicaments between the undistorted and the distorted signal (which may have experienced frame losses) make the FR methods practically inapplicable.

Due to these reasons, the recent research has been focused on developing methods that can evaluate the PQoS level based on metrics, which use only some extracted structural features from the original signal (Reduced Reference Methods) [9-13] or do not require any reference video signal (No Reference Methods). The NR methods can be classified into two classes: The NR-visual based and the NR-coded based. The first methods must initially decode the bit stream

and estimate the video quality at the visual domain [14-19], while the second ones assess the perceived quality directly through the compressed bit stream, without requiring any decoding [20-25].

Finally, some alternative objective methods have been proposed, which move beyond the simple post-encoding quality assessment and introduce the concept of video quality prediction for given encoding parameters and content dynamics at a pre-encoding state [26-28, 40]. In this direction will focus the content of this paper and more specifically the proposed model for the determination of the bit rate values that satisfy specific perceptual levels.

B. Quality Degradation due to Transmission Errors

The issue of mapping the perceptual impact of transmission errors (like packet loss) during the broadcasting on the delivered perceptual video quality at the end-user is a fresh topic in the field of video quality assessment since the relative literature appears to be limited with a small number of relative published works.

In this framework, S. Kanumuri *et al* [29] proposed a very analytical statistical model of the packet-loss visual impact on the decoding video quality of MPEG-2 video sequences, specifying the various factors that affect the perceived video quality and visibility (e.g. Maximum number of frames affected by the packet loss, on what frame type the packet loss occurs etc). However, this study focuses mainly on the pure study of the MPEG-2 decoding capabilities, without considering the parameters of the digital broadcasting or the latest encoding standards.

Similarly, in [30] is presented a transmission distortion model for real-time video streaming over error-prone wireless networks. In this work, an end-to-end video distortion study is performed, based on the modeling of the impulse propagation error (i.e. the visual fading behavior of the decoding artifact). The deduced model, although it is very accurate and robust, enabling the media service provider to predict the transmission distortion at the receiver side, is not a generic one. On the contrary, it is highly dependent on the video content dynamics and the selected encoder settings. More specifically, it is required an initial quantification of the spatial and temporal dynamics of the content, which will allow the appropriate calibration of the model. This prerequisite procedure (i.e. adapting the impulse transmission distortion curve based on the least mean square error criteria) is practically inapplicable by an actual content creator/provider. Moreover, the strong dependence of the proposed model on the spatiotemporal dynamics of the content deprives its implementation on sequences with long duration and mixed video dynamics, since not a unique impulse transmission distortion will be accurate for the whole video duration.

In this context, our paper describes, proposes and testes a generic model for end-to-end video quality prediction for MPEG-based broadcasting services. Our framework consists two discrete parts:

- A method for predicting and specifying for a given content the encoding parameters that satisfy a specific perceptual level at a pre-encoding state
- A model of the perceptual impact of the broadcasting packet loss ratio on the delivered perceived quality of the transmitted service.

Thus, to the best of our knowledge, this work is one of the first models providing end-to-end video quality prediction across all the lifecycle of the media content: From the service generation down to the content consumption at the viewer side.

C. MPEG Video Structure

The MPEG standard [31] defines three frame types for the compressed video streams, namely I, P and B frames. The I frames are also called Intra frames, while B and P are known as Inter frames. The successive frames between two succeeding I frames is defined as Group Of Pictures (GOP). The frame classification is mainly based on the procedure, according to which each frame type has been generated and encoded. This differentiation sets also some special requirements for the successful decoding of each frame type.

More specifically, MPEG I frames (Intra-coded frames) are encoded independently and there is no special requirement in their decoding process, given that all the respective data packets have been successfully received. The encoding of the MPEG P frames (Predictive-coded frames) is based on reference spatial areas from the preceding I or P frames within the specific GOP. Therefore, for their successful decoding -except for the successful reception of their respective data- it is required successful decoding of the reference I or P frames. Finally, MPEG B frames (Bi-directionally predictive-coded frames) are encoded using references from the preceding and succeeding I or P frames. Consequently, for their successful decoding apart from the successful reception of the data packets that carry the B frame, also the respective reference frames must be successfully received and decoded.

The structure of the GOP is generally specified by the selected encoding settings. In the MPEG literature the GOP pattern is described by two parameters GOP(N,M), where N defines the GOP length (i.e. the total number of frames within each GOP) and the M-1 is the number of B frames between I-P or P-P frames. For example, as shown in figure 1, GOP(12, 3) means that the GOP consists one I frame, three P frames, and eight B frames. Also seen in figure 1, the second I frame marks the beginning of the next GOP. The arrows indicate that the B and P frames successful decoding depends on the respective preceding and succeeding I or P frames.

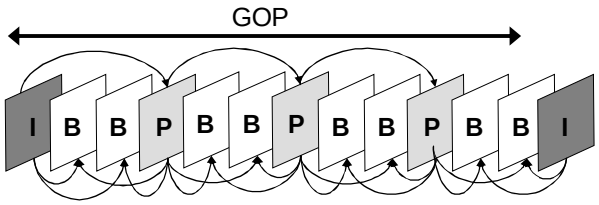


Fig. 1. A sample of MPEG GOP (N=12, M=3)

Therefore, from the hierarchical structure of MPEG encoding as it is depicted on figure 1, a video frame may be considered as directly or indirectly undecodable. Direct undecodable is considered a video frame when there are not enough received packets of the frame in order to achieve a successful decoding. On the other hand, indirect undecodable is considered a video frame when a reference frame is directly or indirectly undecodable. For simplicity, we do not consider video concealment issues in this study and we set the Decodable Threshold (DT) [13] equal to 1.0. Therefore, our analysis provides the worst-case scenario in terms of video quality degradation and decoding robustness.

III. Modeling and Predicting Video Quality

In digital video encoding the Block Discrete Cosine Transformation (BDCT) is exploited, since it exhibits very good energy compaction and de-correlation properties. In this paper, we use the following conventions for video sequences: Every real NxN frame f is treated as a N^2x1 vector in the space R^{N^2} by lexicographic ordering either in rows or columns.

The DCT is considered as a linear transform from $R^{N^2} \rightarrow R^{N^2}$. Thus, for a typical frame f, we can write:

$$F = Bf$$

Since B matrix is unitary, the inverse DCT can be expressed by B^t, where t denotes the transpose of a vector or matrix. Thus, the inverse transform can be described as:

$$f = B^t F$$

The elements of frame $F = Bf$ in the frequency domain can be expressed as the coefficients of the vector f, using the DCT basis in R^{N^2}. Thus

$$f = \sum_{n=1}^{N^2} F_n e_n$$

where e_n is the normalized DCT basis vector and F_n the DCT coefficients of f.

The high compression during the MPEG-related encoding process is (among other procedures) based on the quantization of the DCT coefficients, which in turn results in loss of high frequency coefficients. Within a MPEG block/macroblock, the luminance differences and discontinuities between any pair of adjacent pixels are reduced, by the encoding and compression process. On the contrary, for all the pairs of adjacent pixels, which are located across and on both edge sides of the border of adjacent DCT blocks, the luminance discontinuities are increased by the encoding process. Thus, after the quantization:

$$F_n^{'} = Q[F_n]$$

where $Q[\,]$ denotes the quantization process.

So, at the decoder side, the final reconstructed frame (after motion estimation and compensation modules) will be given by:

$$f^{'} = \sum_{n=1}^{N^2} F_n^{'} e_n$$

Thus, the perceived quality degradation per frame due to the encoding and quantization process can be expressed by an error based framework in the luminance domain Δf_Y between the original and the decoded frames.

$$\Delta f_Y \propto f_Y - f_Y^{'}$$

In this context, an average of the PQoS level for the whole encoding signal, consisting of *N* frames, can be derived by the following error-based equation:

$$< PQoS >_{video} \propto \sum_{i=1}^{N} \Delta f_{Y_i}$$

An objective perceived quality metric, which provides very reliable assessment of the video quality, based on this error-based framework, is the *SSIM* metric. The *SSIM* is a FR objective metric, which measures the structural similarity between two image/video sequences, exploiting the general principle that the main function of the human visual system is the extraction of structural information from the viewing field. Thus, considering that *f* and *f'* depicts the frames of the uncompressed and compressed signal respectively, then the *SSIM* is defined as [3, 6]:

$$SSIM(f, f') = \frac{(2m_f m_{f'} + D_1)(2s_{ff'} + D_2)}{(m_f^2 + m_{f'}^2 + D_1)(s_f^2 + s_{f'}^2 + D_2)}$$

where μ_f, $\mu_{f'}$ are the mean of *f* and *f'*, σ_f, $\sigma_{f'}$, $\sigma_{ff'}$ are the variances of *f*, *f'* and the covariance of *f* and *f'*, respectively. The constants D_1 and D_2 are defined as:

$$D_1 = (K_1 L)^2 \quad D_2 = (K_2 L)^2$$

where *L* is the dynamic pixel range and K_1 = 0.01 and K_2 = 0.03, respectively.

Thus, *SSIM* metric can be considered as a reliable metric for quantifying PQoS for video services. Figure 2 depicts a typical example of the *SSIM* measurement per frame for the video trailer "16 Blocks", which was encoded using the MPEG-4/H.264 standard VBR at 200 Kbps with Common Intermediate Format (CIF) resolution and 25 frames per second (fps). The instant *SSIM* vs. time curve (where time is represented by the frame sequence) varies according to the

spatiotemporal activity of each frame, which causes different quality degradation for the same quantization parameters. For frames with high complexity the instant *SSIM* level drops (i.e. <0.9), while for static frames the instant PQoS is higher (i.e. >0.9 or equal to 1, which denotes no degradation at all).

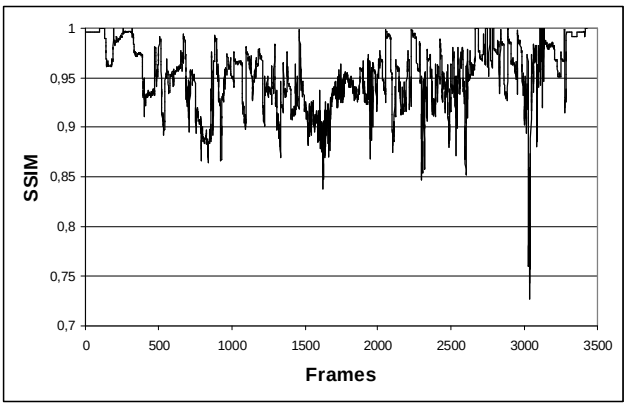


Fig. 2. The instant *SSIM* per frame of "16 Blocks" CIF 200kbps VBR

The concept of averaging the *SSIM* for the whole video duration can be exploited for deriving the mean PQoS as it was earlier defined. However, although the mean PQoS provides a single perceived quality measurement, which is more practical especially for the service providers, for long duration videos, where the spatial and temporal activity level of the content may differ significantly, the deduction of just one measurement of the perceived quality may not be accurate. In such long sequences, the proposed average metric can be combined along with a scene change detector algorithm, which will lead to calculating partial average PQoS for the various scenes. However, this case is not within the purposes of the current paper and it is not examined. The paper aims at quality issues in hand-held and small screen mobile devices, where short in duration signals are broadcasted, such as movie trailers, news or music clips with practically constant and homogeneous level of spatial and temporal activity.

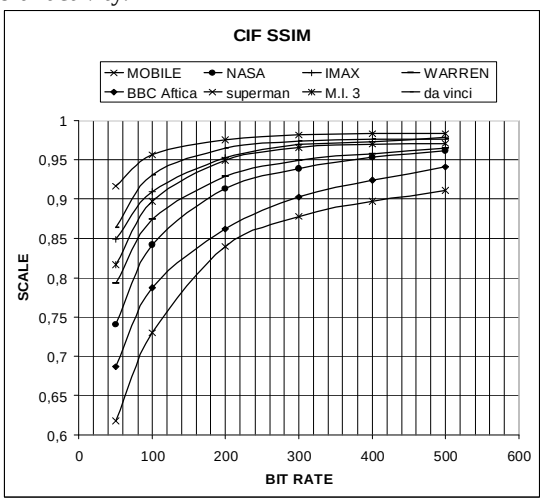


Fig. 3. The <PQoS>$_{SSIM}$ vs. bit rate curves for various test signals

In this context, eight short in duration video clips were selected and used for the needs of this paper. The experimental set consisted trailer video clips with duration up to three minutes. Each trailer clip was transcoded from its original H.264 format with Hi-Def resolution (i.e. 720p) to MPEG-4/H.264 Baseline Profile at diverse VBR bit rates. For each corresponding bit rate, a different MPEG-4/H.264 compliant file with CIF (Common Interface Format) resolution (352x288) was created. The frame rate was set at 25 frames per second (fps) for the transcoding process in all test videos.

Each encoded video clip was then used as input in the SSIM estimation algorithm. From the resulting SSIM vs. time graph (like the one in Figure 2), the <PQoS> value of each clip was calculated. This experimental procedure was repeated for each video clip in CIF resolution. The results of these experiments are depicted in Figure 3.

Referring to the curves of Figure 3, the following remarks can be made:

1. The minimum bit rate of the lowest <PQoS>$_{SSIM}$ value depends on the S-T activity level of the video clip.
2. The variation of the <PQoS>$_{SSIM}$ vs. bit rate is an increasing function, but non linear.
3. The quality improvement of an encoded video clip is not significant for bit rates higher than a specific threshold. This threshold depends on the S-T activity of the video content.

Moreover, each <PQoS>$_{SSIM}$ vs. bit rate curve can be successfully described by a logarithmic function of the general form

$$< PQoS >_{SSIM} = C_1 \ln(BitRate) + C_2$$

where C_1 and C_2 are constants strongly related to the spatial and temporal activity level of the content. Table 1 depicts the corresponding logarithmic functions for the test signals of Figure 3 along with their R^2 factor, which denotes the fitting efficiency of the theoretical curve to the experimental one.

TABLE 1. FITTING PARAMETERS AND R^2 FOR DIFFERENT VIDEO

Test Signal	Logarithmic Function	R^2 factor
Mobile	0.1295ln(x)+0.1274	0.9759
Imax	0.0563ln(x)+0.6411	0.9514
M.I. 3	0.0668ln(x)+0.5747	0.9191
Da Vinci Code	0.0474ln(x)+0.6974	0.8833
Warren	0.0738ln(x)+0.5210	0.9528
Nasa	0.0950ln(x)+0.3892	0.9595
BBC – Africa	0.1098ln(x)+0.2702	0.9875
Superman	0.0282ln(x)+0.8167	0.8859

Based on the aforementioned analysis, we can describe the derived <PQoS>$_{SSIM}$ vs. bit rate curve of each test signal with N total frames, which is encoded at bit rate n from $BitRate_{\min}$ to $BitRate_{\max}$ as a set C, where each element F_n is a triplet, consisting the <PQoS>$_{SSIM}$ of the specific bit

rate, the constants C_1 and C_2, which are derived by the analytical logarithmic expression of Table 1:

$$C_{S-T} @ \{m : (\frac{1}{N}\sum_{i=1}^{N} SSIM(f_i), C_1, C_2)_n = F_n, n \in [BitRate_{\min}, BitRate_{\max}]\}$$

where

- *SSIM( )* is the function that calculates the perceived quality of each frame according to the *SSIM* metric

-*N* the total number of frames f_i that consists the movie *m*

Thus, deriving the sets C_n for various contents, ranging from static to very high Spatial and Temporal (S-T) ones, a reference hyper set RS, containing a range of C_{S-T} sets for specific spatiotemporal levels can be deduced:

$$RS = \{C_{S-T_{Low}}, ..., C_{S-T_{High}}\}$$

Hence, consider an unknown video clip, which is uncompressed and the broadcaster wants to predict its corresponding C_{S-T} set that better describes its perceived quality vs. bit rate curve before the encoding process at a specific quality level. Then, it is defined for all the sets C_{S-T} the Absolute Difference Value (ADV) between the first C_{S-T} triplet element (i.e. the $<PQoS>_{SSIM}$ at a specific encoding $BitRate_i$) and the experimental measurement of the average *SSIM* for the test signal at the encoding bit rate *n*, for which all the reference sets C_{S-T} have been derived, utilizing the logarithmic equations of Table 1:

$$ADV = | F_{BitRate_i} : (\frac{1}{N}\sum_{i=1}^{N} SSIM(f_i)) - F'_{BitRate_i} : (\sum_{i=1}^{N} SSIM(f_i')) |$$

Due to the fact that the additive property is valid, it is concluded that when the *ADV* between the average *SSIM* of the reference $F_{BitRate_i}$ and experimental $F'_{BitRate_i}$ is minimum, then the set C_{S-T}, which contains the triplet element that minimizes the *ADV*, describes better the specific video. Thus, we have successfully approximated the PQoS vs. Bit rate curve of the specific video with actual cost only one testing encoding and assessment at bit rate *n*. Then the service provider can predict analytically through the logarithmic expression all the bit rates that satisfy specific perceived quality levels, without requiring any other testing encoding processes.

Moreover, the proposed technique was also tested on a set of real captured video clips, containing content with duration spanning from 2 minutes up to 10 minutes. These video clips were captured in DV PAL format from common TV programs and then transcoded to MPEG-4/H.264. Applying the proposed model and following exactly the same procedure, the worst case mean error between the experimentally and theoretically derived MPQoS curves for the twenty real captured videos was measured to be approx. 4%. A typical result of this evaluation process is depicted on figure 4, which demonstrates the fitting match between the experimentally derived curve and the predicted one.

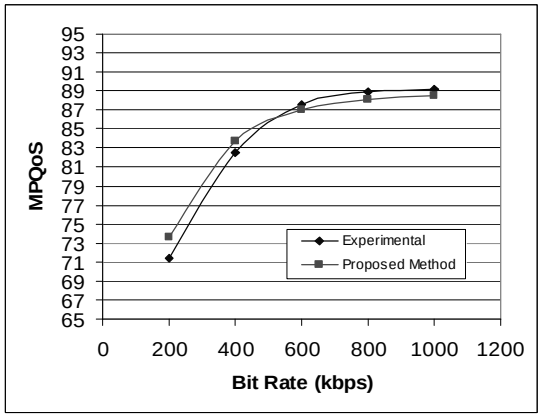


Fig. 4. Comparison of the experimentally and theoretically derived curves

Thus, one only test estimation of the average PQoS at a specific encoding bit rate is adequate for the accurate determination of the MPQoS vs. Bit Rate for a given signal.

In the next section, it is examined the case of the quality degradation during the transmission process of the broadcasting.

IV. Modeling Packet Loss impact on Video Quality

In this section, we discuss the impact of the packet loss during the transmission of a video over a lossy transmission channel. Due to the fact that the frames in a MPEG video sequence are interdependent, considering a packet loss, the visual distortion caused by this packet loss will be not limited only to the frame, on which the specific lost packet belongs to. On the contrary, spatial error propagation will take place, which will infect all the frames that are dependent on the specific frame, on which the loss occurred. Thus, in order to calculate the error propagation due to a packet loss, it must be taken under consideration the interdependencies of the coded frames.

Regarding packetization schemes, all contemporary digital broadcasting systems, including the DVB and ATSC family of standards, are using the MPEG-2 Transport Stream (TS) [36] as the format of baseband data, organized in a statistically multiplexed sequence of fixed-size, 188-byte Transport Packets. Initially intended to convey MPEG-2 encoded audio and video streams, the MPEG-2 TS was eventually used also for the transport of IP traffic, with the adaptation method introduced in [37] and named as Multi Protocol Encapsulation (MPE).

Typical scenarios for fixed-size packetization schemes are a) a packet contains part of one frame, b) a packet contains the end of a frame and c) a packet contains a frame header. Independently of its content, a packet loss will create perceptual degradation and artifacts at the respective decoded frame. Therefore, the initial perceptual error will be

propagated in space and time due to the interdependencies of the encoded frames and the inter-coding procedure of the motion estimation and compensation techniques.

At the user side, the PQoS degradation induced by a packet loss depends on the error concealment strategy used by the decoder. A typical concealment strategy is zero-motion concealment, in which a lost macroblock is concealed using the macroblock in the same spatial location from the closest reference frame.

Therefore, it is expected the visibility of a loss to depend on a complex interaction of its location, the video encoding parameters (i.e. GOP structure) and the underlying characteristics of the video signal itself. In this context, it is proposed a mathematical framework to model the perceptual error propagation caused by packet losses during broadcasting. More specifically, this section studies the packet loss effect on MPEG video decoding quality over error-prone broadcasting channels. We introduce an analytical model, which is used to investigate the video delivered quality and the effect of the packet loss distribution on the delivered video quality.

A. The analytical model of packet loss effect on PQoS

For evaluation purposes of the packet loss impact on the PQoS level of a broadcasting service, it is adopted an objective evaluation metric, known as Decodable Frame Rate (Q). Although the objective Q metric has been used in earlier works [38], our approach is differentiated from the existing ones because it considers the packet loss rate instead of the frame loss rate in the formula, providing a better approach for broadcasting systems. The value of Q lies between 0 and 1.0. The larger the value of Q, the better the video quality received by the end user. Where Q is defined as the fraction of decodable frame rate, which is the number of successfully decoded frames over the total number of frames sent by a video source.

$$Q = \frac{N_{dec}}{(N_{total-I} + N_{total-P} + N_{total-B})}$$

where N_{dec} is the summation of N_{dec-I}, N_{dec-P}, and N_{dec-B}.

Based on this modified Q metric, in the next sub-sections it is analytically calculated the expected numbers of decodable frames per type (i.e. I, B, P) based on a typical structure GOP(12,3), which is widely used in broadcasting applications for moving users due to its robustness characteristics.

In the proposed modeling, the following hypotheses are considered:

- At the decoder it is not implemented any sophisticated error concealment method.
- The decoding threshold is considered equal to one (DT=1), meaning that one packet loss causes significant quality degradation (i.e. unsuccessful decoding) of the respective frame.
- The error propagation affects all the frames that are depended on the frame, where the packet loss took place. Considering that DT=1, the dependent frames are also considered to fail during the decoding procedure.
- The packet loss rate is considered constant during the transmission of the service.

Based on these hypotheses and the modified Q metric, it is clear that the proposed approach of modelling packet loss impact on the degrading percentage of the broadcasting perceived quality of a service is an objective approach. More specifically, it is researched the degradation percentage caused by the transmission packet loss ratio in relevance to the initial quality of the video content. The relative approach provides many advances in comparison to already proposed models, namely the independence from the content dynamics, the coding standard and the structure of the packet.

Following this explanatory section, the proposed model is presented in the next sub-sections, considering constant packet loss ratio p for the whole service duration. For readability purposes, in the appendix of the paper, it can be also found the notation explanation of all the used symbols.

1) The expected number of decodable I frames (N_{dec-I})

In a GOP, the I frame is successfully decodable only if all the packets that belong to the specific frame are intact received. Therefore, the probability that the I frame is successfully decodable is

$$S(I) = (1-p)^{C_I}$$

Consequently, the expected number of correctly decodable I frames for the whole video is

$$N_{dec-I} = (1-p)^{C_I} * N_{GOP}$$

2) The expected number of decodable P frames (N_{dec-P})

In a GOP, P frames are successfully decodable only if the preceding I or P frames are also decodable and all the packets that belong to the P frame under examination have been successfully received. In a GOP, there are N_p P frames, and depending on their position, the probability of a P frame to be decodable is

$$S(P_1) = (1-p)^{C_I} * (1-p)^{C_P} = (1-p)^{C_I + C_P}$$

$$S(P_2) = (1-p)^{C_I} * (1-p)^{C_P} * (1-p)^{C_P} = (1-p)^{C_I + 2C_P}$$

$$\dots\dots$$

$$S(P_{N_P}) = (1-p)^{C_I} * (1-p)^{N_P * C_P} = (1-p)^{C_I + N_P * C_P}$$

Thus, the expected number of successfully decodable P frames for the whole video is

$$N_{dec-P} = (1-p)^{C_I} * \sum_{j=1}^{N_P} (1-p)^{jC_P} * N_{GOP}$$

3) The expected number of decodable B frames (N_{dec-B})

In a GOP, B frames are decodable only if the preceding and succeeding I or P frames are both decodable and all the respective packets that consist the specific B frame have been successfully received. Considering that B frames throughout

the GOP structure have the same dependencies, we examine the consecutive B frames as composing a B group, except for the last B frame in a GOP, which is dependent from the preceding P frame and succeeding I frame (making it straight forward dependent on two successive I frames). In a GOP, the probability of the B frame that is decodable is

$$S(B_1) = (1-p)^{C_I} * (1-p)^{C_P} * (1-p)^{C_B}$$

$$S(B_2) = (1-p)^{C_I} * (1-p)^{2C_P} * (1-p)^{C_B}$$

$\ldots\ldots$.

$$S\left(B_{\frac{N}{M}-1}\right) = (1-p)^{C_I} * (1-p)^{\left(\frac{N}{M}-1\right)*C_P} * (1-p)^{C_B}$$

$$S\left(B_{\frac{N}{M}}\right) = (1-p)^{2C_I} * (1-p)^{\left(\frac{N}{M}-1\right)*C_P} * (1-p)^{C_B}$$

Thus, the expected number of correctly decodable B frames for the whole video is

$$N_{dec\text{-}B} = (M-1)*\sum_{j=1}^{\frac{N}{M}} S(B_j) * N_{GOP}$$

$$= \left[(M-1) * (1-p)^{C_I} * \sum_{j=1}^{N_P} (1-p)^{jC_P} * (1-p)^{C_B} + (M-1) * (1-p)^{2C_I} * (1-p)^{N_P C_P} * (1-p)^{C_B} \right] * N_{GOP}$$

$$= \left[(1-p)^{C_I + N_P C_P} + \sum_{j=1}^{N_P} (1-p)^{jC_P} \right] * (M-1)*(1-p)^{C_I + C_B} * N_{GOP}$$

Based on the aforementioned proposed estimations of successfully decodable frames for each frame type, the modified Q metric becomes:

$$Q = \frac{N_{dec}}{(N_{total\text{-}I} + N_{total\text{-}P} + N_{total\text{-}B})} = \frac{N_{dec\text{-}I} + N_{dec\text{-}P} + N_{dec\text{-}B}}{(N_{total\text{-}I} + N_{total\text{-}P} + N_{total\text{-}B})} \Rightarrow$$

$$Q = \frac{(1-p)^{C_I} * N_{GOP} + (1-p)^{C_I} * \sum_{j=1}^{N_P} (1-p)^{jC_P} * N_{GOP} + \left[(1-p)^{C_I + N_P C_P} + \sum_{j=1}^{N_P} (1-p)^{jC_P} \right] * (M-1)*(1-p)^{C_I + C_B} * N_{GOP}}{(N_{total\text{-}I} + N_{total\text{-}P} + N_{total\text{-}B})}$$

Therefore, considering a transmission channel with constant packet loss ratio p, the respective Q rate of successfully decoded frames (i.e. frames without containing any perceptual degradation) can be analytically estimated. In other words, the proposed model provides a degradation parameter, which acts in a relative way to the initial quality level of the broadcasting service.

B. Experimental Evaluation of the Proposed Model

The proposed model of packet loss impact on the PQoS degradation of the transmitted video is experimentally evaluated considering two discrete packet loss schemes: The random uniform model, which provides the distributed losses with the mean loss rate (p) and the Gilbert-Elliot (GE) model [39], which provides for the same percentage rate, the packet losses grouped in bursts, approximating by this way the behavior of real wireless error-prone transmission channels.

For clarity purposes, Figure 5 provides a graphical representation of the used packet loss schemes.

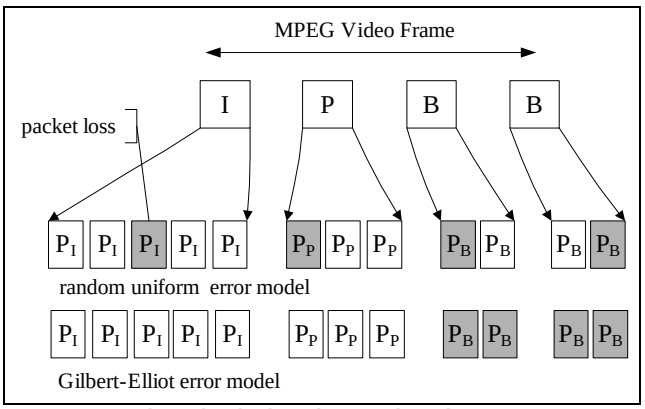


Fig. 5. The used packet loss schemes in the evaluation process

The experiments were performed on NS-2. For the evaluation purposes, the video trace "Aladdin" was selected, which is composed of 89998 video frames, including 7500 I frames, 22500 P frames, and 59998 B frames at QCIF MPEG-4/H.264 format and GOP(12,3).

TABLE 2
STATISTICS OF TEST SIGNAL 'ALLADIN'

	Total	I frame	P frame	B frame
Number of frames	89998	7500	22500	59998
Number of packets	1086789	195010	321444	570335
C_I, C_P, C_B	N/A	26.001	14.286	9.506

Table 2 contains the statistics for the test signal, considering 188 bytes transmission, which is consistent with the MPEG-2 TS and the DVB-H standard.

For both loss schemes under test, the packet loss rate ranges from 0.02 to 0.2, considering intervals of 0.02 and transmitting packet size equal to 188 bytes.

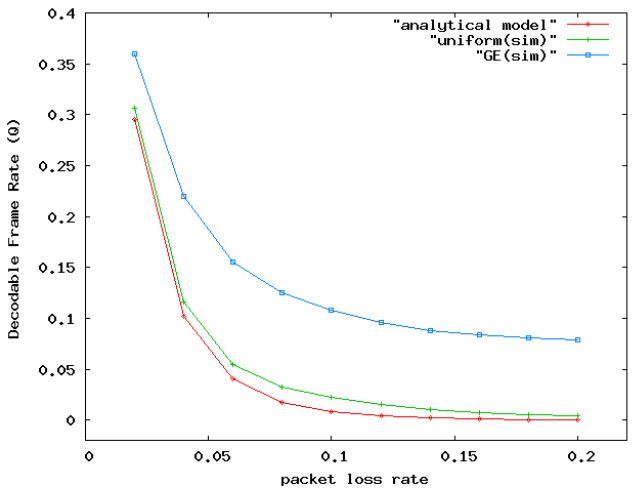


Fig. 6. Comparison of the results derived from the simulation and the analytical model.

Figure 6 shows the successfully decodable frame ratio for varying packet loss rates under the uniform and G-E packet loss schemes. Considering that the uniform distribution corresponds to the theoretically worst case scenario for decoders with DT=1 and that appears to have a significant good match between the theoretically expected video quality degradation curve and the corresponding experimentally derived one, the validity of the proposed model has been proved. For the case of G-E distribution, it is shown that the effect of burst packet losses during the transmission on the delivered video quality causes less severe degradation than the equivalent uniformly distributed case.

Moreover, in both of these models the video quality of simulation is better than analytical model. Hence, the analytical model provides the predicted bounds of the quality of the MPEG video transmission over a lossy transmission channel.

V. The Proposed End-To-End Framework

Based on the aforementioned proposed theoretical models of video quality prediction at a pre-encoding state and packet loss modeling, this section proposes an end-to-end video quality assessment framework of MPEG-based audiovisual broadcasting services for hand-held and mobile wireless broadcast systems, which is based on the combination and exploitation of the two proposed models.

For demonstrating purposes of the proposed end-to-end framework, we consider that a hypothetical Content Provider wants to broadcast a music video clip at various quality levels and possesses the reference hyper set RS, containing the C_{S-T} sets derived from the test signals of Table 1. Initially, the music clip under examination is encoded at MPEG-4/H.264 CIF 100 kbps. Then, the resulted encoded clip is used as input to the *SSIM* algorithm and the resulted instant *SSIM* curve is used for the estimation of the <*SSIM*> value, which is estimated equal to 0.8. Afterwards, using this value as input in the ADV equation, it is defined the C_{S-T} that minimizes the ADV and therefore contains the optimal triplet element for the analytical description of the signal under test. More specifically, the derived <*SSIM*> value, the optimal C_{S-T} set belongs to *BBC Africa* reference clip. Thus, the equation that describes better the variation of the $<PQoS>_{SSIM}$ vs. the bit rate is

$$<PQoS>_{SSIM} = 0.1098\ln(\text{Bit Rate}) + 0.2702$$

Consequently, if the content provider wishes to offer this video clip at the perceptual qualities 0.70, 0.80 and 0.90, then by using the above equation is able to estimate the corresponding bit rates in a pre-encoding process. Table 3 shows the corresponding encoding bit rate values for the specific video clip.

Table 3
Predicted Bit Rate Values for Specific Quality Levels

$<PQoS>_{SSIM}$	BR (Kbps)
0.7	50.12
0.8	124.60
0.9	309.79

Afterwards, considering that a monitoring system provides the average packet loss rate at the transmission channel and it is for example 0.02, then it can be predicted from the packet loss model (see Figure 6) that the worst case degradation percentage is that the end-user will experience video quality degradation for the 70% of the total duration of the sequence. For the rest 30%, the user will watch normal playback without any perceived artifacts. Thus, if the Content Provider would like to calculate a representative value of the Expected Delivered Video Quality (EDVQ) level at the content consumer, the following equation is proposed:

$$EDVQ = (Initial\_Video\_Quality) * (Percentage\_of\_Succesfully\_Decoded\_Frames)$$

where the objective metric Q of the proposed mapping model is used as degradation multiplier to the initial perceived quality level, which has been specified pre-encodingly by the proposed prediction model. Therefore, the combination of the discrete two models provides a prediction for the worst case degradation scenario, if error concealments methods are not taken under consideration and the D.T. is considered equal to 1.0.

VI. Conclusion

This paper presents a theoretical framework for end-to-end video quality prediction for MPEG-based broadcasting services.

The proposed framework encloses two discrete models: i) a model for predicting the video quality of an encoded signal at a pre-encoding stage and ii) a model for mapping packet loss ratio of the transmitting channel to video quality degradation. The efficiency of both discrete models has been experimentally validated, proving by this way the accuracy of the proposed framework, which combines the discrete models into a common end-to-end video quality assessment framework.

The advances of the proposed framework are its generic nature, since it can be applied on MPEG-based encoded sequences, independently of the selected encoding parameters, subject to specific GOP structure. Moreover, it is also introduced the novel issue of predicting the video quality of an encoded service at a pre-encoding state, which provides new facilities at the broadcaster side. Also, by applying the randomly uniform packet loss model, the proposed framework overpasses any stochastic predicaments in mapping the packet loss ratio to video quality degradation, since it calculates and demonstrates the worst case scenario.

ACKONWLEDGEMENT

This paper is an invited extended version of the conference paper H. Koumaras, A. Kourtis, C-H Lin, C-K Shieh, "Theoretical Framework for End-to-End Video Quality Prediction of MPEG-based Sequences" published in ICNS 2007.

Part of the work in this paper has been performed within the research framework of FP7 ICT-214751 ADAMANTIUM Project.

APPENDIX

NOTATIONS USED IN THE PAPER

$N_{total\text{-}I}\ N_{total\text{-}P}\ N_{total\text{-}B}$	The total number of each type of frames.
$N_{dec\text{-}I}\ N_{dec\text{-}P}\ N_{dec\text{-}B}$	The number of decodable frames in each type.
N_{dec}	The total number of decodable frames in the video flow.
N_{GOP}	The total number of GOPs in the video flow.
$C_I\ C_P\ C_B$	The mean number of packets that transport the data of each frame type
p	Packet loss rate

REFERENCES

[1] Wang, Z., H.R. Sheikh, and A.C. Bovik, Objective video quality assessment, in The Handbook of Video Databases: Design and Applications, B. Furht and O. Marqure, Editors. 2003, CRC Press. p. 1041-1078.

[2] VQEG. Final Report From the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment. 2000. Available : http://www.vqeg.org.

[3] Wang, Z., A.C. Bovik, and L. Lu. Why is image quality assessment so difficult? in IEEE International Conference on Acoustics, Speech, and Signal Processing. 2002.

[4] Ulrich Engelke and Hans-JÄurgen Zepernick, "Perceptual-based Quality Metrics for Image and Video Services: A Survey", 3rd EuroNGI Conference on Next Generation Internet Networks, Trondheim, Norway, 21-23 May 2007

[5] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, Apr. 2004.

[6] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measurement," Signal Processing: Image Communication, special issue on "Objective video quality metrics", vol. 19, no. 2, pp. 121-132, Feb. 2004.

[7] M. Ries, C. Crespi, O. Nemethova, M. Rupp, "Content Based Video Quality Estimation for H.264/AVC Video Streaming", in Proc. Proceedings of IEEE Wireless and Communications & Networking Conference, Hong Kong, March, 2007

[8] Eric A. Silva, Karen Panetta, Sos S Agaian, "Quantifying image similarity using measure of enhancement by entropy", Mobile Multimedia/Image Processing for Military and Security Applications 2007, Sos S. Agaian, Sabah A. Jassim, Editors, 65790U, Proceedings of SPIE -- Volume 6579, May. 2, 2007

[9] Gunawan, I.P. and M. Ghanbari. Reduced-Reference Picture Quality Estimation by Using Local Harmonic Amplitude Information. in London Communications Symposium 2003. 2003.

[10] M. Montenovo, A. Perot, M. Carli, P. Cicchetti, A. Neri, Objective evaluation of video services. Proc. of 2nd Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics, 2006.

[11] S. S. Hemami, M. A. Masry, A scalable video quality metric and applications. Proc. of 1st Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics, 2005.

[12] O. A. Lotfallah, M. Reisslein, S. Panchanathan, A framework for advanced video traces: Evaluating visual quality for video transmission over lossy networks. (Article ID 42083) EURASIP Journal on Applied Signal Processing, 2006. 2006.

[13] Zhou Wang, Guixing Wu, Hamid R. Sheikh, Eero P. Simoncelli, En-Hui Yang and Alan C. Bovik, Quality-Aware Images. IEEE Transactions on Image Processing.

[14] H. R. Wu, M. Yuen, A generalized block-edge impairment metric for video coding. IEEE Signal Processing Letters, 1997. 4(11): p. 317-320.

[15] P. Marziliano, F. Dufaux, S. Winkler, T. Ebrahim, A no-reference perceptual blur metric. in Proc. of IEEE Int. Conf. on Image Processing, 2002. 3: p. 57-60.

[16] J. Caviedes, S. Gurbuz, No-reference sharpness metric based on local edge kurtosis. in Proc. of IEEE Int. Conf. on Image Processing, 2002. 3: p. 53-56.

[17] A. Cavallaro, S. Winkler, Segmentation-driven perceptual quality metrics. in Proc. of IEEE Int. Conf. on Image Processing, 2004. 5: p. 3543-3546.

[18] R. R. Pastrana-Vidal, J. C. Gicquel, Automatic quality assessment of video fluidity impairments using a no-reference metric. in Proc. of 2nd Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics, 2006.

[19] M. C. Q. Farias, S. K. Mitra, No-reference video quality metric based on artifact measurements. in Proc. of IEEE Int. Conf. on Image Processing, 2002. 3: p. 141-144.

[20] X. Marichal, W. Y. Ma, H. J. Zhang, Blur determination in the compressed domain using DCT information. in Proc. of IEEE Int. Conf. on Image Processing, 2002. 2: p. 386-390.

[21] R. Ferzli, L. J. Karam, J. Caviedes, A robust image sharpness metric based on kurtosis measurement of wavelet coefficients,. Proc. of 1st Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics, 2005.

[22] X. Marichal, W. Y. Ma, H. J. Zhang, Blur determination in the compressed domain using DCT information. in Proc. of IEEE Int. Conf. on Image Processing, 2002. 2: p. 386-390.

[23] S. Liu, A. C. Bovik, Effcient dct-domain blind measurement and reduction of blocking artifacts. IEEE Transactions on Circuits and Systems for Video Technology, 2002. 12(12): p. 1139-1149.

[24] M. Ries, O. Nemethova, M. Rupp, Reference-free video quality metric for mobile streaming applications. in Proc. of 8th Int. Symp. on DSP and Communication Systems & 4th Workshop on the Internet, Telecommunications and Signal Processing, 2005: p. 98-103.

[25] Lu, L., et al. Full-reference video quality assessment considering structural distortion and no-reference quality evaluation of MPEG video. in IEEE International Conference on Multimedia. 2002.

[26] H. Koumaras, A. Kourtis, D. Martakos, "Evaluation of Video Quality Based on Objectively Estimated Metric", Journal of Communications and Networking, Korean Institute of Communications Sciences (KICS), Vol. 7, No.3, pp.235-242, Sep 2005.

[27] H. Koumaras, A. Kourtis, D. Martakos, J. Lauterjung, "Quantified PQoS Assessment Based on Fast Estimation of the Spatial and Temporal Activity Level", Multimedia Tools and Applications, Springer Editions Vol. 34(3), pp. 355-374, September 2007.

[28] H. Koumaras, E. Pallis, G. Xilouris, A. Kourtis, D. Martakos, J. Lauterjung, "Pre-Encoding PQoS Assessment Method for Optimized Resource Utilization", 2nd Inter. Conference on Performance Modelling and Evaluation of Heterogeneous Networks, Het-NeTs04, Ilkley, United Kingdom, 2004.

[29] S. Kanumuri, P. C. Cosman, A.R. Reibman, V.A. Vaishampayan, "Modeling Packet-Loss Visibility in MPEG-2 Video", IEEE transactions on Multimedia, Vol.8, No.2, pp.341-355, April 2006.

[30] Z. He, H. Xong, "Transmission Distortion Analysis for Real-Time Video Encoding and Streaming over Wireless Networks", IEEE Transactions on Circuits and Systems for Video Technology, Vol.16, No.9, pp.1051-1062, September 2006

[31] J. Mitchell and W. Pennebaker. MPEG Video: Compression Standard. Chapman and Hall, 1996. ISBN 0412087715

[32] Cheng-Han Lin, Chih-Heng Ke, Ce-Kuen Shieh, Naveen Chilamkurti, "The Packet Loss Effect on MPEG Video Transmission in Wireless Networks", The IEEE 20th International Conference on Advanced Information Networking and Applications (AINA'06), April 18-20, 2006, Vienna, Austria

[33] A. Ziviani, B. E. Wolfinger, J. F. Rezende, O. C. M. B. Duarte, and S. Fdida, "*Joint Adoption of QoS Schemes for MPEG Streams*," Multimedia Tools and Applications Journal, to appear.

[34] J. P.Ebert, A.Willig, *A Gilbert-Elliot Bit Error Model and the Efficient Use in Packet Level Simulation*, Technical Repoert, TKN-99-002, Technical University of Berlin, March 1999.

[35] C.-H.-Ke, C.-H.-Lin, C.-K. Shieh, and W.-S. Hwang, "A Novel Realistic Simulation Tool for Video Transmission over Wireless Network," presented at The IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC2006), Taiwan, 2006.

[36] ISO/IEC 13818-1, Generic Coding of Moving Pictures and Associated Audio Information (MPEG-2) Part 1: Systems, 1996

[37] ETSI EN 301 192, Digital Video Broadcasting (DVB): DVB Specification for data broadcasting, European Standard, v.1.4.1, Nov.2004

[38] A. Ziviani, B. E. Wolfinger, J. F. Rezende, O. C. M. B. Duarte, and S. Fdida, "Joint Adoption of QoS Schemes for MPEG Streams," Multimedia Tools and Applications Journal, vol. 26, no. 1, pp. 59-80, May 2005.

[39] J. P.Ebert, A.Willig, A Gilbert-Elliot Bit Error Model and the Efficient Use in Packet Level Simulation, Technical Repoert, TKN-99-002, Technical University of Berlin, March 1999.

[40] H. Koumaras, A. Kourtis, C-H Lin, C-K Shieh, A Theoretical Framework for End-to-End Video Quality Prediction of MPEG-based Sequences, Third International Conference on Networking and Services ICNS07, 19-25 June 2007 Page(s):62 – 62, Athens, Greece 2007.

Harilaos Koumaras was born in Athens, Greece in 1980. He received his BSc degree in Physics in 2002 from the University of Athens, Physics Department, his MSc in Electronic Automation and Information Systems in 2004, being scholar of the non-profit organization Alexander S Onassis, from the University of Athens, Physics and Informatics Department and his PhD in 2007 on digital video quality prediction from the University of Athens, Informatics Department, having granted the four-year scholarship of NCSR "Demokritos". He has received twice the Greek State Foundations (IKY) scholarship during the academic years 2000-01 and 2003-04. He has also granted with honors the classical piano and harmony degrees from the classical music department of Attiko Conservatory. He joined the Digital Telecommunications Lab at the National Centre of Scientific Research "Demokritos" in 2003 and since then he has participated in EU-funded and national funded projects with presentations and publications at international conferences, scientific journals and book chapters. At the same time, he is an associate lecturer at the Business College of Athens (BCA) and City University of Seattle, teaching modules related to Information Technology, Data Networks and Mathematics. His research interests include objective/subjective evaluation of the perceived quality of multimedia services, video quality and picture quality evaluation, video traffic modeling, digital terrestrial television and video compression techniques. Currently, he is the author or co-author of more than 30 scientific papers in international journals, technical books and book chapters, numbering 41 non-self citations. He is an editorial board member of Telecommunications Systems Journal and a reviewer of EURASIP Journal of Applied Signal Processing and IEEE Transactions on Broadcasting. Dr. Koumaras is a member of IEEE, SPIE and National Geographic Society.

Cheng-Han Lin is currently a Ph.D. candidate studying in the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan. Lin received his MS and BS degree from the Electrical Engineering Department of National Chung Cheng University in 2002 and 2004. His current research interests include wireless MAC protocols, multimedia communications, and QoS network.

Ce-Kuen Shieh is currently a professor teaching in the Department of Electrical Engineering, National Cheng Kung University. He received his PhD, MS, and BS degrees from the Electrical Engineering Department of National Cheng Kung University, Tainan, Taiwan. His current research areas include distributed and parallel processing systems, computer networking, and operating systems.

Anastasios Kourtis received his B.S. degree in Physics in 1978 and his Ph.D. in Telecommunications in 1984, both from the University of Athens. Since 1986, he has been a researcher in the Institute of Informatics and Telecommunications of the National Centre for Scientific Research "Demokritos", currently ranking as Senior Researcher. His current research activities include, digital terrestrial interactive television, broadband wireless networks, Perceived Quality of video services, end to end QoS and real time bandwidth management in satellite communications. He is author or co-author of more than 80 scientific publications in international scientific journals, edited books and conference proceedings. Dr. Kourtis has a leading participation in many European Union funded research projects in the frame of IST/FP5/FP6 (MAMBO, SOQUET, CREDO, WIN, LIAISON, ENTHRONE). He has also coordinated three European funded Specific Targeted Research Projects (REPOSIT, ATHENA, IMOSAN).

Fast Convergence Least-Mean-Square Algorithms for MMSE Receivers in DS-CDMA Systems

Constantin Paleologu[1], *Călin Vlădeanu*[1], *and Safwan El Assad*[2]

[1]Department of Telecommunications, University Politehnica of Bucharest
Bucharest, Romania
{pale, calin}@comm.pub.ro
[2]IREENA, Ecole Polytechnique de l'Université de Nantes
Nantes, France
safwan.elassad@polytech.univ-nantes.fr

***Abstract*—This paper considers a minimum mean-squared error (MMSE) single user adaptive receiver for the asynchronous direct-sequence code-division multiple-access (DS-CDMA) system, based on the least-mean-square (LMS) algorithm. It is known that in this context the adaptive algorithm can be iterated several times during the same bit interval in order to achieve a faster convergence rate, which further reduces the length of the training sequence. The objective of this paper is twofold. First, instead of using such multiple iterations, we propose a single equivalent formula for updating the receiver coefficients, saving significant time processing. Secondly, in order to further increase the convergence rate, a division-free version of the gradient adaptive lattice (GAL) algorithm is proposed. Since the lattice predictor orthogonalizes the input signals, this algorithm achieves a faster convergence rate than the transversal LMS algorithm.**

Keywords: Code-division multiple-access (CDMA) system; gradient adaptive lattice (GAL) algorithm; least-mean-square (LMS) algorithm; minimum mean-squared error (MMSE) receiver.

I. INTRODUCTION

There are a lot of mobile communications systems that employ the code-division multiple-access (CDMA) technique, where the users transmit simultaneously within the same bandwidth by means of different code sequences. This technique has been found to be attractive because of such characteristics as potential capacity increases over competing multiple access methods, anti-multipath capabilities, soft capacity, narrow-bandwidth anti-jamming, and soft handoff. In the Direct Sequence CDMA (DS-CDMA) system [1], each code sequence is used to spread the user data signal over a larger bandwidth and to encode the information into a random waveform. A simple multiplication between the data signal and the code sequence waveform is needed, and the resulted signal inherits its spectral characteristics from the spreading sequence. Due to its linear signal processing function this scheme may be a subject for possible performance improvements by developing new signal processing techniques for the receiver.

In DS-CDMA systems the conventional matched filter receiver distinguishes each user's signal by correlating the received multi-user signal with the corresponding signature waveform. The data symbol decision for each user is affected by multiple-access interference (MAI) from other users and by channel distortions. Hence, the conventional matched filter receiver performances are limited by its original purpose. It was designed to be optimum only for a single user channel where no MAI is present and to be optimum for a perfect power control, so it suffers from the near-far problem. Motivated by these limitations, adaptive minimum mean-squared error (MMSE) receivers have been introduced [2], [3]. The principle consists of a single user detector that works only with the bit sequence of that user. In this case, the detection process is done in a bit by bit manner, and the final decision is taken for a single bit interval from the received signal. The complexity of an adaptive MMSE receiver is slightly higher than that of a conventional receiver, but with superior performance. Besides its facile implementation the adaptive MMSE receiver has the advantage that it needs no supplementary information during the detection process.

The "brain" of an adaptive MMSE receiver is the adaptive algorithm. There are two major categories of such algorithms [4]. The first one contains the algorithms based on the mean square error minimization, whose representative member is the least-mean-square (LMS) algorithm. The second category of algorithms uses an optimization procedure in the least-squares (LS) sense, and its representative is the recursive-least-squares (RLS) algorithm. The LMS algorithm with its simple implementation suffers from slow convergence, which implies long training overhead with low system throughput. On the other hand, LS algorithms offer faster convergence rate, paying with increased computational complexity and numerical stability problems. Due to these reasons, LMS based algorithms are still preferred in practical implementations of adaptive MMSE receivers. Lattice structures have also been considered for this type of applications [5], [6], [7]. Since the lattice predictor orthogonalizes the input signals, the gradient adaptation algorithms using this structure are less dependent on the eigenvalues spread of the input signal and may converge faster than their transversal counterparts. The computational complexity of these algorithms is between transversal LMS and LS algorithms. In addition, several simulation examples and also numerical comparison of the analytical results have shown

This work was supported by the UEFISCSU Romania under Grant PN-II-PCE no. 331/01.10.2007.

that adaptive lattice filters have better numerical properties than their transversal counterparts [8], [9]. Moreover, stage-to-stage modularity of the lattice structure has benefits for efficient hardware implementations.

A solution for increasing the convergence rate of the MMSE receiver is to adjust the filter tap weights iteratively several times every transmitted bit interval [10]. Moreover, this procedure can be combined with a parallel interference cancellation (PIC) mechanism for further reducing the multiple access interference (MAI) [11]. A drawback of these approaches is that they are time consumers. During a bit interval, every single iteration has to "wait" the result from the previous one, which is the natural function mode for every iterative process. Anyway, from the time processing reason, it would be more convenient to use a single formula instead of those multiple iterations. A first objective of this paper is to propose a relation for updating the LMS adaptive filter coefficients, which is equivalent with the multiple iterations algorithm [12]. We will demonstrate that the multiple iterations process is equivalent with an unique iteration with a particular step size of the algorithm. Secondly, for further increasing the convergence rate, a lattice MMSE receiver based on a division-free gradient adaptive lattice (GAL) algorithm [13] is developed.

The paper is organized as follows. In Section II we briefly describe the asynchronous DS-CDMA system model. The analytical expression equivalent with the multiple iterations of the LMS algorithm is developed in Section III. Section IV is focused on the lattice receiver, revealing in this context the division-free GAL algorithm. The experimental results are presented in Section V. Finally, Section VI concludes this work.

II. DS-CDMA System Model

In the transmitter part of the DS-CDMA system, each user data symbol is modulated using a unique signature waveform $a_i(t)$, with a normalized energy over a data bit interval T, $\int_0^T \|a_i(t)\|^2 dt = 1$, given by $a_i(t) = \sum_{j=1}^{N} a_i(j) p_c(t - jT_c)$ [1], with $i = 1, \ldots, K$, where K is the number of users in the system. Parameter $a_i(j)$ represents the jth chip of the ith user's code sequence and $p_c(t)$ is the chip pulse waveform defined over the interval $[0; T_c)$, with T_c as the chip duration (it is related to the bit duration through the processing gain N by $T_c=T/N$). In the following analysis we consider binary-phase shift keying (BPSK) transmission. The ith user transmitted signal is [1]

$$s_i(t) = \sqrt{2P_i} b_i(t) a_i(t) \cos(\omega_0 t + \theta_i), \; i = \overline{1, \; K} \qquad (1)$$

where $b_i(t) = \sum_{m=1}^{N_b} b_i(m) p(t - mT)$ is the binary data sequence for ith user (N_b is the number of received data bits), with $b_i(m) \in \{-1, \; +1\}$, P_i is the ith user bit power, ω_0 and θ_i represent the common carrier pulsation and phase, respectively.

After converting the received signal to its baseband form using a down converter, the received signal is given by [1]:

$$r(t) = \sqrt{\frac{P_i}{2}} \sum_{i=1}^{K} b_i(t - \tau_i) a_i(t - \tau_i) \cos(\theta_i) + n(t) \cos(\omega_0 t) \quad (2)$$

where $n(t)$ is the two-sided power spectral density $N_0/2$ additive white Gaussian noise. The asynchronous DS-CDMA system consists of random initial phases of the carrier $0 \le \theta_i < 2\pi$ and random propagation delays $0 \le \tau_i < T$ for all the users $i = \overline{1, \; K}$. There is no loss of generality to assume that $\theta_k = 0$ and $\tau_k = 0$ for the desired user k, and to consider only $0 \le \tau_i < T$ and $0 \le \theta_i < 2\pi$ for any $i \ne k$ [2]. Assuming perfect chip timing at the receiver, the received signal from (2) is passed through a chip-matched filter followed by sampling at the end of each chip interval to give for the mth data bit interval:

$$r_{m,\,l} = \int_{mT + lT_c}^{mT + (l+1)T_c} r(t) p(t - lT_c) dt, \quad l = 0, \; 1, \; \ldots, \; N - 1 \quad (3)$$

where $p(t)$ is the chip pulse shape, which is taken to be a rectangular pulse with amplitude $1/\sqrt{N}$. Using (3) and taking the kth user as the desired one, the output of the chip matched filter after sampling for the mth data bit is given by:

$$\begin{aligned} r_{m,l} &= \sqrt{\frac{P_k}{2N}} T_c b_k(m) a_k(l) + \\ &+ \frac{1}{\sqrt{2N}} \sum_{\substack{i=1 \\ i \ne k}}^{K} \sqrt{P_i} \cos\theta_i b_i(m) I_{i,k}(m,l) + n(m,l) \end{aligned} \qquad (4)$$

where

$$I_{i,k}(m,l) = \begin{cases} b_i(m-1)[\varepsilon_i a_i(N - 1 - N_i + l) + \\ \quad + (T_c - \varepsilon_i) a_i(N - N_i + l)], & 0 \le l \le N_i - 1 \\ b_i(m-1) \varepsilon_i a_i(N-1) + \\ \quad + b_i(m)(T_c - \varepsilon_i) a_i(0), & l = N_i \\ b_i(m)[\varepsilon_i a_i(l - N_i - 1) + \\ \quad + (T_c - \varepsilon_i) a_i(l - N_i)], & N_i + 1 \le l \le N - 1 \end{cases} \qquad (5)$$

with $\tau_i = N_i T_c + \varepsilon_i$, $0 \le N_i \le N - 1$, $0 < \varepsilon_i < T_c$.

A block diagram of the transversal MMSE receiver structure is depicted in Fig. 1. In the training mode, the receiver adapts its coefficients using a short training sequence employing an adaptive algorithm. After training is acquired, the receiver switches to the decision-directed mode and continues to adapt and track channel variations.

Let us consider the following vectors:

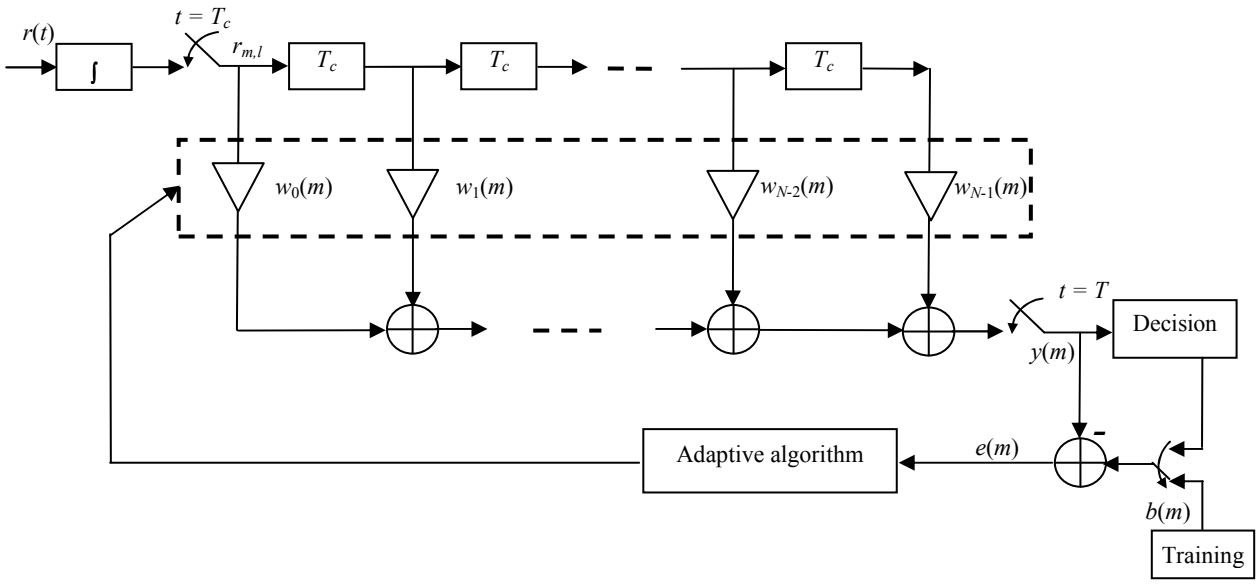


Figure 1. Transversal MMSE receiver scheme

$$\mathbf{w}(m) = [w_0(m), w_1(m), ..., w_{N-1}(m)]^T$$
$$\mathbf{r}(m) = [r_{m,0}, r_{m,1}, \ldots, r_{m,N-1}]^T \quad , \qquad (6)$$

with $r_{m,l}$ given by (4). The output signal $y(m)$ will be an estimate of $b(m)$. For the estimation of $\mathbf{w}(m)$ a stochastic-gradient approach based on the LMS adaptive algorithm is used. The output signal is $y(m)=\mathbf{w}^T(m)\mathbf{r}(m)$. The receiver forms an error signal $e(m) = b(m) - y(m)$ and a new filter tap weight vector is estimated according to

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \mu\mathbf{r}(m)e(m) \, . \qquad (7)$$

The parameter μ represents the adaptation step size of the algorithm; its value is based on a compromise between fast convergence rate and low mean squared error [4].

A solution to increase the overall performances of the MMSE receiver is to adjust the filter tap weights iteratively several times every transmitted bit interval [10]. The coefficients obtained after the Gth iteration of the mth data bit is used by the algorithm in the first iteration of the (m+1)th data bit. It is obvious that this multiple iterations process will increase the computational complexity. This procedure can be combined with a PIC mechanism for further reducing the MAI [11]. This approach also makes use of the available knowledge of all users' training sequences at the base-station receiver to jointly cancel MAI and adapts to the MMSE optimum filter taps using the combined adaptive MMSE/PIC receiver.

III. MULTIPLE ITERATIONS LMS ALGORITHM

It can be noticed that the LMS adaptive algorithm used for MMSE receiver does not work in a sample-by-sample manner (chip-by-chip), as usually adaptive filters works, but in a block-by-block mode (bit-by-bit).

Therefore, by using multiple iterations for every input data bit the converge rate of the adaptive algorithm increases. Nevertheless, this process is time consumer and it would be more convenient to use a one step formula instead of those multiple iterations per bit. Using the notations from the previous section, the multiple iterations LMS algorithm (without PIC mechanism) for the mth input data bit can be resume as follows:

for $g = 1, 2, \ldots, G$ (G is the number of iterations per bit)

$$y^{(g)}(m) = \mathbf{w}^{(g)^T}(m)\mathbf{r}(m)$$
$$e^{(g)}(m) = b(m) - y^{(g)}(m)$$
$$\mathbf{w}^{(g+1)}(m) = \mathbf{w}^{(g)}(m) + \mu\mathbf{r}(m)e^{(g)}(m)$$

end

As we have mentioned in the previous section, the first iteration for the (m+1)th data bit begins with the initial values of the filter coefficients given by $\mathbf{w}^{(1)}(m+1) = \mathbf{w}^{(G+1)}(m)$. In order to simplify the presentation, we renounce at the temporal index m and we will use the following notations: $\mathbf{r}(m) = \mathbf{r}$, $b(m) = b$, $\mathbf{w}^{(1)}(m) = \mathbf{w}_0$, $y_0 = \mathbf{w}_0^T\mathbf{r}$, $e^{(1)}(m) = b - y_0 = e_0$, $s = \mathbf{r}^T\mathbf{r}$. In the first three iteration we have:

$g = 1$: $e^{(1)}(m) = b - y_0 = e_0$

$\mathbf{w}^{(2)} = \mathbf{w}_0 + \mu\mathbf{r}e_0$

$g = 2$: $e^{(2)} = b - \mathbf{w}^{(2)^T}\mathbf{r} = b - (\mathbf{w}_0 + \mu\mathbf{r}e_0)^T\mathbf{r} = e_0(1 - \mu s)$

$\mathbf{w}^{(3)} = \mathbf{w}^{(2)} + \mu\mathbf{r}e^{(2)} = \mathbf{w}_0 + \mu\mathbf{r}e_0(2 - \mu s)$

$g = 3$:

$$e^{(3)} = b - \mathbf{w}^{(3)T}\mathbf{r} = b - \left(\mathbf{w}_0 + \mu\mathbf{r}e_0\left(2 - \mu s\right)\right)^T \mathbf{r} =$$
$$= e_0\left(1 - 2\mu s + \mu^2 s^2\right)$$
$$\mathbf{w}^{(4)} = \mathbf{w}^{(3)} + \mu\mathbf{r}e^{(3)} = \mathbf{w}_0 + \mu\mathbf{r}e_0\left(3 - 3\mu s + \mu^2 s^2\right)$$

By performing simple polynomial operations we obtain in a similar manner:

$$\mathbf{w}^{(5)} = \mathbf{w}_0 + \mu\mathbf{r}e_0\left(4 - 6\mu s + 4\mu^2 s^2 - \mu^3 s^3\right)$$
$$\mathbf{w}^{(6)} = \mathbf{w}_0 + \mu\mathbf{r}e_0\left(5 - 10\mu s + 10\mu^2 s^2 - 5\mu^3 s^3 + \mu^4 s^4\right)$$
$$\mathbf{w}^{(7)} = \mathbf{w}_0 + \mu\mathbf{r}e_0\left(6 - 15\mu s + 20\mu^2 s^2 - 15\mu^3 s^3 + 6\mu^4 s^4 - \mu^5 s^5\right)$$

Let us denote $-\mu s = \alpha$. It can be noticed that

$$\mathbf{w}^{(G+1)} = \mathbf{w}_0 + \mu\mathbf{r}e_0 F\left(\alpha\right), \tag{8}$$

where

$$F\left(\alpha\right) = \sum_{k=0}^{G-1} f_k^{(G)}\alpha^k = \mathbf{f}^{(G)T}\boldsymbol{\alpha}^{(G)}. \tag{9}$$

The column vectors from (9) are

$$\mathbf{f}^{(G)} = \left[f_0^{(G)}, f_1^{(G)}, \ldots, f_{G-1}^{(G)}\right]^T, \tag{10}$$

$$\boldsymbol{\alpha}^{(G)} = \left[1, \alpha, \alpha^2, \ldots, \alpha^{G-1}\right]^T. \tag{11}$$

The first element of the vector from (10) is $f_0^{(G)} = G$. Comparing (8) with (7) we can conclude that the LMS algorithm with multiple iterations per bit is equivalent with a LMS algorithm with a single iteration per bit but using a particular step size parameter given by

$$\mu^{(G)} = \mu F\left(\alpha\right). \tag{12}$$

To compute this parameter we need for the elements of the vector from (10). Let us consider the row vectors:

$$\underline{\mathbf{f}}^{(g)} = \left[\mathbf{f}^{(g)T}, \mathbf{0}_{1\times(G-g)}\right], \quad g = \overline{1,G}. \tag{13}$$

It can be noticed that $\underline{\mathbf{f}}^{(G)} = \mathbf{f}^{(G)T}$. Using the vectors from (13) we can obtain the matrix

$$\mathbf{F}^{(G)} = \begin{bmatrix} \underline{\mathbf{f}}^{(1)} \\ \underline{\mathbf{f}}^{(2)} \\ \vdots \\ \underline{\mathbf{f}}^{(G)} \end{bmatrix}. \tag{14}$$

Each row of this matrix corresponds to a specific iteration. The elements of this matrix can be computed using

$$\mathbf{F}^{(G)}\left(i,j\right) = \mathbf{F}^{(G)}\left(i-1,j-1\right) + \mathbf{F}^{(G)}\left(i-1,j\right), \quad i,j = \overline{2,G}. \tag{15}$$

It is obvious that $\mathbf{F}^{(G)}\left(j,1\right) = f_0^{(j)} = j$, with $j = \overline{1,G}$, and $\underline{\mathbf{f}}^{(1)} = \left[1, \mathbf{0}_{1\times(G-1)}\right]$. Once we have set the value of parameter G, this matrix can be computed a priori and we will use only the elements of its last row. For example, assuming that we choose $G = 10$, the matrix from (14) results

$$\mathbf{F}^{(10)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 10 & 10 & 5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 6 & 15 & 20 & 15 & 6 & 1 & 0 & 0 & 0 & 0 \\ 7 & 21 & 35 & 35 & 21 & 7 & 1 & 0 & 0 & 0 \\ 8 & 28 & 56 & 70 & 56 & 28 & 8 & 1 & 0 & 0 \\ 9 & 36 & 84 & 126 & 126 & 84 & 36 & 9 & 1 & 0 \\ 10 & 45 & 120 & 210 & 252 & 210 & 120 & 45 & 10 & 1 \end{bmatrix}.$$

The last row of this matrix contains the elements of $\mathbf{f}^{(10)}$ which will be a priori computed in this manner. Of course, there is a need for the elements of the vector from (11), which have to be computed for each input data bit. After that, the parameter from (9) has to be computed, which requires G multiplication operations and $(G - 1)$ addition operations. Finally, we update the filter coefficients according to (8). From the computational complexity point of view our approach requires almost the same number of operations as the multiple iterations per bit algorithm. Nevertheless, there is a significant time processing gain because our final result from (8) does not "wait" for the previous G iterations results, as in the multiple iterations process.

According to the theory of stability [4], the LMS adaptive algorithm is stable if the value of the step size parameter (which is a positive constant) is smaller than λ_{max}, which is the largest eigenvalue of the correlation matrix of the input signal (i.e., the input data bit). In the case of the multiple iterations LMS algorithm, the stability condition has to take into account the number of iteration per bit. In our approach this is very facile because we use a single iteration as in (8), with the step

size parameter given by (12). Consequently, the stability condition become

$$\mu^{(G)} < \frac{2}{\lambda_{\max}}. \tag{16}$$

Even if the multiple iterations algorithm is used, the previous condition can be used to determine the stability upper bound value of the step size parameter μ. According to (12) and (16)

$$\mu < \frac{2}{\lambda_{\max} F(\alpha)}. \tag{17}$$

An interesting situation appears if we use the Normalized LMS (NLMS) adaptive algorithm [4], which is another member of the stochastic gradient family. In this case, the algorithm step size parameter is computed as

$$\mu_{NLMS} = \frac{\bar{\mu}}{\mathbf{r}^T \mathbf{r}} = \frac{\bar{\mu}}{s}. \tag{18}$$

where $\bar{\mu}$ is a positive constant. In the case of the classical NLMS algorithm the previous constant has to be smaller than 2, in order to assure the stability of the algorithm [4]. Following the same procedure, it will result that

$$\mathbf{w}^{(G+1)} = \mathbf{w}_0 + \mu_{NLMS} \mathbf{r} e_0 F(\bar{\mu}), \tag{19}$$

where

$$F(\bar{\mu}) = \sum_{k=0}^{G-1} f_k^{(G)} (-\bar{\mu})^k = \mathbf{f}^{(G)^T} \boldsymbol{\beta}^{(G)}, \tag{20}$$

$$\boldsymbol{\beta}^{(G)} = \left[1, -\bar{\mu}, (-\bar{\mu})^2, \ldots, (-\bar{\mu})^{G-1}\right]^T. \tag{21}$$

Similarly, the NLMS algorithm with multiple iterations per bit is equivalent with a NLMS algorithm with a single iteration per bit but using a step size parameter given by

$$\mu_{NLMS}^{(G)} = \mu_{NLMS} F(\bar{\mu}) = \frac{\bar{\mu} F(\bar{\mu})}{\mathbf{r}^T \mathbf{r}} \tag{22}$$

The following condition has to be satisfied for the stability:

$$0 < \bar{\mu} F(\bar{\mu}) < 2 \tag{23}$$

Because it is more facile to compute the elements of the vector from (21), as compared with the elements of the vector from (11), the NLMS algorithm could be an attractive alternative to the LMS algorithm. Moreover, it is a more robust algorithm because it overcomes in some sense the gradient noise amplification problem associated with the LMS algorithm [4]. Nevertheless, a division operation is required for computing the step size parameter from (22), which could be a difficulty and a source of numerical errors especially in a fixed-point implementation.

IV. GAL ALGORITHM FOR MMSE RECEIVER

Expected advantages of the adaptive lattice filters over the conventional LMS transversal filters include faster convergence rates with spectrally deficient inputs, automatic determination of the system's order, stage-to-stage modularity for efficient hardware implementations, and better data tracking abilities.

The $(N-1)$-th-order multistage lattice predictor from Fig. 2 is specified by the recursive equations

$$\begin{aligned} e_p^f(l) &= e_{p-1}^f(l) + k_p^*(l) e_{p-1}^b(l-1) \\ e_p^b(l) &= e_{p-1}^b(l-1) + k_p(l) e_{p-1}^f(l) \end{aligned} \tag{24}$$

with p=1,…,N–1. We denoted by $e_p^f(l)$ the forward prediction error, by $e_p^b(l)$ the backward prediction error, and by $k_p(l)$ the reflection coefficient at the pth stage and chip-time l. The initial prediction errors are $e_0^f(l) = e_0^b(l) = r_{m,l}$.

The cost function used for the estimation of $k_p(l)$ is [4]

$$J_p(l) = \frac{1}{2} E\left\{\left|e_p^f(l)\right|^2 + \left|e_p^b(l)\right|^2\right\} \tag{25}$$

where E is the statistical expectation operator. Substituting (24) into (25), then differentiating the cost function $J_p(l)$ with respect to the complex-valued reflection coefficient $k_p(l)$ and imposing the gradient equal to zero, the optimum value of the reflection coefficient for which the cost function is minimum results

$$k_p^{opt} = -\frac{2E\left\{e_{p-1}^b(l-1) e_{p-1}^{f*}(l)\right\}}{E\left\{\left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2\right\}} \tag{26}$$

Assuming that the input signal is ergodic, the expectations can be substituted by time averages, resulting the Burg estimate for the reflection coefficient k_p^{opt} for stage p:

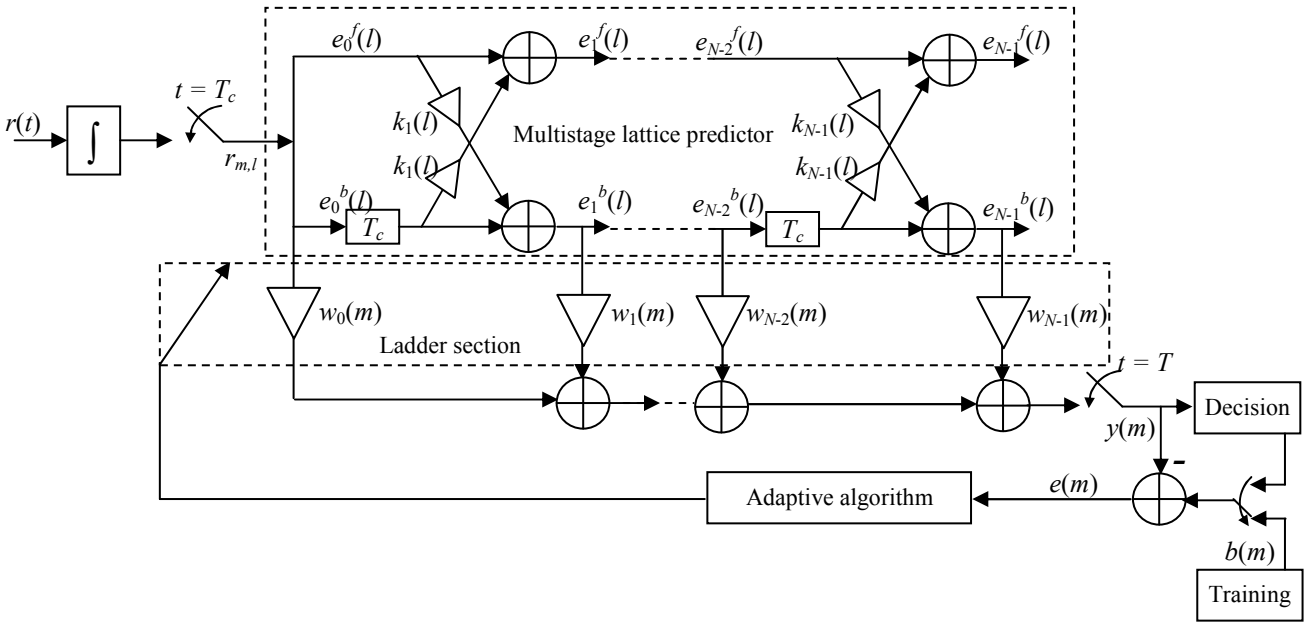


Figure 2. Lattice MMSE receiver scheme.

$$k_p(l) = -\frac{2\sum_{q=1}^{l} e_{p-1}^{b}(q-1)e_{p-1}^{f*}(q)}{\sum_{q=1}^{l}\left[\left|e_{p-1}^{f}(q)\right|^2 + \left|e_{p-1}^{b}(q-1)\right|^2\right]} \tag{27}$$

Let us denoted by $W_{p-1}(l)$ the total energy of both the forward and backward prediction errors at the input of the pth lattice stage. It is expressed as:

$$\begin{aligned} W_{p-1}(l) &= \sum_{q=1}^{l}\left[\left|e_{p-1}^{f}(q)\right|^2 + \left|e_{p-1}^{b}(q-1)\right|^2\right] = \\ &= W_{p-1}(l-1) + \left|e_{p-1}^{f}(l)\right|^2 + \left|e_{p-1}^{b}(l-1)\right|^2 \end{aligned} \tag{28}$$

It can be demonstrated [4] that the GAL algorithm updates the reflection coefficients using

$$k_p(l+1) = k_p(l) - \frac{\eta}{W_{p-1}(l)} \cdot \left[e_p^{f*}(l)e_{p-1}^{b}(l-1) + e_p^{b*}(l)e_{p-1}^{f}(l)\right] \tag{29}$$

where the constant η controls the convergence of the algorithm. For a well-behaved convergence of the GAL algorithm, it is recommended to set $\eta < 0.1$. In practice, a minor modification is made to the energy estimator from (28) by writing it in the form of a single-pole average of squared data:

$$W_{p-1}(l) = \beta W_{p-1}(l-1) + (1-\beta) \cdot \left[\left|e_{p-1}^{f}(l)\right|^2 + \left|e_{p-1}^{b}(l-1)\right|^2\right] \tag{30}$$

where $0 < \beta < 1$. The introduction of parameter β in (30) provides the GAL algorithm with a finite memory, which helps it to deal better with statistical variations when operating in a nonstationary environment. As it was reported in [8] and it was demonstrated in [14], the proper choice is $\beta = 1 - \eta$.

As we see in Fig. 2, the basic structure for the estimation of the user desired response $b(m)$, is based on a multistage lattice predictor that performs both forward and backward predictions, and an adaptive ladder section. We have an input column vector of the backward prediction errors

$$\mathbf{e}_N^b(m) = [e_0^b(m), e_1^b(m), \ldots, e_{N-1}^b(m)]^T \tag{31}$$

and a corresponding column vector $\mathbf{w}(m)$ containing the N coefficients of the ladder section of the adaptive filter. For the estimation of $\mathbf{w}(m)$, we may use a stochastic-gradient approach. The discrete output signal $y(m)$ is given by:

$$y(m) = \mathbf{w}^T(m)\mathbf{e}_N^b(m) \tag{32}$$

The receiver forms the error signal $e(m)$ and a new filter tap weight vector is estimated according to:

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \tilde{\mu} e(m) \mathbf{e}_N^b(m) \tag{33}$$

The parameter $\tilde{\mu}$ is the ladder structure adaptation step size, chosen to optimize both the convergence rate and the misadjustment of the algorithm.

Summarizing, we will use (24), (30), and (29) for the lattice predictor part of the scheme, together with (32) and (33) for the ladder section. As compared to its transversal counterpart based on the LMS algorithm, the lattice MMSE receiver implies an increased computational complexity due to the multistage lattice predictor. Nevertheless, due to the fact that the lattice predictor orthogonalizes the input signals, a faster convergence rate is expected.

It can be noticed that in the lattice predictor part of the classical GAL algorithm a division operation per stage is used, which significantly grows the computational complexity in a fixed-point implementation context. Due to cost considerations, equipment manufactures generally prefer the use of fixed-point Digital Signal Processors (DSPs) over floating-point ones in their products. In fixed point DSPs, every division operation requires a number of iterations equal to the word length (i.e., the number of representation bits), while the multiplication and addition operations can be performed in a single iteration. In the following, we will propose an approximate version of the GAL algorithm that replaces the division operation using three multiplication operations and one addition operation instead. This is much more convenient from the computational complexity point of view in a fixed-point DSP implementation.

We start from equation (30) and we may write:

$$\begin{aligned} &\frac{1}{W_{p-1}(l)} = \\ &= \frac{1}{\beta W_{p-1}(l-1) + (1-\beta)\left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2} = \\ &= \frac{1}{\beta W_{p-1}(l-1)} \cdot \frac{1}{1 + \frac{1-\beta}{\beta} \cdot \frac{\left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2}{W_{p-1}(l-1)}} \end{aligned} \tag{34}$$

Let us denote

$$T_{p-1}(l-1) = \frac{1}{W_{p-1}(l-1)} \tag{35}$$

and

$$c_{p-1}(l) = \left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2 \tag{36}$$

Using the previous notations we may rewrite equation (34) as follows:

$$T_{p-1}(l) = \frac{1}{\beta} T_{p-1}(l-1) \cdot \frac{1}{1 + \frac{1-\beta}{\beta} c_{p-1}(n) T_{p-1}(l-1)} \tag{37}$$

Supposing that the sum of the squared prediction errors at time l, i.e., $c_{p-1}(l)$, is much smaller than the sum of all squared prediction errors until that moment of time, $W_{p-1}(l-1)$, we may write that

$$\left|c_{p-1}(l) T_{p-1}(l-1)\right| << 1 \tag{38}$$

Taking into account that

$$\frac{1}{1+x} = \sum_{n=0}^{\infty} (-1)^n x^n \qquad \text{for} \qquad |x| < 1 \tag{39}$$

we may use the following approximate relation instead of equation (37):

$$T_{p-1}(l) \cong \frac{1}{\beta} T_{p-1}(l-1) \cdot \left(1 - \frac{1-\beta}{\beta} c_{p-1}(l) T_{p-1}(l-1)\right) \tag{40}$$

In order to prevent any unwanted situations that can affect the supposition made in equation (38) (e.g., impulse perturbation of the input signal) we may compute the following minimum value:

$$t_{p-1}(l) = \min\left(\lambda, \frac{1-\beta}{\beta} c_{p-1}(l) T_{p-1}(l-1)\right) \tag{41}$$

where $\lambda < 1$ is a positive constant, and than rewrite the equation (40) as follows:

$$T_{p-1}(l) \cong \frac{1}{\beta} T_{p-1}(l-1)(1 - t_{p-1}(l)) \tag{42}$$

Finally, the reflection coefficients are updated using

$$\begin{aligned} &k_p(l) = k_p(l-1) - \eta T_{p-1}(l) \cdot \\ &\cdot \left[e_p^{f*}(l) e_{p-1}^b(l-1) + e_p^{b*}(l) e_{p-1}^f(l)\right] \end{aligned} \tag{43}$$

The new step-size parameter of the division-free GAL algorithm is $\eta T_{p-1}(l)$ and it acts similar to $\eta / W_{p-1}(l)$ from the classical GAL algorithm. It can be noticed that the "unwanted" division operation from the classical GAL algorithm is replaced by three multiplication operations and one addition operation, which is more suitable in a fixed-point implementation context.

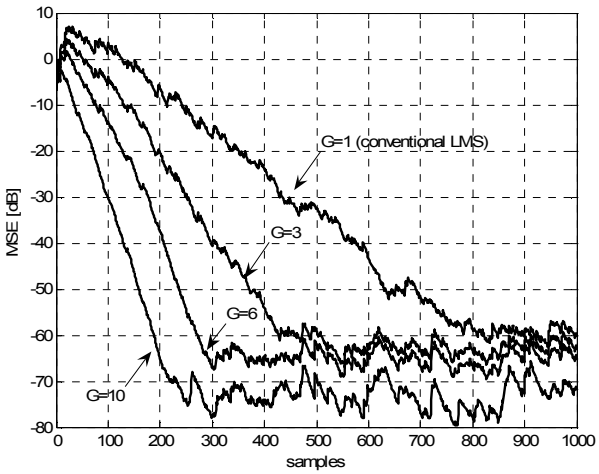


Figure 3. Convergence for the multiple iterations LMS algorithms.

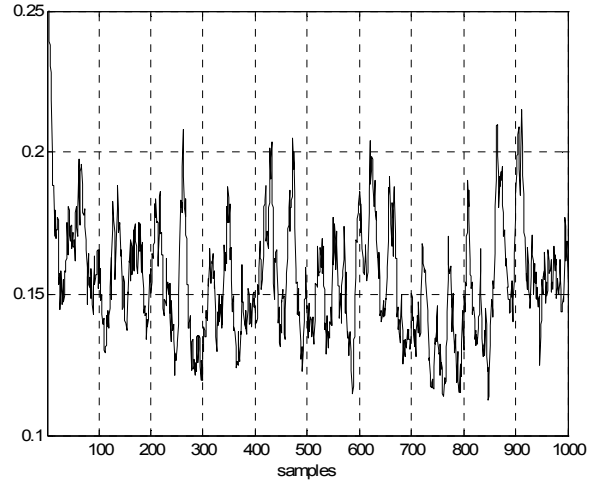


Figure 4. Evolution of the step size parameter $\mu^{(10)}$ given by (12).

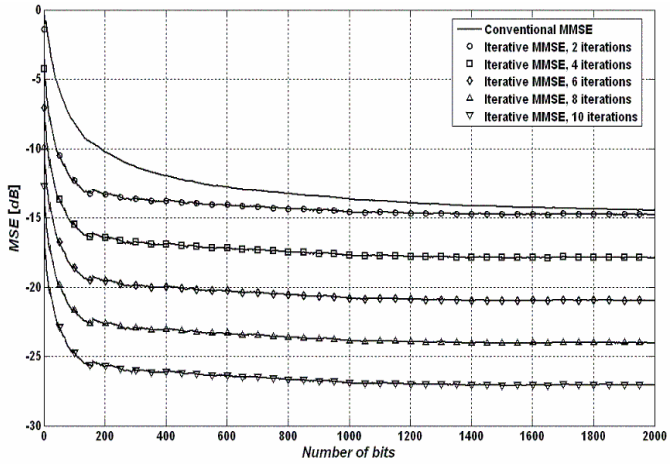


Figure 5. Convergence of the MMSE receiver based on the multiple iterations LMS algorithm. K = 8, N = 16, SNR = 15dB.

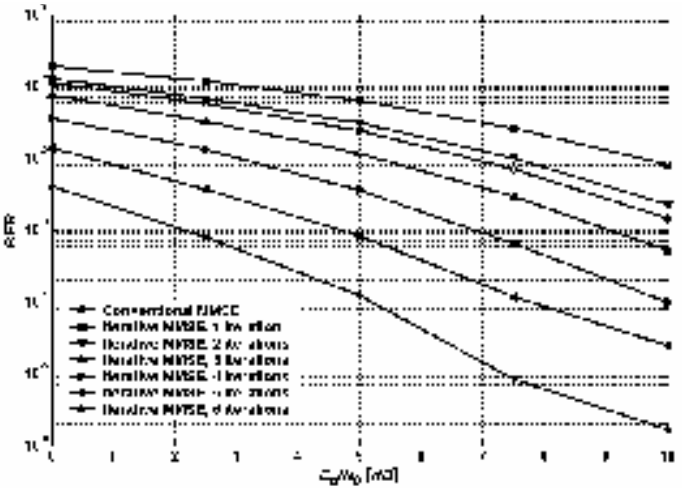


Figure 6. BER performance of the MMSE receiver based on the multiple iterations LMS algorithm. Other conditions as in Fig. 5.

V. SIMULATION RESULTS

In order to analyze the convergence of the multiple-iteration LMS algorithm, a first set of experimental tests is performed in a simple "system identification" configuration. In this class of applications, an adaptive filter is used to provide a linear model that represents the best "fit" to an unknown system. The adaptive filter and the unknown system are driven by the same input. The unknown system output supplies the desired response for the adaptive filter. These two signals are used to compute the estimation error, in order to adjust the filter coefficients. The input signal is a random sequence with an uniform distribution on the interval (–1;1). In this first set of experiments the adaptation process is performed in a sample-by-sample manner. The mean square error (MSE) is estimated by averaging over 100 independent trials. In Fig. 3 are depicted the convergence curves for the multiple iterations LMS algorithm and the proposed equivalent algorithm, for different values of G. As expected, the curves for these two algorithms are perfectly matched, due to the mathematical equivalency between them. It can be notice that the convergence rate of the algorithms increases with the value of G and the MSE decreasing. In Fig. 4 we plot the evolution of the step size parameter of the proposed algorithm, given by (12), for G =10.

A second set of simulations employs the asynchronous DS-CDMA system, using the transversal MMSE iterative receiver. A BPSK transmission in a training mode scenario was considered. The binary spreading sequences are pseudo-random. The system simulation parameters are N = 16 and K = 8. The signal-to-noise ratio (SNR) is 15 dB. The LMS adaptive algorithm is iterated several times for each data bit using the proposed equivalent relation from (8). The adaptive process works now in a block-by-block manner. The MSE is also estimated by averaging over 100 independent trials. The results are presented in Fig. 5. It can be noticed that the MSE is decreased every new iteration.

Next, we investigated the steady-state BER (Bit Error Rate) performance as a function of the energy per bit to noise power

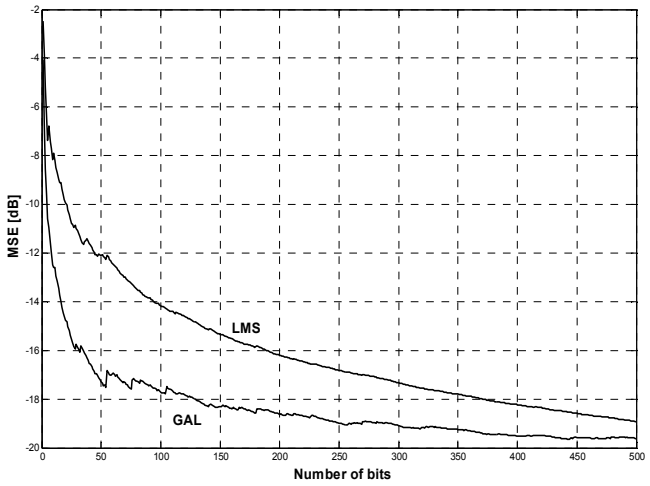


Figure 7. Convergence of the MMSE receivers. $K = 32$, $N = 64$, SNR = 15dB.

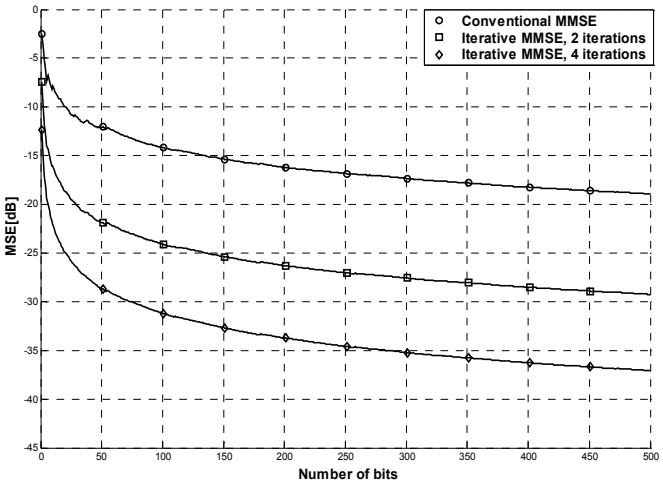


Figure 9. Convergence of the iterative LMS receiver. Other conditions as in Fig. 7.

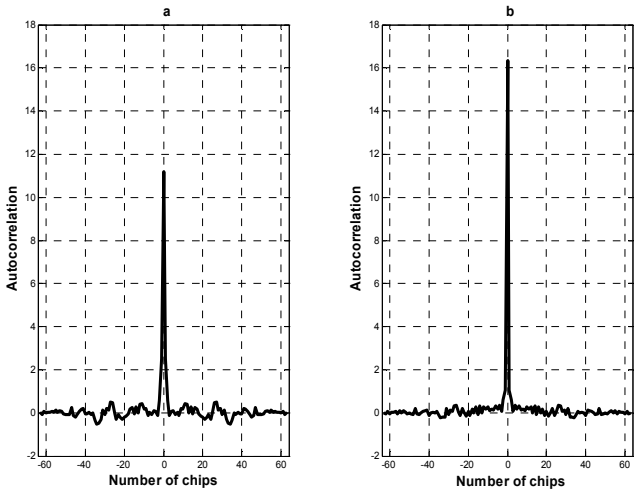


Figure 8. Autocorrelation functions for (a) $\mathbf{r}(m)$ - LMS input, (b) $\mathbf{e}^b_N(m)$ - GAL ladder section input. Other conditions as in Fig. 7.

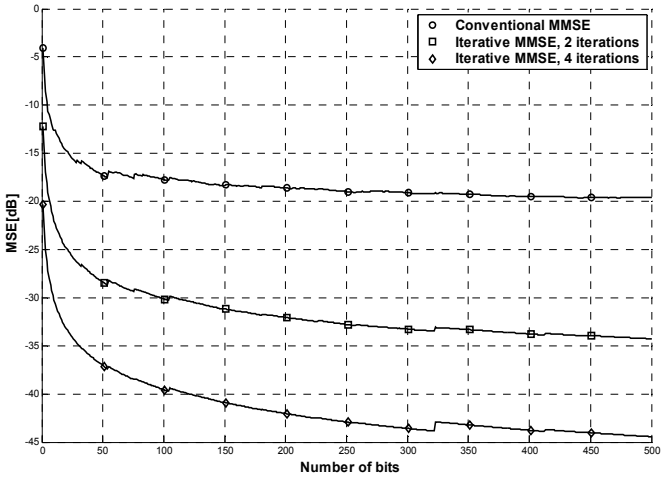


Figure 10. Convergence of the iterative GAL receiver. Other conditions as in Fig. 7.

spectral density ratio (E_b/N_0) of the iterative MMSE receiver considered above. These results are shown in Fig. 6, where we compared the performance of the iterative MMSE receiver using 1 to 6 additional iterations (i.e., G = 2 to 7) with the conventional adaptive LMS receiver, using one iteration per bit (i.e., G =1). The simulation results were obtained using 2000 bits training period for each value of E_b/N_0, in order to assure that the steady-state is reached. It is very important to note that under the same conditions the BER is improved every new iteration. Nevertheless, one should make a compromise between the computational complexity and the BER performances.

For further increasing the convergence rate, the lattice MMSE receiver based on the division-free GAL algorithm is tested as compared with its transversal counterpart based on the LMS algorithm. The simulation parameters were fixed as follows. The processing gain is $N = 64$, the number of users is $K = 32$, and SNR = 15 dB. The MSE is estimated by averaging over 100 independent trials. First, the conventional MMSE receivers are compared (Fig. 7). The superior convergence rate achieved by the GAL algorithm as compared to the transversal LMS algorithm is obvious. This faster convergence rate of the GAL algorithm over the transversal LMS algorithm can be explained by the fact that the lattice predictor orthogonalizes the input signals. Hence, the gradient adaptation algorithm using this structure is less dependent on the eigenvalues spread of the input signal. In order to support these remarks, in Fig. 8 the mean autocorrelation function is depicted for both inputs, i.e. $\mathbf{r}(m)$ (used as the direct input for the transversal LMS receiver) and the sequence of backward prediction errors $\mathbf{e}^b_N(m)$ (the input of the ladder section of the GAL receiver). It can be noticed that the input of the GAL ladder section has a higher variance as compared to the LMS input sequence. We should note that the faster convergence rate offered by the iterative MMSE receivers over the conventional receivers is mainly due to the interference rejection capability of the

multiple iterations procedure within the adaptive algorithm. This improvement leads to a reduced level of MAI, which translates into a convergence rate close to the single user case.

Finally, a last set of simulations is performed in order to compare the multiple iterations MMSE receivers based on both LMS and GAL algorithms. Both algorithms are iterated for G = 4 within each data bit. The results are presented in Figs. 9 and 10. In both cases it can be noticed that MSE is decreased every new iteration. Also, the division-free GAL algorithm outperforms the LMS algorithm in terms of convergence rate.

VI. CONCLUSION AND PERSPECTIVES

In this paper we first propose a relation for updating the LMS adaptive filter coefficients, which is equivalent with a multiple iterations LMS algorithm used in MMSE receivers for DS-CDMA communications systems. It was demonstrated that the multiple iterations process is equivalent with an unique iteration with a particular step size of the algorithm. Our proposed solution requires almost the same number of operations as the multiple iterations per bit algorithm but a significant time processing gain is achieved. The stability condition for the proposed algorithm was also derived. The simulation results proved the equivalency between the classical multiple iteration LMS algorithm and our proposed approach.

The MMSE iterative receiver considered in this paper was shown to improve the asynchronous DS-CDMA system performances. Thus, the MSE is decreased every new iteration by reducing MAI. This decrease offers a faster training mode for the receiver. A very important result is that the BER is considerably improved every new iteration.

The lattice MMSE receiver based on the division-free GAL algorithm improves the convergence rate as compared to the transversal LMS receiver. The lattice predictor orthogonalizes the input signals, so that the GAL algorithm is less dependent on the eigenvalues spread of the input signal and it converges faster than their transversal counterpart, the LMS algorithm. As a practical consequence, the lattice receiver will require a shorter training sequence as compared to the transversal one. Also, the GAL iterative receiver was shown to improve the asynchronous DS-CDMA system performances. The MSE decrease offers a faster training mode for the receiver. Hence, the designing procedure may consider two aspects, i.e., to shorten the training sequence for maintaining the same MAI in the system or to strongly reduce the MAI by keeping the same length of the training sequence. An analytical estimation of BER for the GAL receiver will be considered in perspective.

REFERENCES

[1] S. Glisic and B. Vucetic, *CDMA Systems for Wireless Communications*, Artech House, 1997.

[2] S. Miller, "An adaptive direct-sequence code-division multiple-access receiver for multi-user interference rejection," *IEEE Trans. Communications*, vol. 43, pp. 1746-1755, Apr. 1995.

[3] P. Rapajic and B. Vucetic, "Adaptive receiver structures for asynchronous CDMA systems," *IEEE J. Select. Areas Communic.*, vol. 12, no. 4, pp. 685-697, May 1994.

[4] A. H. Sayed, *Adaptive Filters*, Wiley, 2008.

[5] F. Takawira, "Adaptive lattice filters for narrowband interference rejection in DS spread spectrum systems," *Proc. IEEE South African Symp. Communications and Signal Processing*, 1994.

[6] J. Wang and V. Prahatheesan, "Adaptive lattice filters for CDMA overlay," *IEEE Trans. Commun.*, vol. 48, no. 5, May 2000, pp. 820-828.

[7] C. Paleologu, C. Vlădeanu, and A.A. Enescu, "Lattice MMSE single user receiver for asyncronous DS-CDMA systems," *Proc. of IEEE Int. Symp. ETc 2006*, pp. 97-102.

[8] V. J. Mathews and Z. Xie, "Fixed-point error analysis of stochastic gradient adaptive lattice filters," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 38, no. 1, January 1990, pp. 70-79.

[9] R. C. North, J. R. Zeidler, W. H. Ku, and T. R. Albert, "A floating-point arithmetic error analysis of direct and indirect coefficient updating techniques for adaptive lattice filters," *IEEE Trans. on Signal Processing*, vol. 41, no. 5, May 1993, pp. 1809-1823.

[10] W. Hamouda and P. McLane, "Multiuser interference cancellation aided adaptation of a MMSE receiver for direct-sequence code-division multiple-access systems," *Proc. IEEE Communications Theory Mini-Conf. (GLOBECOM)*, 2001.

[11] W. Hamouda and P. McLane, "A fast adaptive algorithm for MMSE receivers in DS-CDMA systems," *IEEE Sign Proc. Letters*, vol. 11, no. 2, pp. 86-89, Feb. 2004.

[12] C. Paleologu and C. Vlădeanu, "On the behavior of LMS adaptive algorithm in MMSE receivers for DS-CDMA systems," *Proc. of ICCGI 2007*, pp.12-17.

[13] C. Paleologu, S. Ciochină, A. A. Enescu, and C. Vlădeanu, "Gradient adaptive lattice algorithm suitable for fixed point implementation," *Proc. of ICDT 2008*, pp. 41-46.

[14] C. Paleologu, S. Ciochină, and A.A. Enescu, "Modified GAL algorithm suitable for DSP implementation," *Proc. of IEEE Int. Symp. ETc 2002*, vol. 1, pp. 2-7.

Adaptive Congestion Detection and Control at the Application Level for VoIP

Teck-Kuen Chua and David C. Pheanis
Computer Science and Engineering
Ira A. Fulton School of Engineering
Arizona State University
Tempe, Arizona 85287-8809
Email: TeckKuen.Chua@gmail.com or David.Pheanis@ASU.edu

Abstract

For decades, researchers have worked extensively in the area of congestion control for packet-switched networks. Many proposed solutions take advantage of the congestion-control mechanism in Transmission Control Protocol *(TCP), and these approaches work well for networks that have heavy TCP traffic. However, these approaches are not universally effective — they fail completely for protocols that do not implement congestion-control mechanisms. In particular, these approaches do not work with the* User Datagram Protocol *(UDP). Real-time media-streaming technologies such as* Voice over Internet Protocol *(VoIP) and video conferencing use UDP and therefore do not respond well to existing congestion-avoidance techniques. We propose a new, adaptive, responsive, end-to-end technique to implement application-level congestion detection and control for real-time applications such as VoIP. Unlike existing methods, which rely on packet loss as a signal to reduce the transmission rate, our solution proactively reacts to network congestion to* prevent *packet loss, thus improving the QoS of applications that employ our algorithm.*

Keywords: VoIP, QoS, congestion detection, congestion avoidance, adaptive transmission control, real-time media.

1. Introduction

In a packet-switched network, a router connects multiple ingress streams to various egress ports. When a heavy burst of network traffic occurs, congestion builds up at the routers, which form the bottlenecks of the network. When congestion in a router becomes severe enough, the incoming packets consume all of the buffer resources in the router and leave no room for additional inbound packets, thus resulting in lost packets.

The output rate at an egress queue in a router can be only as fast as the serialization rate of the hardware. When an egress queue receives packets from multiple ingress ports at a combined rate that is higher than the serialization rate of the egress port, the egress queue grows, eventually causing congestion. When the egress queue becomes full, the router has to discard all additional packets destined for that egress queue. This dropping of packets is an unwelcome condition known as *tail drop*.

Dropped packets are undesirable at any time, of course, but tail drop is especially undesirable because it leads to an adverse effect called *global synchronization*. Global synchronization occurs when tail drop causes all of the transmitting devices to receive congestion signals at the same time. The transmitting devices consequently reduce their transmission rates in unison, and the link utilization quickly falls well below the optimal level due to the sudden, simultaneous reduction in transmission rates from all of the senders. When congestion eases, the transmitting devices increase their transmission rates all at once, leading to another episode of severe congestion in the network.

Researchers have done considerable work in this area and have proposed numerous solutions to manage and avoid congestion in a packet-switched network. Proposed congestion-avoidance methods include *Random Early Detection* (RED) [1] and its variants [2] [3] [4] [5], and proposed approaches also include BLUE [6], *Stochastic Fair BLUE* (SFB) [7], *Generalized Random Early Evasion Network* (GREEN) [8], and *Explicit Congestion Notification* (ECN) [9]. Each of these suggested techniques exploits the transmission-control mechanism in *Transmission Control Protocol* (TCP). Since real-time IP applications generally use the *User Datagram Protocol* (UDP) as the transport protocol, however, these proposed congestion-control approaches are almost entirely ineffective at curbing real-time media traffic.

We propose a new application-level adaptive congestion-detection and congestion-control mechanism for avoiding

congestion with real-time IP applications such as *Voice over Internet Protocol* (VoIP)[1]. We start by explaining how existing congestion-avoidance algorithms work with TCP, and we discuss the problems that existing approaches have with UDP and the intrinsic challenges of congestion control with real-time applications. Then we present our new application-level adaptive congestion-detection and congestion-control solution for real-time IP applications, and we provide measurements that demonstrate the effectiveness of our system.

2. Existing approaches

Congestion has been a serious problem in packet-switched networks since packet switching first came into existence. The importance of the congestion problem is evident from the extensive amount of work that researchers have done to provide ways of avoiding or at least managing congestion. In this section we examine a few well-known congestion-avoidance techniques and show how these methods alleviate congestion in packet-switched networks, so this section provides a basis for understanding the new real-time congestion-control technique that we present later.

2.1. Random Early Detection (RED)

Random Early Detection (RED) is an active queue-management (AQM) technique with the aim of achieving low average delay and high throughput [1]. RED uses the average queue size as an early indication of congestion and signals the transmitting devices to reduce their transmission rates temporarily before the congestion actually occurs. When the average queue size exceeds a predetermined minimum threshold, RED starts dropping randomly selected packets. The probability of dropping packets increases either linearly or exponentially as the average queue size grows beyond the minimum threshold. When the average queue size passes a predetermined maximum threshold, RED starts dropping *all* incoming packets. RED uses the average queue size over some period of time instead of using the instantaneous queue size, so the technique can absorb spikes or short bursts of network traffic without overreacting.

When a TCP host detects packet loss, the host temporarily slows down its transmission rate. TCP increases its transmission rate quickly when all packets reach the destination, an indication that the period of congestion has passed. RED randomly selects packets to drop, thereby distributing the packet loss among assorted hosts at various times. As a result, the hosts reduce their transmission rates at different times and avoid global synchronization.

Weighted RED (WRED) incorporates the IP precedence feature into the RED algorithm. WRED gives preferential handling to packets that have higher priority. When congestion is building, WRED randomly selects packets with lower priority as the first packets to discard. This scheme creates differentiated QoS characteristics for different classes of service.

Some researchers have proposed dynamic RED implementations that adapt to ever-changing network conditions. For example, *adaptive RED* [2], *Dynamic Weighted Random Early Drop* (DWRED) [3], and other similar approaches are adaptive RED variants that are designed to address the sensitivity weaknesses of RED. Both the throughput and the average queue size of RED are very sensitive to the traffic load and RED parameters [10][11], so RED produces unpredictable results in volatile network environments.

Flow-based RED (FRED) [4] includes a mechanism to enforce fairness in resource utilization for each active flow of traffic. This technique uses information such as destination addresses, source addresses, and ports to classify traffic into different flows. The flow-based algorithm maintains state information for every active flow, and the algorithm uses the flow information to ensure that each active flow gets a fair portion of the buffer resources. Flows that monopolize resources receive heavier penalties when packet dropping becomes necessary.

Stabilized RED (SRED) [5] is another flow-based RED approach where the algorithm provides a method to estimate the number of active flows and to identify misbehaving flows without keeping per-flow state information. SRED controls buffer usage by tuning the drop probabilities based on the estimated number of active flows.

2.2. Stochastic Fair BLUE (SFB)

BLUE is an AQM approach that uses packet-loss and link-utilization information instead of average queue length in the congestion-avoidance algorithm [6]. BLUE uses a single probability to drop or mark packets. If the link is idle or the queue becomes empty, BLUE decreases the drop/mark probability. On the other hand, if the queue consistently loses packets due to buffer overflow, BLUE increases the drop/mark probability. As a result of the increased probability, we drop or mark more packets and therefore send out congestion notifications at a higher rate. This adaptive procedure allows BLUE to learn the correct rate for sending congestion signals to the transmitting hosts.

Stochastic Fair BLUE (SFB) uses the BLUE algorithm to protect TCP flows against flows — such as UDP flows — that do not respond to congestion notifications [7]. SFB maintains a small amount of flow-related state information in order to enforce fairness among all the flows. The SFB algorithm quickly drives the drop/mark probability to a very high value, perhaps even one, for an unresponsive flow. In contrast, TCP flows usually maintain low drop/mark prob-

[1] Patent pending

abilities because TCP flows reduce their transmission rates in response to congestion notifications. When the SFB technique identifies an unresponsive flow by observing a high drop/mark probability, SFB applies a bandwidth-limiting policy on that particular flow. The rate-limit policy enforces an allowable amount of data that the flow can enqueue into the buffer, and SFB therefore drops more packets of the unresponsive flows. In simplified terms, the SFB algorithm identifies unresponsive flows and subjects them to rate limiting while allowing responsive TCP flows to perform normally with low drop/mark probabilities.

2.3. GREEN

Generalized Random Early Evasion Network (GREEN) is a proactive queue-management (PQM) method that applies a mathematical model of the steady-state behavior of a TCP connection to drop or mark packets proactively. GREEN attempts to maintain low packet loss and high link utilization while reducing latency and delay jitter. Based on the mathematical model, GREEN is able to give each flow its fair share of bandwidth at the router. The router can use GREEN to identify and police flows that do not respond to congestion notification.

Using the mathematical model that Mathis et al. [12] recommend, Feng et al. [8] derive a mathematical model for a *drop* probability that allows a fair share of bandwidth for every flow. Equation 1 shows that Feng's fair-share drop probability depends on the number of active flows, N, and the round-trip time, RTT. A GREEN router sends out congestion notifications more aggressively when there are more active flows (larger N) or when the round-trip time is shorter (smaller RTT). In the equation, MSS is the maximum segment size, L is the outgoing link throughput of a router, and c is a constant that depends on the acknowledgment strategy (i.e., *delay* or *every packet*).

$$p = \left(\frac{N \times MSS \times c}{L \times RTT} \right)^2 \quad (1)$$

2.4. Explicit Congestion Notification (ECN)

The Internet Engineering Task Force, IETF, has proposed *Explicit Congestion Notification* (ECN) as an alternative to using packet loss for signaling congestion [9]. Instead of dropping packets as congestion increases, routers using this congestion-avoidance algorithm set the congestion-indicator bit in an IP packet to signal congestion. When a receiving device receives a packet with the congestion-indicator bit set, the receiving device uses the transport-level acknowledge message to communicate the congestion indication to the transmitting device. Upon receiving the explicit congestion notification, the sending device decreases its transmission rate temporarily until the traffic condition of the network improves.

Using IP ECN instead of the traditional packet-loss approach can obviously reduce packet loss and thereby improve performance. However, the ECN technique requires explicit support from both communicating devices to be successful, and both devices have to agree to use this scheme. In addition, all of the routers in the entire network must support this method for ECN to be effective. Outdated routers that do not support ECN might drop packets with the ECN bit set, thus causing the IP ECN approach to fail.

2.5. (Pre-) Congestion Notification (PCN)

The Congestion and Precongestion Notification Working Group of IETF has developed an Internet draft for a Pre-Congestion Notification (PCN) architecture [13]. PCN is an architecture for flow admission and/or termination based on (pre-) congestion information that nodes in a Diffserv domain provide. The aim of PCN is to protect the QoS of inelastic flows within the Diffserv domain. The PCN approach gives an "early warning" of potential congestion in the PCN domain before there is any significant congestion.

If the flow rate through a PCN-enabled interior node exceeds the PCN threshold rate, the node marks all packets with PCN threshold-rate markings. If the flow rate exceeds the PCN excess rate, the node marks some packets with PCN excess-rate markings while marking all remaining packets with PCN threshold-rate markings. These (pre-) congestion markings propagate to PCN egress nodes, and the PCN boundary nodes use this (pre-) congestion information to make decisions on flow admission and/or termination. Fig. 1 shows how the PCN admission and termination controls operate in a PCN domain with three encoding states as the rate of PCN traffic increases.

Note that PCN applies to a Diffserv domain with PCN-enabled nodes, so PCN is not a general solution for all environments.

3. Problems with existing approaches

Real-time IP applications, such as VoIP and IP video conferencing, use the *Real-time Transport Protocol* (RTP) to transport real-time media packets, and RTP runs on top of UDP. Unlike TCP, UDP does not support *acknowledge* (ACK) messages in the protocol, nor does UDP implement any transmission-control mechanism to allow the network to alter the transmission rate of a flow.

Lacking ACK messages, an application that uses UDP as the transport protocol has no way to determine when a router drops any of the packets that the application sends. Packet loss is a reliable indication of congestion, but UDP transmitters, being unable to detect packet loss, cannot react to congestion in the network. Even if a transmitter could

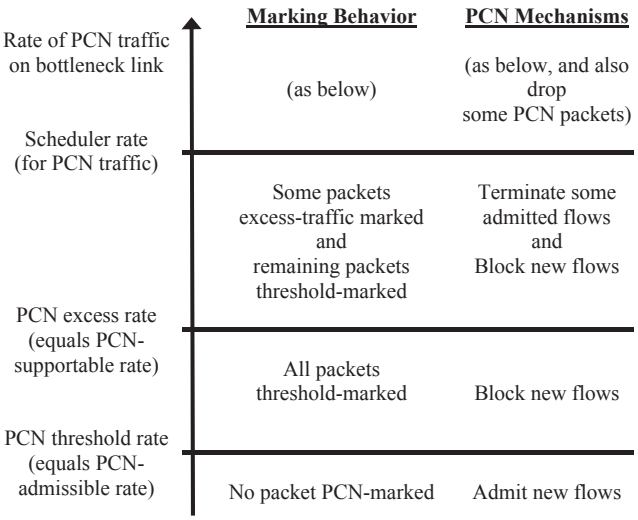


Figure 1. PCN technique

somehow detect the occurrence of packet loss, UDP does not have a mechanism to control its transmission rate. Consequently, UDP applications do not respond well to any of the existing congestion-control algorithms.

Even congestion-avoidance techniques that try to enforce fairness, as flow-based RED and SFB do, are not fully effective against UDP streams, which do not respond to congestion-control mechanisms. Since UDP flows are not responsive to congestion-control signals, these flows simply monopolize the resources of the router in an environment of congestion. Consequently, these streams receive harsh punishment in the form of many lost packets, and they therefore suffer greatly reduced *Quality of Service* (QoS).

TCP-Friendly Rate Adaptation Based on Loss (TRABOL) is an application-level congestion-control algorithm designed specifically for UDP-based applications [14]. TRABOL employs a technique that is similar to the approach of TCP, but TRABOL implements congestion control at the application level while TCP uses the protocol level. Like TCP, TRABOL relies on lost packets to adjust the sender's transmission rate. Unlike TCP, however, TRABOL uses the loss rate computed over a period of time at the receiver side as feedback to tell the sender how to adjust the transmission rate. If the period for calculating the loss rate in TRABOL is too large, the sender could react to the congestion too late or simply adjust the transmission rate incorrectly. If the period for calculating the loss rate in TRABOL is too small, on the other hand, the system could overreact to minor disturbances that do not really indicate congestion. In this case, the flow rate would oscillate needlessly. Furthermore, TRABOL does not address the real-time criterion of real-time UDP-based applications such as VoIP.

The inherent characteristics of real-time applications such as VoIP present additional challenges to the implementation of transmission control. Real-time IP applications typically produce output data at a constant rate, and we need to transport the continuous output stream to the receiving endpoint with minimal delay to maintain the usefulness of the data and a high QoS. Delivering only part of the output stream to reduce the output data rate inevitably degrades QoS. Holding back the real-time data transmission in order to wait for the end of a congestion period increases delay variation (i.e., *jitter*) and lengthens the end-to-end delay, further impairing QoS.

4. Our congestion-control mechanism

We propose a solution that allows real-time IP applications to implement adaptive congestion control in the face of congestion while meeting all of the intrinsic challenges of real-time media systems. Our solution consists of two independent components, congestion detection/notification and adaptive transmission control. Unlike existing approaches that implement congestion control at the protocol level, our technique implements the solution at the application level. As a result, we do not need to change existing protocols or the existing infrastructure to put our approach into practice.

4.1. Congestion detection/notification

Network congestion normally occurs at the routers, and detecting congestion at the router bottleneck is probably the detection approach that provides the most reliability and accuracy. Therefore, the router is the best device for performing congestion detection. Since the router is the first network device that can observe congestion, the router can deliver a notification of congestion sooner than any other device in the network. However, the innumerable routers of the Internet are beyond our control, so deploying new procedures for congestion detection and notification in all the routers of the Internet is impractical.

Our solution employs congestion detection at the receiving endpoint. We merely require both communicating endpoints to agree on the scheme of congestion detection and notification instead of requiring many devices beyond our control to cooperate with our algorithm.

When congestion occurs at the router, the average queue size in the router grows. High queue occupancy at the router increases the transmission delay of the packets since a packet takes more time to work its way through a longer router queue, and packets arrive at the destination later than anticipated. Real-time IP applications commonly transmit data packets at constant intervals, so data packets arrive at the receiving endpoint with consistent periods and minimal variation if the network is idle. Therefore, an increase in the time between the arrivals of consecutive packets at the destination is a good indication of congestion [15].

If congestion is severe enough, a router that implements a congestion-avoidance algorithm, such as RED or its variants, starts to drop packets to curb the congestion. Consequently, data packets disappear from the network, and the receiving endpoint can use packet loss as a clear indication of congestion. If the routers on the transmission path implement IP ECN, the receiving endpoint can recognize the set ECN flag as a sign of congestion.

Statistics that we have collected over a period of several months on the Internet and on a corporate local-area network (LAN) show that even a lightly loaded router experiences sudden microbursts of traffic. These abrupt and brief periods of congestion are severe enough to cause the router to drop packets, but the times between the arrivals of previous consecutive packets do not always show any early sign of congestion. This kind of unforeseen and acute congestion is different from the congestion that builds over a longer period of time, and our congestion-detection algorithm must be able to detect both types of congestion effectively.

The receiving endpoint can recognize slowly building congestion in the network by implementing an algorithm to detect inter-packet delays that are longer than usual. In order to detect variations in inter-packet delays, of course, the receiver must know the arrival rate of the packets. The transmitting party or parties can reveal the transmission-rate information during the negotiation process at the start of a session, or the receiver can quickly and easily calculate the arrival rate of packets after the session starts.

When congestion builds up gradually, the detection algorithm easily detects the pattern of growing inter-packet arrival times before the congestion can cause any noticeable harm. When the filtered arrival time between consecutive packets exceeds a predetermined threshold, the receiver declares a state of congestion.

Unfortunately, there may be no warning sign of growing inter-packet arrival times before a microburst suddenly causes a lost packet, and the inter-packet arrival time would not reveal this problem until the arrival of the next packet after the lost packet(s). Therefore, the detection algorithm employs a time-out procedure to detect a delayed or missing packet immediately. The time-out mechanism declares a congestion condition when a packet has not arrived at the receiver by some predetermined time after the estimated packet-arrival time but before the estimated arrival time of the subsequent packet. The time-out procedure detects a congestion condition well ahead of the transmission time of the subsequent packet, so the transmitter can delay transmission briefly to avoid losing more packets. With the inclusion of the time-out feature, the algorithm detects sudden microbursts as well as slowly growing episodes of congestion.

Additionally, the system uses packet sequence numbers to detect out-of-order packets and considers an out-of-order packet to be an indication of a lost packet even though the "lost" packet may arrive later or may have previously arrived out of order. When packets are out of order, we certainly have congestion.

Upon detecting congestion, the receiver sends the transmitter a congestion notification containing an estimate of the severity of the congestion. This notification tells the transmitter that congestion is present and to what degree congestion is present. The receiver can piggyback the notification message with the next data packet that the receiver transmits to the sender, thereby avoiding the undesirable effect of adding notification packets to a network that is already congested. Both ends of a VoIP system continually send packets to each other at brief intervals, typically twenty milliseconds, so timely notification without any added packets is entirely feasible.

Our extensive study of the characteristics of packet-switched network traffic shows that network congestion is direction dependent. In other words, a congestion condition on the path from A to B does not necessarily imply a similar congestion condition on the path from B to A. In fact, we often observe relatively idle traffic on the opposite direction of a congested network path. Therefore, sending a congestion notification from the receiver to the transmitter typically does not elevate the severity of the congestion.

Nevertheless, the network may drop the packet containing the congestion notification, so the receiver should send multiple congestion notifications. However, each of the notification packets for the same occurrence of congestion must contain the same unique identifier so the sender reacts only once to the first notice it receives and ignores the rest. The receiver can stop transmitting the notification messages when it observes a lower transmission rate from the sender, or the receiver can stop transmitting the notification messages after some duration, especially when the congestion subsides.

4.2. Our implementation of congestion detection

Our congestion-detection algorithm measures and evaluates the inter-packet arriving intervals, so the algorithm not only detects congestion in the network but also estimates the severity of the congestion. Additionally, our algorithm anticipates future congestion that is likely to occur soon after an episode that is part of a longer period of network congestion. Our detection procedure computes the absolute value of the difference between the inter-packet arriving interval and the original inter-packet transmission interval. Then we pass the resultant value through a simple first-order infinite impulse response (IIR) filter to obtain the severity of the congestion in the network.

Equation 2 shows the first-order IIR filter that our algorithm uses. Y_n is the current output value of the filter, and it is the estimate of the severity of the congestion in the net-

```
CurrentTime = GetCurrentTime();
Xn = ABS((CurrentTime -
          PrevPacketArrTime) -
         PacketTransmissionInterval);
if (NOT Timeout)
  { PrevPacketArrTime = CurrentTime; }
if (LostPacket OR Timeout)
  { Xn = MIN(Xn, MAX_Xn_LIMIT);
    Xn = MAX(Xn, Yprevious); }
if (Xn >= Yprevious)
  { C1 = 0.9;
    C2 = 0.1; }
else
  { C1 = 0.03;
    C2 = 0.97; }
Yn = (C1 * Xn) + (C2 * Yprevious);
if (Yn >= CONGESTION_THRESHOLD)
  { DeclareCongestion(Yn); }
Yprevious = Yn;
```

Figure 2. Pseudo code of congestion-detection algorithm

work. Y_{n-1} is the previous output value of the filter, so it provides feedback. X_n is the current input sample to the filter, and it is the absolute value of the difference between the current inter-packet arriving interval and the inter-packet transmission interval. C_1 and C_2 are coefficients of the IIR filter.

$$Y_n = (C_1 \times X_n) + (C_2 \times Y_{n-1}) \tag{2}$$

Fig. 2 shows an excerpt from the pseudo code for our congestion-detection algorithm [16]. Our implementation uses an IIR filter with fast-rise and slow-decay characteristics. The fast-rise characteristic of the filter allows the measurement of congestion severity to increase quickly when congestion builds up. The slow-decay characteristic of the filter allows the congestion-severity value to fall gradually after the network congestion subsides, thus anticipating the likely prospect that network congestion might persist or that additional network congestion might occur shortly after the current episode of congestion. In comparison to an episode of moderate congestion, a period of severe congestion raises the measurement of congestion severity to a higher value, and a higher congestion-severity value requires a longer time to decay to a level below the congestion threshold. We can use different C_1 and C_2 coefficients to adjust the rise rate and decay rate of the IIR filter if necessary. A slower decay rate allows the system to anticipate congestion that might occur further into the future.

Fig. 3 illustrates inter-packet arriving intervals that are typical for data that we captured from the Internet during highly congested periods. The transmitting endpoint was transmitting one VoIP packet every twenty milliseconds in this experiment, and the receiving endpoint observed erratic inter-packet arriving intervals ranging from zero milliseconds (i.e., two or more packets arriving at about the same time) to ninety milliseconds or more during this period of severe congestion. The receiving endpoint even observed lost packets and, in some cases, out-of-order packets.

Fig. 4 illustrates the output from our congestion-detection mechanism for the inter-packet arriving intervals of Fig. 3. As we can see in Fig. 3, the receiving endpoint repeatedly observed extreme delays in the arriving packets. These extreme delays produce significant congestion-severity values that remain above the congestion threshold throughout the slow decay of the IIR filter. Our congestion-detection algorithm therefore reports that congestion continues throughout the entire episode of network congestion.

Fig. 5 shows inter-packet arriving intervals that we captured from a corporate LAN. As with the previous experiment, the transmitting endpoint transmits a VoIP packet every twenty milliseconds. During this period, the receiving endpoint observed three episodes of mild to serious congestion. Using the data in Fig. 5, our congestion-detection algorithm generates the output that appears in Fig. 6.

4.3. Adaptive transmission control

The real-time application at the transmitting endpoint implements the adaptive transmission-control mechanism. Upon receiving a congestion notice from the endpoint at the receiving end of the transmission, the transmitting end-

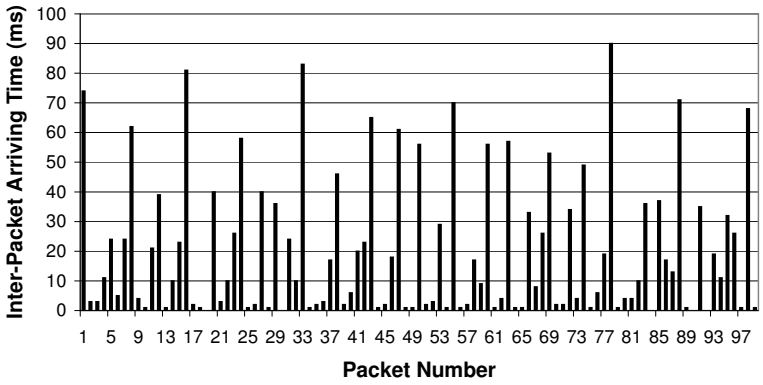


Figure 3. Inter-packet arriving interval (Internet)

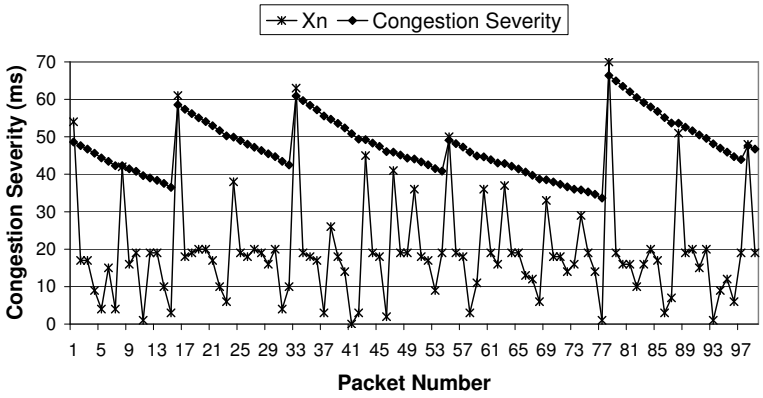


Figure 4. Estimate of congestion severity (Internet)

point lowers its transmission rate to reduce bandwidth consumption. This reduction of the transmission rate is not as simple for a real-time application such as VoIP as it is for a TCP application, though, because the transmitting endpoint must deliver all of the real-time data with minimal delay to maintain a high QoS.

One straightforward approach for reducing bandwidth consumption is to switch to a different compression algorithm that can compress the real-time data to a greater degree and thereby produce an output stream with a lower bit rate. Generally, real-time VoIP applications already use compression algorithms to compress real-time data before sending the data across the network because uncompressed voice data typically consumes too much bandwidth. The transmitting endpoint can further reduce bandwidth consumption by switching to a different compression algorithm that achieves a lower bit rate. The greater compression typically results in a slight QoS penalty, but this penalty is mild in comparison to the extreme QoS penalty that occurs as a result of the packet loss that normally stems from network congestion. This compression-switching approach requires both communicating endpoints to support the same set of compression schemes.

Various alternative data-compression algorithms have differing bandwidth requirements, and some compression algorithms support multiple compression ratios. For example, the ITU-T (International Telecommunication Union Standardization Sector) standard G.729 compresses audio data to 8 kbps [17], G.729 with annex D compresses audio data to 6.4 kbps, and G.729 with annex E compresses audio data to 11.8 kbps. The ETSI (European Telecommunications Standards Institute) GSM 06.90 standard, GSM adaptive multi-rate (GSM-AMR), supports multiple bit rates of 4.75 kbps, 5.15 kbps, 5.9 kbps, 6.7 kbps, 7.4 kbps, 7.95 kbps, 10.2 kbps, and 12.2 kbps [18]. A transmitter can reduce bandwidth by simply switching to a different compression algorithm in the same family.

Another method for reducing bandwidth usage is to transmit the same data with fewer transmissions, thus reducing packet-header overhead. The sending endpoint achieves

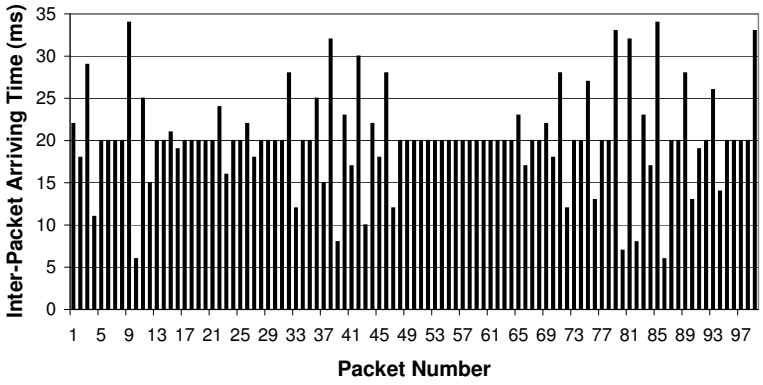


Figure 5. Inter-packet arriving interval (LAN)

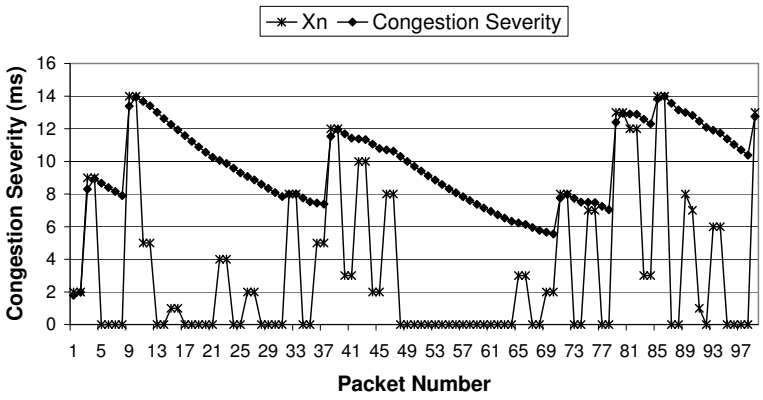


Figure 6. Estimate of congestion severity (LAN)

this goal by covering a longer period of time with the data that it packs into each IP packet. For example, a VoIP endpoint that transmits twenty milliseconds of audio data per IP packet can cut the number of transmissions by a factor of two if the endpoint sends forty milliseconds of audio data per IP packet. This approach significantly lowers bandwidth consumption by reducing the amount of bandwidth that the transmitter consumes with packet-header overhead. Obviously, this technique achieves its goal at the cost of adding a slight delay to the real-time data stream, but this approach does not trim or further compress any audio data.

A VoIP endpoint that packages twenty milliseconds of G.729 compressed audio per IP packet requires 24.0 kbps, including the overhead of the IPv4, UDP, and RTP headers at forty bytes per packet. If the VoIP endpoint packages forty milliseconds of G.729 compressed audio per IP packet, the required bandwidth diminishes to only 16.0 kbps, including header overhead. This simple scheme results in a tremendous bandwidth saving of 8.0 kbps or 33%. This approach does introduce an additional twenty milliseconds of delay into the audio stream, of course, but the effect of this small added delay on QoS is typically insignificant [19]. In fact, this method can actually *reduce* the overall delay because the technique alleviates queue delays in the routers, often more than compensating for the small delay between consecutive transmissions. Table 1 shows the bandwidth consumptions and savings with different amounts of compressed G.729 audio data in each IP packet.

The transmitting endpoint can simultaneously employ *both* of the bandwidth-reduction techniques that we have proposed since the two methods are independent of each other. The transmitting device can switch to a compression algorithm that produces a lower bit rate, and the transmitter

Table 1. G.729 Bandwidth utilization

Audio Length (milliseconds)	Bandwidth (kbps)	Packets per Second	% Savings
10	40.000	100.00	-
20	24.000	50.00	40.00%
30	18.667	33.33	53.33%
40	16.000	25.00	60.00%
50	14.400	20.00	64.00%

can concurrently package more data into each IP packet to lower the number of transmissions and save bandwidth on the packet-header overhead.

The congestion-severity information in the congestion notification that the receiver sends to the transmitter allows the transmitter to gauge its response according to the severity of the congestion. Using the congestion-severity information, the transmitting endpoint selects the bandwidth-reduction method that is most appropriate for the level of congestion, thereby maintaining optimal link utilization and throughput while simultaneously curbing congestion.

When network congestion recedes, the transmitting endpoint adjusts its transmission rate back to the original setting. The receiving endpoint detects the improvement in the congestion, and the receiving endpoint conveys this information to the transmitting endpoint. After learning that the traffic condition has improved, the transmitting endpoint increases its transmission rate to reduce delay and improve the grade of service. The transmitting endpoint can alternatively operate with the optimized transmission procedure for a predetermined period of time. When the fixed duration for using the lower transmission rate expires, the transmitting endpoint automatically resets its transmission rate.

5. Measure of improvement

To illustrate the effectiveness of our technique, let us examine the performance improvement that we can achieve. Consider a scenario in which the router throughput is 10.0 Mbps and we have 700 VoIP streams going through the router. If we use G.729 to compress the audio data and pack 20 milliseconds of audio into each IP packet, we require 16.80 Mbps of bandwidth, 6.80 Mbps more than the router can handle. As a result, the router drops an average of 20.238 packets per second for each flow. That drop rate translates into more than 400 milliseconds of lost audio data in each second for each flow, a devastating loss of more than 40%!

If the VoIP applications employ our solution and begin to package 40 milliseconds of audio data into every IP packet, we require only 11.20 Mbps for all 700 audio streams combined. That bandwidth reduction cuts the drop rate to 2.679 packets per second per flow for an average of about 107 milliseconds of audio loss in a second for each flow, a tolerable loss rate of just 10.7%. If the VoIP applications switch to use G.729 with annex D and also pack 40 milliseconds of audio per IP packet, the required bandwidth decreases to 10.08 Mbps. In this case, each audio stream loses an average of only 0.198 packets per second for a loss rate of just 0.79%, almost *zero* loss! Fig. 7 illustrates the loss comparison for the case that we have examined.

6. Delay versus packet loss

Since one aspect of our adaptive transmission control increases the overall delay by using larger, less-frequent packets, we must consider the potential negative impact of increasing the delay versus the positive impact of reduced packet loss.

We can use the ITU-T G.107 E-Model to analyze the QoS impact of the additional delay that our transmission-control method introduces into the audio stream. The E-model is an analytical model that evaluates the conversational quality of a telephony system. The E-model includes many items (e.g., room noise, echo, and circuit noise) that are independent of both packet loss and delay, but we isolate the effect of delay and thereby determine the quality differences that are due to the delay variations.

When the total delay does not exceed 100 milliseconds, the E-model indicates that there is *no degradation at all* due to the delay. This case is the relevant case for most VoIP systems since designers try to make the delays low enough to eliminate or at least minimize the effects of delays. The delay degradations remain insignificant until the overall delay reaches a level of about 200 milliseconds, and the degradation grows as the delay approaches the talker-overlap threshold of 250 milliseconds. The worst case occurs when an added delay of 20 or 40 milliseconds above the base delay pushes the total delay beyond the talker-overlap threshold, in which case the MOS rating degrades

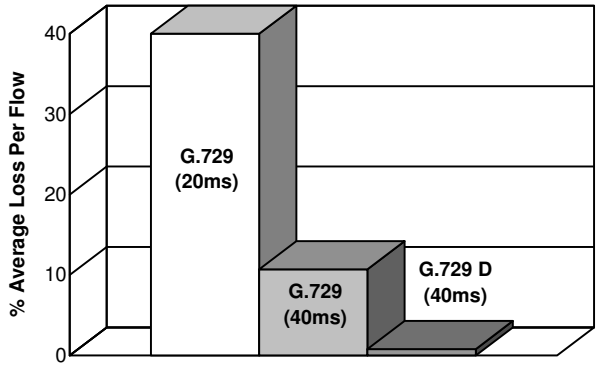


Figure 7. Loss-analysis case study

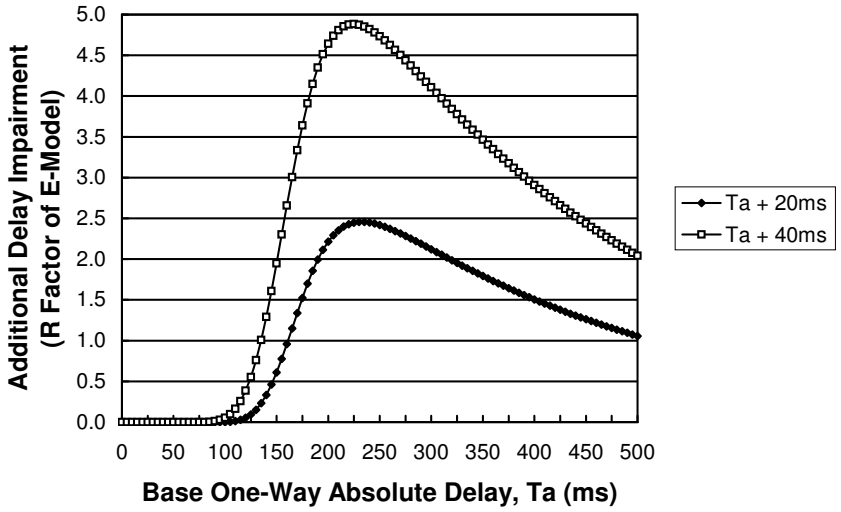


Figure 8. E-model evaluation of delay impairment

by approximately 0.1 for an added delay of 20 milliseconds or by 0.2 for an added delay of 40 milliseconds. This worst-case situation is not important for practical applications, though, because any system that is close to the talker-overlap threshold is already a marginal system for VoIP.

Fig. 8 illustrates the added delay impairment — in units of the R-factor of the E-Model — that results from the introduction of delay increases of 20 milliseconds and 40 milliseconds. Based on the E-model, the increase of 20 milliseconds in the delay in the audio stream typically has *zero* impairment to at most 2.5 units of R-factor — about 0.1 in Mean Opinion Score (MOS) — of impairment in the QoS.

In our research on packet loss in a real network, we evaluated the MOS ratings of G.729 streams transmitting at a 20-millisecond interval in a simulated real-network environment. Our study showed that a difference of 30% (e.g., from 10% to 40%) in the loss of audio data translates to a difference in QoS impairment of more than one full unit on the MOS scale [19]. Fig. 9 illustrates the MOS ratings of a G.729 stream with various degrees of packet loss in a real network.

Our results show that adding a small delay to reduce packet loss is clearly a good tradeoff. The QoS degradation from the added delay is typically zero or at most minuscule while the resulting QoS improvement from the reduced packed loss is significant.

7. Conclusion

Congestion control and congestion avoidance are popular research topics, so investigators have done considerable work in these areas. However, most of the well-known congestion-avoidance techniques exploit the transmission-control mechanism in TCP. Therefore, these approaches are

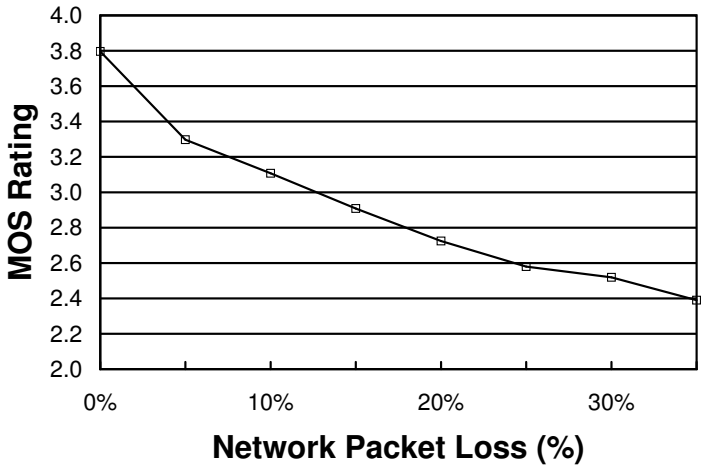


Figure 9. MOS ratings with packet loss

not effective for real-time IP applications, which typically use UDP as the transport protocol. The intrinsic characteristics of real-time media applications pose additional inherent problems to the already-challenging congestion-control issue.

Our new technique for congestion detection and adaptive transmission control for real-time IP applications such as VoIP is an application-level approach. For that reason, we can implement our solution on the current infrastructure using existing protocols. Our method requires only simple upgrades to the implementations of the transmitting and receiving endpoints. As the network becomes congested, the communicating endpoints using our method reduce bandwidth utilization to adapt to the congested environment. When the congestion subsides, the endpoints adapt to the improvement in the traffic condition and return to their original transmission settings.

Since our method is an application-level approach that runs on the endpoints, the implementation is fully scalable with respect to the size of the network and the number of flows. Each endpoint performs a simple task that demands very little in terms of processing power or memory. Together, all of the endpoints in the system cooperate to alleviate the problems that congestion poses for real-time applications such as VoIP.

In contrast to existing congestion-avoidance techniques, which simply identify and impose heavy penalties on unresponsive real-time flows, our technique cooperatively lessens the bandwidth consumption of these flows by lowering their transmission rates. As a result, real-time IP applications that use our approach experience fewer lost packets and achieve higher QoS.

References

[1] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, Vol. 1, No. 4, August 1993, pp. 397–413.

[2] Wu-Chang Feng, Dilip D. Kandlur, Debanjan Saha, and Kang G. Shin, "A Self-Configuring RED Gateway," *Proc. IEEE Conference on Computer Communications, INFOCOM 1999*, New York, New York, March 1999, pp. 1320–1328.

[3] Eric Horlait and Nicolas Rouhana, "Dynamic Congestion Avoidance Using Multi-Agent Systems," *Proc. Mobile Agents For Telecommunication Applications, MATA 2001*, Montréal, Canada, August 2001, pp. 1–10.

[4] Dong Lin and Robert Morris, "Dynamics of Random Early Detection," *Proc. ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, SIGCOMM 1997*, Cannes, France, September 1997, pp. 127–137.

[5] T. Ott, T. Lakshman, and L. Wong, "SRED: Stabilized RED," *Proc. IEEE Conference on Computer Communications, INFOCOM 1999*, New York, New York, March 1999, pp. 1346–1355.

[6] Wu-Chang Feng, K. Shin, D. Kandlur, and D. Saha, "The BLUE Active Queue Management Algorithms," *IEEE/ACM Transactions on Networking*, Vol. 10, No. 4, August 2002, pp. 513–528.

[7] Wu-Chang Feng, Dilip D. Kandlur, Debanjan Saha, and Kang G. Shin, "Stochastic Fair BLUE: A Queue Management Algorithm for Enforcing Fairness," *Proc. IEEE Conference on Computer Communications, INFOCOM 2001*, Anchorage, Alaska, April 2001, pp. 1520–1529.

[8] Wu-Chun Feng, Apu Kapadia, and Sunil Thulasidasan, "GREEN: Proactive Queue Management over a Best-Effort Network," *IEEE Global Telecommunications, GLOBECOM 2002*, Vol. 21, No. 1, November 2002, pp. 1784–1788.

[9] IETF RFC-2481, A Proposal to Add Explicit Congestion Notification (ECN) to IP, January 1999.

[10] M. May, J. Bolot, C. Diot, and B. Lyles, "Reasons Not to Deploy RED," *Proc. International Workshop on Quality of Service, IWQoS 1999*, London, UK, June 1999, pp. 260–262.

[11] Vishal Misra, Wei-Bo Gong, and Donald Towsley, "Fluid-Based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED," *Proc. ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, SIGCOMM 2000*, Stockholm, Sweden, August 2000, pp. 151–160.

[12] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm," *ACM Computer Communication Review*, Vol. 27, No. 3, July 1997, pp. 67–82.

[13] IETF Internet draft, Pre-Congestion Notification (PCN) Architecture, October 2008.

[14] S. Bangolae, A. Jayasumana, and V. Chandrasekar, "TCP-Friendly Congestion Control Mechanism for a UDP-Based High-Speed Radar Application and Characterization of Fairness," *Proc. IEEE Communication Systems, ICCS 2002*, Singapore, November 2002, pp. 164–168.

[15] P. Maryni and F. Davoli, "Load Estimation and Control in Best-Effort Network Domains," *Journal of Network and Systems Management*, Volume 8, Issue 4, December 2000, pp. 527–541.

[16] T. Chua and D. Pheanis, "Application-Level Adaptive Congestion Detection and Control for VoIP," *Proc. IARIA International Conference on Networking and Services, ICNS 2007*, Athens, Greece, June 2007, pp. 84–89.

[17] ITU-T Recommendation G.729, Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear-Prediction (CS-ACELP), March 1996.

[18] ETSI GSM 06.90, Digital Cellular Telecommunications System (Phase 2+); Adaptive Multi-Rate (AMR) Speech Transcoding, 1998.

[19] T. Chua and D. Pheanis, "QoS Evaluation of Sender-Based Loss-Recovery Techniques for VoIP," *IEEE Network*, Vol. 20, No. 6, November/December 2006, pp. 14–22.

Ambient Networks Gateway Selection Architecture

Mikko Majanen, Kostas Pentikousis and Jukka Mäkelä
VTT Technical Research Centre of Finland
Kaitoväylä 1, FI-90571 Oulu, Finland
Email: {firstname.lastname}@vtt.fi

***Abstract*— Many anticipate a future wireless world filled by a multitude of user devices and wireless technologies. Effective management of this kind of heterogeneous, mobile, and rapidly changing ad hoc networks will be a challenging task. We present and evaluate the Ambient Networks Gateway Selection Architecture (GSA), which provides support for gateway discovery, management, and selection for mobile nodes within dynamic routing groups. A routing group (RG) is a cluster of nodes in physical proximity, aware of the group membership, with a common goal of optimizing mobility management and routing functionality in the group. A gateway is a mobile node that provides packet relaying and connectivity services to other nodes in the RG. GSA can be also used outside the Ambient Networks architecture, and we present how it can be used with two existing mobility management protocols, namely Mobile IP and Host Identity Protocol, especially in the case of moving networks. Our simulation studies show the benefits gained from group formation when compared to same functionalities implemented in every individual node. We also compare the GSA hybrid signaling strategy with proactive and reactive approaches; the simulation results show that the hybrid approach scales better when the routing group size grows.**

***Keywords*—Ambient Networks, gateway selection, Host Identity Protocol (HIP), mobile computing, Mobile IP (MIP), mobility management, moving networks, routing group**

I. INTRODUCTION

Many anticipate a future wireless world filled by a multitude of user devices and wireless technologies. Effective management of this kind of heterogeneous, mobile, and rapidly changing ad hoc networks will be a challenging task. The Ambient Networks project [1] addressed this challenge by developing innovative network solutions based on the dynamic composition [2] of networks providing access through the instant establishment of inter-network agreements. The Ambient Networks concept [3] includes the Ambient Control Space (ACS) [4], which provides common control functions to a wide range of different applications and access technologies, enabling the integrated, scalable and transparent control of network capabilities.

Mobility management, a key component of Ambient Networks, can be defined as the set of functions that allow a communications system to adapt itself, seamlessly and optimally, to changes in physical and logical topology of the network. A goal of the Ambient Networks mobility solution is to provide a framework within which existing mobility solutions can be deployed and interoperate, whilst ensuring that new mobility solutions can be added as and when they become available. Novel mobility concepts (e.g., see [5]) have been developed to better support moving groups of nodes and users, such as personal area networks and networks formed in mass transport vehicles, such as commuter trains.

Within Ambient Networks, nodes in a moving network can be linked to form a cluster referred to as the Routing Group (RG) [6]. Let us clarify the distinction between the terms *cluster* and *routing group*. Take a set of mobile nodes, U, and a set of base stations, B, connected to the wired network. Each $b_j \in B$ can provide wireless connectivity to all nodes $x \in U$ within its coverage area. A cluster, $S \subseteq U$, is defined as the set of nodes from U that can (i) communicate with each other, (ii) are physically close to each other and, (iii) are likely to remain so. Although (i) and (ii) can be determined using information from layers 1–3, (iii) can be determined only by taking into consideration other situational and context information. Identification and formation of such clusters can enable communication and shared use of applications, while several other optimizations, related to routing and mobility management can be pursued.

In each cluster one node is elected to act as the *cluster head*. Each cluster head is aware of the cluster topology, including the nodes and their roles. Within each cluster, one or more nodes can act as gateways, relaying packets for other nodes and providing connectivity to other networks. A routing group (RG) is defined as the set of nodes $R \subseteq S$, in which the nodes are aware of group membership. This allows even more possibilities for optimizations than a cluster.

In previous work [7], the Gateway Selection Architecture (GSA) was introduced to provide support for gateway identification, management, and selection within a routing group. Of course, one might expect that by grouping nodes and delegating mobility management to the cluster head and the gateways certain performance optimizations are possible as discussed in [7]. Later, the performance of the GSA was evaluated by simulations in [8]. In this paper, we willl elaborate GSA, provide a detailed description of GSA and present performance evaluation results, delivering for the first time a complete coherent view of GSA.

The paper is organized as follows. In Section II we take a look at related work in the area of mobility management. Section III describes the GSA architecture and Section VI shortly compares GSA to other related work and discusses the possible benefits of GSA. In Sections IV and V we describe how the GSA architecture can be used outside the Ambient Networks framework with existing mobility management protocols, namely Mobile IP (MIP) and Host Identity

Protocol (HIP). Section VII describes the simulation scenario and results and Section VIII concludes the paper. Table I lists the acronyms used throughout the paper for easy reference.

II. Mobility Management

In the context of mobile/wireless networks three approaches have been followed with respect to gateway discovery. The first is a *proactive* strategy whereby the gateways broadcast advertisements to the whole network. The nodes requiring gateway services choose the most suitable gateway based on the advertisements they received. In *reactive* strategies, the initiative lies with the nodes, which broadcast request messages to the network and select the most suitable gateway based on the replies that are unicasted to them. In *hybrid* strategies, gateway advertisements are usually broadcasted only to the nodes "near" the gateway. For instance, the advertisements may have a limited time to live (TTL) value, say, three hops. Nodes farther than this amount of hops have to use request messages to receive gateway services. That is, if a node x does not receive a broadcasted advertisement from any gateway, it will broadcast a gateway request message.

Gatewaying can be seen as a service, so the gateway discovery problem is similar, to some degree, with the general service discovery problem. The Service Location Protocol (SLP) [9] is an IETF protocol for service discovery and advertisement. There are three entities in the SLP: service agents (SAs), user agents (UAs) and directory Agents (DAs). SAs advertise the service to the network or to DAs, UAs try to find services for the applications. DAs cache the information about available services based on SAs' advertisements.

The most popular way to provide Internet access to nodes within ad-hoc networks and in mobile networking scenarios seems to be extending the Mobile IP (MIP) protocol for either IPv4 or IPv6 networks. In the following subsections we briefly go through the basics of MIP and study how it has been extended to work with moving networks.

A. Mobile IP

In MIPv4 [10], the base station (BS) nodes act as Home (HA) and Foreign Agents (FA) for the mobile nodes. The HA keeps a list of mobile nodes that are attached to it, i.e. the mobile nodes that belong to the same subnet as the HA. When the mobile node moves away from the HA, it eventually starts using another BS as its network connection point. The new BS will act as FA for the mobile node and it provides a care-of address (CoA) from its subnet address space for the mobile node. The CoA is transmitted also to the mobile node's HA, which establishes a tunnel between the HA and FA. Tunneling means that packets destined to the mobile node are forwarded from the HA to the FA using IP-in-IP encapsulation [11]. The FA decapsulates the packet and transmits it to the mobile node, which is currently in its subnet. Thus, the HA and FA nodes (i.e. BSs) act as gateways for the mobile nodes.

In MIPv4, the HAs and FAs advertise themselves by broadcasting periodically beacons, i.e. a proactive approach is adopted. However, if the mobile node does not have a connection to any BS, it may broadcast a solicitation message to find one. BSs that receive the solicitation message will reply by sending the beacon packet. Thus, MIPv4 supports also the reactive approach, even though it mainly relies on the use of proactive approach. The beacons are not forwarded; MIP supports only one wireless hop.

In MIPv6 [12], the mobile node has the FA functionality built in. When the mobile node is outside its home network, it sends a binding update to its Home Agent informing its current care-of address. It may also send the binding update to its correspondent node if that supports MIPv6. In that case, packets from CNs may be routed directly to the mobile node's care-of address, without going via the HA. In addition to the optimal route, the overhead is also smaller since instead of IP-in-IP encapsulation, IPv6 routing header can be used.

B. Extending MIP to multi-hop ad hoc networks

Since MIP supports only one wireless hop, several approaches have been presented to extend MIP to make Internet connections available for the ad hoc network nodes that do not have a one hop route to the FA. Sun et al. [13] present an architecture where MIP is combined with the Ad hoc On-Demand Distance Vector (AODV) protocol [14] and a reactive approach to solicit FA advertisements is used. Ratachandani et al. [15] on the other hand use a hybrid approach where the FA advertisements are flooded within a limited number of hops from the FA; nodes outside this hop limit use reactive approach.

The simulation studies in MIPMANET (Mobile IP for Mobile Ad Hoc Networks) [16] show that it is highly valuable to be able to choose the closest access point to the Internet since it reduces the overall load in the moving network. In the scenario used in [16], broadcasting the MIP FA advertisements was found better than unicasting them to each MIP using node inside the moving network. Unicasting the advertisement meant in this case that the FA unicasted the advertisement to every moving node that was registered with it. The broadcasting approach provided better options for mobile nodes to change the FA to a better one since the advertisements were broadcasted periodically. In the unicasting approach, the mobile nodes used solicitation messages when they did not have a connection to any FA.

Lee et al. [17] propose a hybrid GW advertisement scheme for connecting ad hoc networks to the Internet. In that approach, Dynamic Source Routing (DSR) [18] is used as the ad hoc routing protocol. Unnecessary flooding of GW discovery packets is avoided by using advertisement schemes based on the mobility and traffic patterns of the moving network.

Ghassemian et al. [19] present a performance comparison between proactive, reactive and a hybrid GW discovery approaches. In the hybrid approach, the GW advertisements' time to live was limited, and nodes further away had to use a reactive approach to solicit advertisements. In the scenario considered, the proactive approach performed best in case of packet delivery ratio and the packet delay. The reactive approach performed worst and the hybrid one was between

these two. On the other hand, with respect to signalling overhead, the reactive approach was better than the proactive one.

C. Network Mobility (NEMO)

In all previous approaches, the GW nodes, i.e. the BSs, are stationary, so they are not moving. Also, all nodes in the moving network perform mobility management actions independently.

The Network Mobility (NEMO) Basic Support Protocol [20] extends MIPv6 to manage network mobility. A similar protocol has been proposed also for IPv4 moving networks in [21]. NEMO enables reachability and session continuity for all nodes belonging to the moving network. With NEMO, mobility is transparent to the moving network nodes. This is achieved by introducing a special Mobile Router (MR) node that connects the moving network to the Internet. The MR binds a network prefix with a care-of address (CoA) indicating its current location together. MR uses binding update messages to inform its current CoA to its HA. Nodes within the moving network are allocated an address from the MR's prefix. Thus, they can connect to the Internet without having to participate in the mobility management since the MR updates the HA for the whole network, not just for itself. Traffic destined towards the moving network (i.e. MR's network prefix) is intercepted at the HA and tunneled to the MR using IP-in-IP encapsulation. MR decapsulates the packets and forwards them to the correct mobile node. In the opposite direction, reverse tunneling is used, i.e. packets are tunneled from the MR to the HA, and then directed towards the correspondent node. The NEMO Basic Support Protocol does not support route optimizations to correspondent nodes.

NEMO solves the basic problem of network mobility but, since it is MIP-based, it has some disadvantages inherent to MIP: MR introduces a single point of failure on the routing path, tunneling adds overhead, and the routes are not optimal (so called dog leg routes) since the binding updates to CNs are not supported.

D. MOCCA

The Mobile Communication Architecture (MOCCA) [22] is designed for inter-vehicular systems that consists of vehicular ad hoc networks, road-side Internet Gateways (IGWs), and a proxy between IGWs and the Internet. MOCCA uses a modified version of Mobile IP (called Mobile IPv6*) to support the mobility of vehicles. The Proxy maintains the vehicles home agents (HAs), IGWs function as foreign agents (FAs) and the vehicles represent the Mobile Nodes (MNs). The Correspondent Node (CN) in the Internet sends its data packets to the MNs home address (i.e. the HA in the Proxy). The Proxy tunnels them to the FA, which decapsulates and forwards them to the MN. The Proxy also separates the transport layer end-to-end connections in order to prohibit e.g., TCP connections to time out.

Since MIP does not support multi-hop ad hoc networks (the MN may be more than one hop away from the IGW), MOCCA employs a modified version of the Service Location Protocol [9] for discovering the IGWs. In MOCCA, the Service Agent is located on the IGWs and it announces periodically its Internet access service. The service advertisements are geocasted, i.e. broadcasted in a geographically restricted area. The Directory Agent is located in the vehicles. It extracts the information from the service announcements and caches it to a local database. The advertisements include information about the current number of clients using the IGW, IGW's available bandwidth, geographical position and optionally some other information. The User Agent (i.e. mobile device) within the vehicle queries the database and configures Mobile IPv6* to use one of the available IGWs as its FA. In case multiple IGWs are available, the UA selects the IGW that fits best to the requirements of the applications. The selection is made based on a fuzzy logic algorithm that predicts QoS parameters like expected delay, dropouts and the probability of disconnection for the connection.

The MOCCA implementation covers both network and transport layer protocols. As such, it does not support the mobility of legacy applications running on devices inside the car. For supporting legacy applications (without modifications to those), MOCCA includes also another proxy inside the vehicle. This proxy hides the device from the Internet, so it is not reachable outside the vehicle. However, the device can access Internet services. To be reachable outside the vehicle, the device should additionally support MIPv4, too.

E. Host Identity Protocol

The problems in MIP-based mobility are based on the TCP/IP stack architecture. The IP address is used for both identifying the host and stating the host's current network attachement point, i.e. the location of the host. When the host is moving, it has to change its attachement point, which means changing its IP address. On the other hand, the transport layer connections are bound to certain IP address and port. Keeping the transport layer connections while moving requires also update on the transport layer.

In the Host Identity Protocol (HIP) architecture [23] the host identifiers and locators are separated. A new layer is introduced between transport and network layers. Transport layer connections are not anymore bound to IP address and port, instead a Host Identifier (HI) is used. IP address is used only for forwarding packets. This allows new possibilities for mobility and multihoming, as described e.g., in [24]. HIP-based mobile routers, like [25] and [26], are especially interesting from the moving network support perspective.

III. Gateway Selection Architecture in Ambient Networks

Both proactive and reactive gateway discovery schemes have their pros and cons. A reactive approach does not create unnecessary traffic, but on the other hand, it exhibits longer delays. A proactive approach comes with smaller delays, but introduces possibly unnecessary traffic. On the other hand, ability to select the most suitable gateway is worth of some

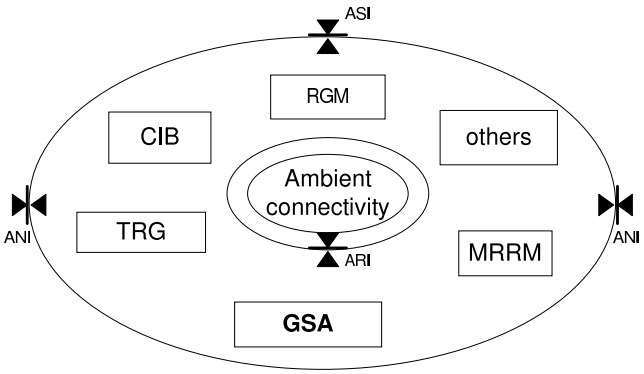


Fig. 1. GSA as part of ACS

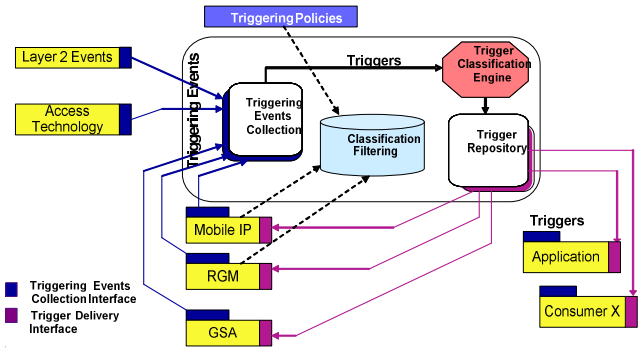


Fig. 2. TRG components

extra traffic, as argued in [16]. Proactive approach suits better for this purpose since the status of the gateway is updated periodically in the advertisements and the new gateways can be discovered earlier.

GSA adopts a hybrid approach for gateway discovery, introducing a special kind of nodes called gateway selectors (GWS). In GSA, service advertisements and requests are unicasted to the gateway selectors, thus simplifying information dissemination and updates regarding gateway (or more enhanced mobile router) nodes and their capabilities. This should not only decrease the amount of signaling overhead, but also allow the majority of the nodes to have only limited computational capabilities and battery power by keeping the intelligence in the gateway selectors. By introducing GWS nodes, GSA borrows a little from the Service Location Protocol (SLP) [9], with GWSs resembling to Directory Agents, gateways to Service Agents, and other RG nodes to User Agents.

As illustrated in Figure 1, GSA is part of the ACS and it is supported by many other ACS functional entities such as triggering (TRG) [27], [28], Routing Group management (RG) [6], context information management (CIB) [29], [30], and multi-radio resource management (MRRM) [31], which are capable of providing a wealth of information related to gateway discovery and selection. GSA is designed to utilize this extra information aiming at making optimal gateway selections.

Figure 1 shows also the three interfaces that are used to access the ACS functionalities. The Ambient Service Interface (ASI) is used by higher layer applications and services to issue requests to the ACS concerning the establishment, maintenance and termination of the end-to-end connectivity. The Ambient Resource Interface (ARI) is used for managing the connectivity plane resources such as routers, switches, and radio equipment. The Ambient Network Interface (ANI) is used for transferring information between different Ambient Networks.

The triggering (TRG) functional entity is a vital part of the ACS since it informs the other functional entities about different events in the Ambient Networks. Main elements of the TRG, as detailed in [28], are the entities which create events (producers) and the entities that use the trigger information (consumers). TRG collects the event information from various producers via a specific collection interface, processes the collected events and distributes the created triggers to the interested consumer entities. A producer, as well a consumer, can be any entity implementing the collection interface. In other words, the same entity can act both as a producer and a consumer. Figure 2 illustrates TRG with different producers and consumers.

TRG might have several event collectors, which may be distributed, collecting different types of events. A number of collectors might be needed since the producer might be the entity implemented in kernel space or an application in user space. Having a separate collector per producer entity with a dedicated inteface allows the communication between nodes with different operating systems as well.

In order to use the collection interface, producers need to register their triggers with TRG. By registering, each producer and their triggers can be identified and, further on, interested consumers can subscribe to get certain identified triggers. All this is a part of the processing mechanism that supports also the filtering of triggers. With filtering, consumers get only those triggers they are subscribed to. Using this filtering functionality together with the support for system wide policies, TRG can not only provide the way for efficient distribution of right triggers to the right consumers, but also provides a way to control consumer access to event sources.

Figure 3 shows the internal structure of the GSA functional entity. GSA uses TRG for implementing its signaling, i.e. the gateway advertisements and requests are transmitted as triggers. GSA includes a GSA Trigger Consumer for receiving triggers. Depending on the trigger received, its information may be stored to the GSA Parameter Collection or Policies data storage, and/or it may be further processed by the GSA Decision Engine. Based on the processing results, a new trigger may be generated and sent by the GSA TRG Producer. The actual behaviour of the trigger processing depends on the node's role, i.e. whether the node is a GW, GWS, or GW service user.

The gateway node's GSA Decision Engine uses the GSA Trigger Producer to periodically (or when needed) generate updates about its GW service status by sending a gateway

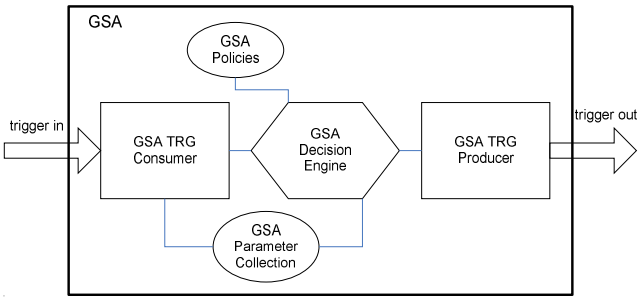


Fig. 3. GSA internal structure

advertisement trigger to TRG. The GW service parameters are maintained in the Parameter Collection data base. The GSA TRG Consumer subscribes to all triggers related to the node's context, RAN status, and so on. Policies may contain rules, such as whether the node is allowed to provide the GW service to other nodes.

An RG node starts the GW request process when its GSA Trigger Consumer gets a request trigger sent by an application. The application communicates with the ACS via the ASI interface. The GSA Trigger Producer creates a GW request message including service requirements, and sends it to TRG.

Subsequently, GWS's GSA Trigger Consumer receives the GW advertisement triggers and stores the status information of the gateway node to Parameter Collection. It also receives the GW request triggers from RG nodes that need GW service. GWS's GSA Decision Engine compares the service request parameters to the available gateways' parameters and selects the best match. The result is transmitted to the requesting node in a form of GW response trigger containing the address of the gateway and its GW service parameters. The actual algorithm to select the best GW is out of the scope of this paper but, for example, it can be a weighted sum over selected parameters (this approach is used e.g., in the selection of the cluster head node in [32]).

Usually, TRG is located on the same node as GWS, so the communication between TRG and GWS is node-internal and does not consume network resources. If the RG has also a cluster head, it is usually collocated also on the same node. If the cluster head (i.e. the RGM entity in Figure 1) or TRG is located at a different node than GWS, the information is then transmitted as triggers between the nodes. Thus, GWS's GSA Trigger Consumer also receives and GSA Decision Engine handles triggers dealing with e.g., topology changes in the routing group. In every case, GWS has always up-to-date information about the RG and its nodes. Actions (e.g., re-selection of the GW for certain RG nodes) are launched whenever deemed necessary.

Although GSA was designed originally to work within the Ambient Networks architecture, there are no reasons why GSA could not be used also outside Ambient Networks. In Ambient Networks the ACS binds the different functional entities together, but on the other hand, these entities, or the information they produce, may be used also separately.

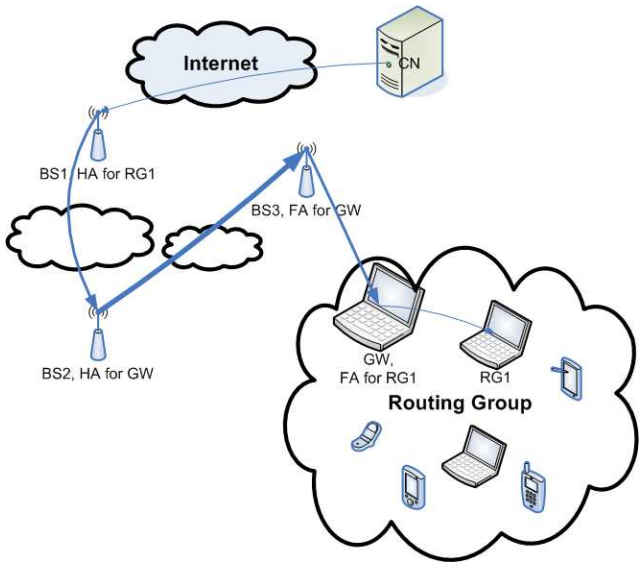


Fig. 4. Moving FA as a gateway for Routing Group nodes

For example, TRG and GSA can be set up to any moving network; they are not dependent on any Ambient Networks architecture specific entities. Actually, TRG is the first step in the Ambient Networks migration plan [33]. Gateway selection related triggers are then perhaps produced by some other entities as in Ambient Networks, but still, GWS can make the decisions based on the information that is available. In fact, GSA can be used even without TRG; its principles can be easily applied to existing MIP and HIP implementations for mobility management optimizations for moving networks. In the following two sections we briefly explain how this can be done.

IV. GSA WITH MOBILE IP

In MIPv4 [10], base stations act as Home and Foreign Agents (HA and FA, respectively) for mobile nodes. In moving networks, the gateway nodes can act as FAs for all nodes in the RG, forming a hierarchical set of FAs as illustrated in Figure 4. HAs are still located at the base stations. The gateway nodes use base stations as their FAs, so they handle their own mobility like normal mobile nodes in MIP. Alternatively, we can call these gateways as NEMOv4 Mobile Routers since this is how they work. The traffic destined to RG nodes goes via two HAs and two FAs before reaching the destination. The line width in Figure 4 illustrates the tunneling overhead between HAs and FAs.

The gateway discovery and selection process starts with the election of GWS. Normally this functionality lies with the same node as the cluster head. The cluster head collects and manages information related to the RG management. Gateway issues are part of this management so it is natural to include GWS functionality in the same node. The GWS and cluster head can be also on different nodes but that may add some more overhead due to the information exchange between them. The elected cluster head (and GWS) node informs the whole

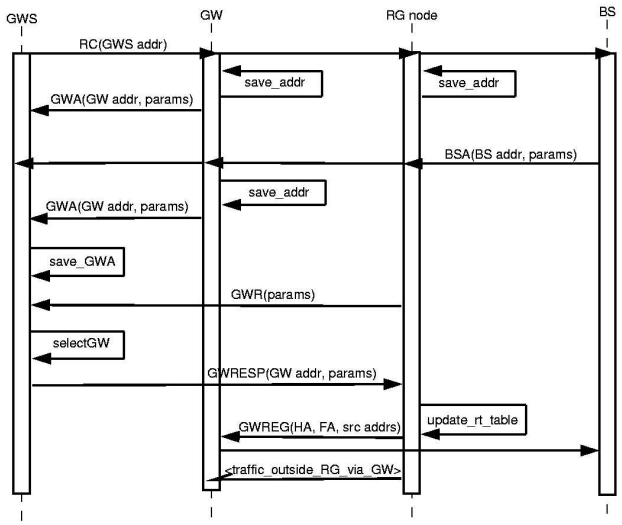


Fig. 5. GSA signalling in MIP-like mobility management

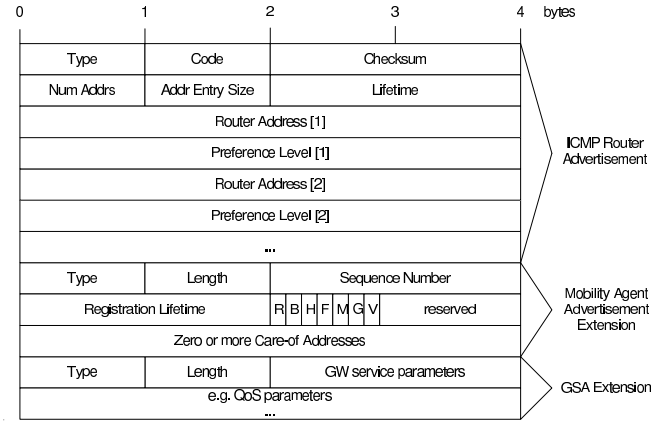


Fig. 6. ICMP Router Advertisement message header extended with MIP and GSA extensions

RG about its role by broadcasting a role claim (RC) message (see top part of Figure 5). RG nodes save the address of the GWS.

MIP makes use of ICMP [34] Router Advertisement and Solicitation messages. The ICMP Router Advertisement message is extended by a Mobility Agent Advertisement Extension that contains e.g., the care-of address(es) of the BS and the lifetime of the advertisement. The ICMP Router Solicitation message is used unchanged by MIP. GSA utilizes the same messages as MIP, but extends them by using optional TLV-encoded fields for providing extra information (e.g., QoS parameters), as depicted in Figures 6 and 7. Note that the messages are no longer broadcasted, as depicted in Figure 5.

Base stations (BS) broadcast periodically their own advertisements that contain the care-of address(es) of the BS and optionally some other information about the BS (e.g., QoS parameters). We call these extended MIP BS beacons Base Station Advertisement (BSA) messages. BSA messages are not forwarded inside the RG (they carry a TTL set to one). RG nodes that receive a BSA message and are willing to act

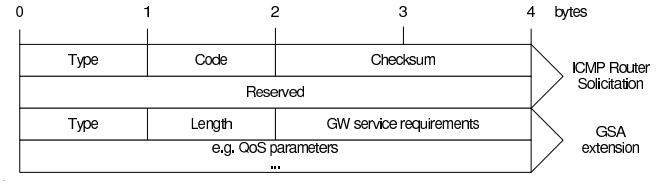


Fig. 7. ICMP Router Solicitation message header extended with GSA extension

as a gateway exploit the information contained in the BSA in forming a Gateway Advertisement (GWA) message describing the gateway service it can provide.

GWA messages are unicasted to GWS by gateway nodes in response to the received BSA message. A GWA message is also sent as a response to a received RC(GWS) message, that is, when the RG is formed and the GWS is selected. The GWA message includes the address of the GW and its parameters, for instance, bandwidth, battery state of charge, supported radio access networks, and connection monetary cost (free of charge vs. charge based on traffic volume or connection duration), to name a few. In short, a GWA is another kind of MIP BS beacon, extended with some additional information just like BSA, but instead of broadcasting it with TTL=1 it is unicasted to the GWS (with TTL>1). So, in GSA, both the BS and GW nodes send extended MIP BS beacons (as illustrated in Figure 6).

The gateway selector saves the gateway's parameters to the list from the received GWA message (Figure 5, middle part). The RG nodes make a gateway request (GWR) to GWS when they need gateway service. The GWR message contains requirement parameters for the GW service (same as the GWA message). As depicted in Figure 7, it is a type of extended MIP solicitation message. When GWS receives a GWR message it browses through its list of gateways and selects the most suitable one. GWS replies with a response message (GWRESP) that includes the address of the selected GW and its parameters (a sort of extended MIP BS beacon). In case the node is not satisfied with the service chosen/available, it may cancel/postpone the connection or make a new request. Otherwise, it updates its routing table so that the traffic destined outside the RG is routed via the selected gateway. It also sends a registration message (GWREG) to its HA (as is the case in MIP).

V. GSA WITH HOST IDENTITY PROTOCOL

In the Host Identity Protocol (HIP) Architecture [23] hosts are identified by public keys (Host Identities), not with IP addresses. This helps in mobility and multi-homing issues since the nodes can change their IP addresses and still be reachable via the same Host Identity.

The HIP base exchange [35] allows any two HIP-enabled hosts to authenticate with each other and create a HIP association between them. The base exchange consists of four packets: I1 is the trigger packet sent by the Initiator to the Responder. I1 contains only the Host Identity Tag (HIT) of the Initiator and possibly the HIT of the Responder (if

known). The second packet, R1, starts the actual exchange. It contains a puzzle, initial Diffie-Hellman parameters and a signature covering part of the packet. I2 contains the solution to the puzzle, Diffie-Hellman parameter for the Responder. The packet is signed. R2 is a signed message finalizing the base exchange.

Before starting the base exchange, the Initiator has to acquire the Responder's IP address. The HIP Rendezvous Extension [36] introduces a Rendezvous server (RVS) that serves as an additional initial contact point for its client HIP nodes. With RVS, the initial contact can be made by using RVS's IP address. RVS's clients become reachable via RVS's IP address. This is very beneficial in case of mobile nodes that change their network attachment point, and thus also their IP address, frequently. After the base exchange, the communication is based on Host Identities, even though the IP address changes have to be signalled to the peer hosts so that packets can be routed correctly at the IP layer. Address changes are made by sending an UPDATE packet containing the new location information.

The base exchange can also include information about available or requested services [37]. A HIP host capable and willing to act as a service provider includes also the REG_INFO parameter in its R1 packets, thus announcing its available services. The UPDATE packet can also be used for this purpose if new services become available after the HIP association has been established. To request registration with a service, a requester includes a corresponding REG_REQUEST parameter in an I2 or UPDATE packet.

There are two ways for HIP nodes to initiate the service discovery process [38]. In the so-called on-path service discovery a HIP node sends a Service Discovery Packet (SDP) towards the peer node in the Internet (for example its own RVS). Each host on the SDP's path that provides services responds with a Service Available Packet (SAP). SAP may contain information on all services it provides. Alternatively, in case the SDP requested only a particular service, only those services are included in the SAP. SAP also includes the R1 parameters. Thus, after receiving a SAP, the HIP base exchange can be completed with I2 and R2 messages; SDP corresponds to the I1 packet in this case. If the HIP node wants to search services available only on a certain network region, it may use different multicast addresses instead of the address of the peer node in the Internet.

In certain cases it is not feasible to use the on-path service discovery. The HIP hosts can then use the so-called passive discovery method. In this method, the HIP service providing nodes "sniff" passing HIP packets. If a packet fulfilling certain conditions is detected, a SAP can be created and sent to the HIP node that originated the matched packet.

GSA can be used also with HIP, especially in moving networks with HIP-based Mobile Routers. The same principles apply as with MIP: certain messages are extended with some additional information and the destination of some messages may be different than in normal HIP service discovery. All HIP packets contain a common header part and optional TLV-encoded parameters, as shown in Figure 8, so extending HIP packets with MR selection related extensions is straightforward. The signalling is depicted in the Figure 9.

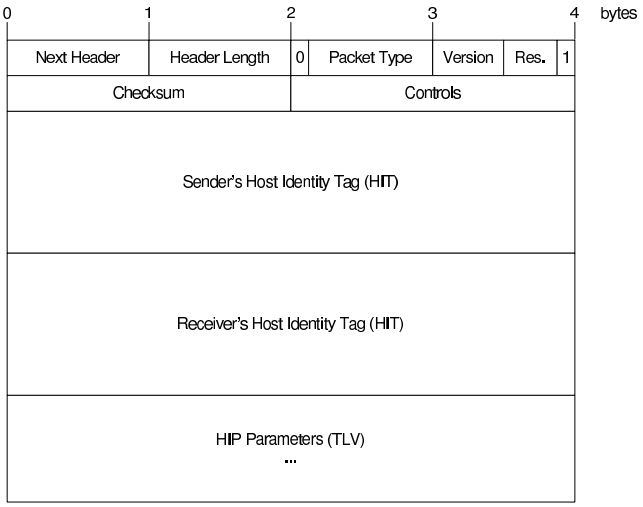


Fig. 8. HIP packet header format

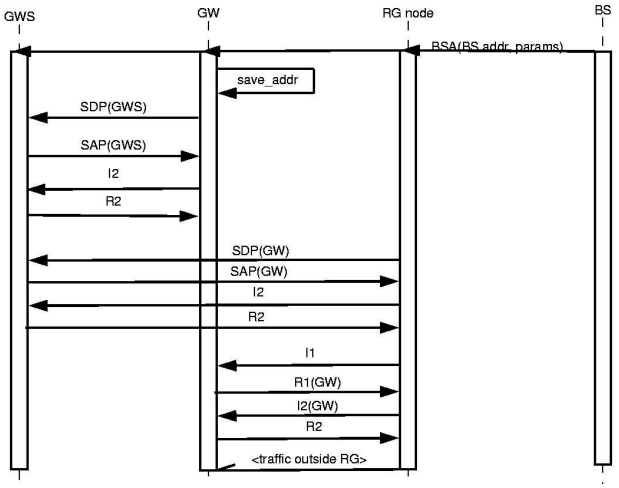


Fig. 9. GSA signalling in HIP-like mobility management

In case of HIP, GWS can be seen as a service. GWS provides the MR selection service. Nodes capable and willing to provide mobile router service register with GWS. Four-way base exchange extended with service discovery and registration information is needed for that at first time. The IP address of the GWS is known since it is signalled during the routing group formation process. After registration, MRs can use UPDATE packets as their MR service advertisements. HIP nodes requiring MR services can send their requests to the GWS, which replies with the best available MR for the requester's needs. After that the requester registers with the MR service and starts using it. UPDATE packets can be used in case of any changes regarding the service, e.g., location updates.

VI. Discussion

Of course, one might expect that by grouping nodes and delegating mobility management to the cluster head and the gateways certain performance optimizations are possible as discussed in [7]. The motivation for including the GWSs aims at simplifying the signalling overhead regarding gateways and their capabilities. By using GWSs, the topology for advertising and finding a GW is a unicasted star rather than the whole network flooded with messages. As a hybrid solution it has the benefits of both proactive and reactive approaches: the status of the gateways is known in the GWS all the time due to the advertisements, but the whole network is not unnecessarily flooded with them. The nodes requiring gateway service, request it from the GWS; the requests are not flooded to the whole network. New gateways are discovered as they become available, and GWS can direct the nodes to use them if they are more suitable for the nodes. GSA also allows the majority of the nodes to have limited computational capabilities and battery power because the intelligence is kept in the GWSs; thus, there is no need for spending so much resources (e.g., battery or computation power) in the other RG nodes.

A moving network such as a NEMO network can be seen as a RG with only a single MR. As identified in section II, dog leg routing is an issue in NEMO (or in general, MIP) based moving networks. This is especially true if there are nodes belonging to another home network as opposed to that of the MRs home network. In that case, all packets destined to or sent by such nodes need to go through two Home Agents, as illustrated in Figure 4. GSA is advantageous in this situation since the RG may have also other GWs as the MR. One GW could have the same home network as the other nodes in the RG and then the GWS, using context information, may direct them to use that GW instead of the MR. There might be also some other situations (e.g., load balancing) when the MR is not the best option for all RG nodes; in these situations GSA can provide for better GW selection results for the nodes.

Another drawback of the NEMO architecture is that the MR adds a possible single point of failure to the moving network. In GSA, GWS could be able to direct the nodes to use possible other GW nodes in case the MR fails. On the other hand, GWS nodes may fail in the GSA. This is why the RGs can have also secondary GWS nodes containing the same functionalities.

In the MOCCA architecture [22], the service agent inside the moving car caches the service advertisements sent by the road side gateways. Nodes inside the car ask the list of available gateways from it and select the most appropriate GW into use based on a fuzzy logic algorithm. So, MOCCA uses partly the same approach as GSA since the advertisements are gathered in one place and service users ask them from there. However, in MOCCA, and in the approaches that extend MIP to work in multi-hop ad hoc networks (such as [13], [15], [16] and [17]) the mobile nodes handle their mobility individually (as opposed to NEMO or HIP Mobile Routers) and make the GW decision by themselves. In GSA, the GWS makes the decision. The gateway or mobile router has to be moving along with the moving network so that mobility management can be hidden from the nodes in the moving network, which is the case in GSA architecture.

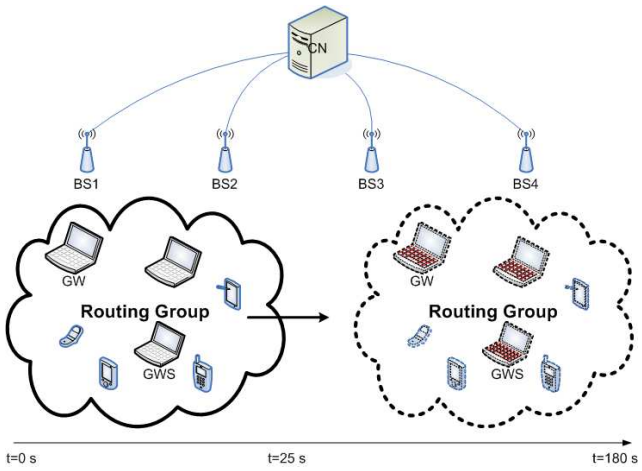


Fig. 10. The simulation scenario

VII. Evaluation

In the following two subsections we attempt to quantify the benefits of the GSA in simulation scenarios where several nodes move together in a mass transit vehicle.

A. Methodology

We use the ns-2 network simulator (version 2.28) [39] to evaluate GSA's MIP-like mobility management (as illustrated in Figure 4) in a commuter train scenario, as illustrated in Figure 10 (the figure is not to scale). We are interested in quantifying (a) the gains of GSA vs. standard MIP and (b) the advantage of GSA vs. general proactive and reactive strategies. The scenario includes a commuter train (total length=70 m, approx. 3 wagons; wagon width=3 m), and n passenger devices, which are randomly distributed inside the 210 m^2 area of the commuter train. For the purposes of this study, we configure only one mobile device to act as a gateway. The gateway functionality was implemented in ns-2 by adding the BS node's FA functionality to a mobile node, too.

During the first 25 s of the simulation, the mobile devices form a single, stable routing group. At $t = 25$ s the train starts moving at a constant speed of 11 m/s along a straight railway track. From $t = 25$ till $t = 180$ s the train passes by four base stations located along the railway track. The base stations are placed far from each other so that there is no coverage area overlapping. The first one, BS1, was configured to be the HA for all mobile nodes. The base stations were connected to each other via wired links and a wired node. The wired links have a bandwidth of 100 Mb/s with propagation delay set to 2 ms. The wired node also acted as a correspondent node to a mobile node, by sending constant bit rate UDP traffic to one of the mobile devices on the train. The IEEE 802.11 MAC layer data rate was set to 11 Mb/s, and it was used by all nodes in the train, using the free space signal propagation model and the

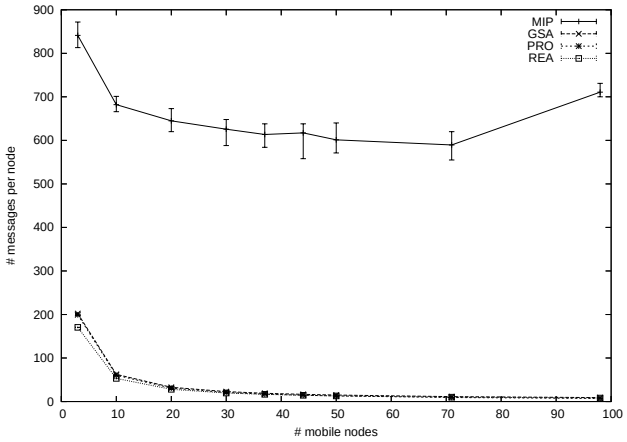


Fig. 11. Mean number of sent messages per node using MIP, GSA, proactive (PRO), and reactive (REA) approaches, excluding RG formation and routing protocol messages; error bars indicate min and max values

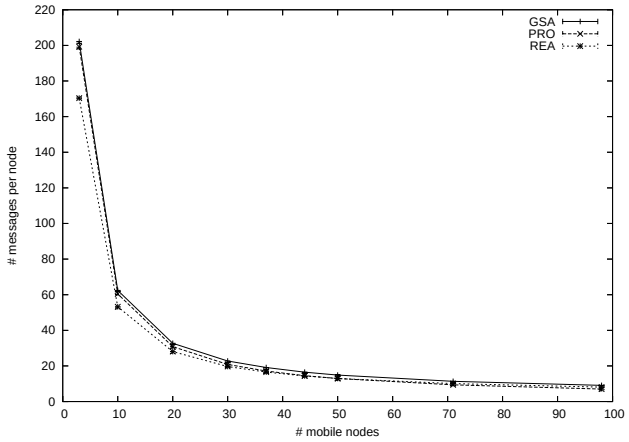


Fig. 12. Zoom of Figure 11 for GSA, proactive (PRO) and reactive (REA) approaches

DSDV routing protocol inside the RG. For RG formation and management we used the stability-based clustering protocol described in [32]. Unless mentioned otherwise, we run the simulation using the default values and settings in ns-2.

We evaluate the performance of GSA centering on the amount of required control signaling and compare it primarily with proactive and reactive algorithms, but also with the case where every mobile node manages its own mobility using MIP. As such, in all results reported below, we consider only the signaling required to provide the *same* functionality that MIP provides, and we exclude, for instance, routing group formation related signaling and DSDV messages. Further studies of MIP covering, for example, the effect of the velocity to the handoff, throughput and packet loss are presented in [40] and the references therein. The following subsection presents results from ten independent replications, for each of the scenario configurations presented above.

B. Results

First, we consider the number of sent messages per mobile node in either of the four alternative strategies. Figure 11 presents the mean number of sent control messages per mobile node. The error bars indicate the min and max values. The standard deviation σ varies between 9 and 24 with MIP and 0.02 and 0.6 in all other cases. Overall, as the routing group size increases, on average, nodes send fewer control messages. Clearly, forming a routing group is beneficial as compared to having each and every node use normal MIP to manage its mobility. The gains are typically an order of magnitude and increase as more nodes are added to the routing group. For example, in the case of $n = 3$, on average per node, MIP has to send more than four times the number of messages than GSA. At the other end of the range we explored, with $n = 98$ the difference is over 78 times. This is because there are many more nodes replying to BS advertisements and sending registration updates to HAs in MIP than in case of a routing group.

Comparing GSA with proactive and reactive approaches only (Figure 12), which also take advantage of group formation, we note that GSA underperforms. When employing GSA, on average, nodes have to send more messages than if they had used a proactive and reactive approaches. This is due to its hybrid strategy. When $n = 3$, using GSA nodes transmit 1.5% and 18.7% more signaling packets than when using proactive and reactive approaches, respectively. As n increases, the proportional difference between the number of messages sent by GSA and proactive approaches increases, reaching 29.1% when $n = 98$. Similarly, GSA's hybrid strategy underperforms the reactive approach as the number of nodes increases, although the proportional difference becomes smaller: with $n = 98$, GSA nodes send, on average 12.4% more messages.

This underperformance is due to the hybrid strategy GSA adopts. Both gateway nodes and routing group members send advertisements and requests, respectively, to the gateway selector. If a proactive strategy is used, then only the former messages are sent, while if reactive is opted for, only the latter messages are needed. Nevertheless, in GSA all messages are *unicasted* whereas in the proactive and reactive approaches, all messages are *broadcasted*, which forces mobile nodes to spend resources processing these messages regardless of whether they are useful in their current state. Broadcasting is a considerably "heavier" operation when compared to unicasting. Moreover, when employing the reactive approach, as implemented in the simulation model, it was not possible to re-select the gateway before the connection was lost (break-before-make handover). On the other hand, in both GSA and proactive approaches the status of the gateways is reported in the advertisement messages periodically, enabling seamless, make-before-break handovers and better load sharing, which may assist avoiding congestion incidents.

Figure 13 presents the amount of processed control messages and tells a similar story with Figure 11 with respect to the benefits of forming a routing group as opposed to using standard MIP. We refer to *processed control messages* as all

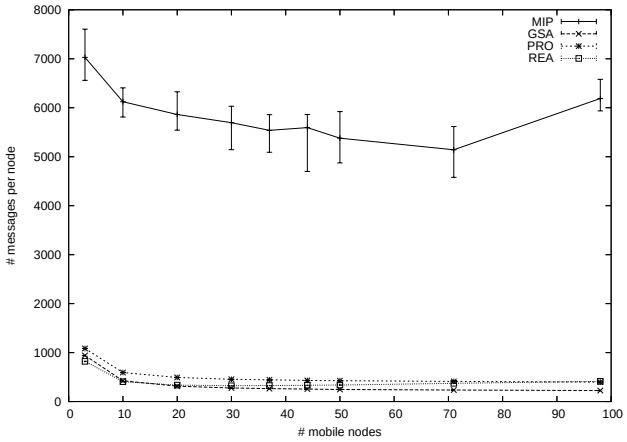


Fig. 13. Mean number of processed messages per node using MIP, GSA, proactive, and reactive approaches, excluding RG formation and routing protocol messages; error bars indicate min and max values

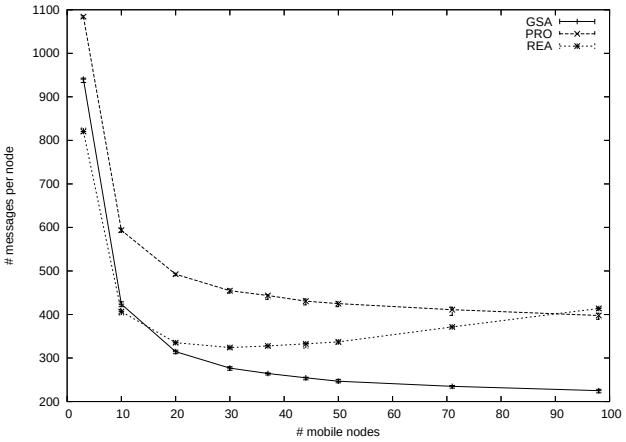


Fig. 14. Zoom of previous Figure 13

sent, forwarded, received and dropped packets handled above the MAC layer. As before, RG management and DSDV routing protocol messages were excluded, and so the average number of processed control messages is a good indicator of the total resource costs needed for gateway discovery and selection. MIP clearly underperforms the other three approaches. The standard deviation σ varies between 159 and 395 with MIP and 1.3 and 6.6 in all other cases.

The real gains when using GSA instead of proactive or reactive strategies are illustrated in Figure 14. First, GSA's hybrid signaling algorithm outperforms a proactive approach in all configurations. In fact the gains increase with n: for $n = 3$, on average, GSA nodes process 13.2% less control messages; with $n = 98$, they process 43.4% less messages. Second, GSA underperforms a reactive approach in small routing groups ($n \leq 15$). For $n = 3$, GSA nodes process, on average, 14.7% more messages (940.7 vs. 820.4). As n increases, GSA's relative performance against the reactive approach improves. With $n = 98$, GSA nodes need to process 50% less control messages than nodes using a reactive approach.

We note no other significant differences besides those mentioned above. Connection lost time between base stations was effectively the same in all scenarios, with small variations due to the locations of the nodes. The delay introduced by gateway discovery was not studied here because the gateway selection was triggered well before the GW service was actually needed. Nevertheless, we can say that GSA has a smaller (or equal) delay than reactive approaches, because the requesting node has to wait for only one response from the GWS; in reactive approach the node has to wait for a certain time in order to gather responses from all possible gateways and do the selection among those. Proactive approaches typically have very small delays—gateway selection occurs whenever needed among the saved advertisements.

VIII. Conclusion

We presented the Ambient Networks Gateway Selection Architecture, which manages the gateway discovery and selection mechanisms. We also showed how GSA can be used outside the Ambient Networks architecture, with Mobile IP and Host Identity Protocols.

We evaluated GSA with respect to sent and processed control messages in a moving commuter train scenario using simulation. The number of nodes in the train was varied between 3 and 98, which is quite large value for typical simulation studies, but on the other hand, also a representation of a real case on limit. We found that GSA has a considerable advantage over other alternatives. In particular, although GSA nodes transmit slightly more but unicasted control messages, as opposed to broadcasted control messages used in reactive and proactive strategies, GSA nodes need to process only 75% or less of the control messages processed using the alternative strategies, for medium-size routing groups. As an aside, we also verify the benefits from forming a routing group as opposed to having each node use Mobile IP independently. Our results show that GSA has more lightweight signaling than proactive or reactive approaches and that it scales much better as the routing group size grows.

Our future work includes the development of the gateway selection algorithm based on the parameters in gateway advertisements and requests. This includes also the definition of more detailed GSA extensions to ICMP Router Advertisement and Solicitation messages. With the selection algorithm we can study further the effects of selecting the most suitable gateway for the routing group nodes. The effects of the gateway discovery delay to service quality are also part of our future work.

Acknowledgment

Part of this work has been earlier published in [8]; the invitation to submit this extended version is highly appreciated. The simulation work was carried out in the framework of the Ambient Networks project, which was partially funded

TABLE I
LIST OF ACRONYMS

ACS	Ambient Control Space
ANI	Ambient Network Interface
AODV	Ad hoc On-Demand Distance Vector
ARI	Ambient Resource Interface
ASI	Ambient Service Interface
BS	Base Station
BSA	Base station Advertisement
CIB	Context Information Base
CN	Correspondent Node
CoA	Care-of Address
DA	Directory Agent
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic Source Routing
FA	Foreign Agent
GSA	Gateway Selection Architecture
GW	Gateway
GWA	Gateway Advertisement
GWR	Gateway Request
GWREG	Gateway Registration
GWRESP	Gateway Response
GWS	Gateway Selector
HA	Home Agent
HI	Host Identity
HIP	Host Identity Protocol
HIT	Host Identity Tag
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGW	Internet Gateway
IP	Internet Protocol
MIP	Mobile IP
MIPMANET	Mobile IP for Mobile Ad Hoc Networks
MN	Mobile Node
MOCCA	Mobile Communication Architecture
MR	Mobile Router
MRRM	Multi-Radio Resource Management
NEMO	Network Mobility
PRO	Proactive
QoS	Quality of Service
RAN	Radio Access Network
REA	Reactive
RG	Routing Group
RVS	Rendezvous Server
SA	Service Agent
SAP	Service Available Packet
SDP	Service Discovery Packet
SLP	Service Location Protocol
TCP	Transmission Control Protocol
TLV	Type, Length and Value
TRG	Triggering
TTL	Time To Live
UA	User Agent
UDP	User Datagram Protocol

by the Commission of the European Union. Writing of this work was supported by TEKES (Finnish Funding Agency for Technology and Innovation) as part of the Future Internet program of TIVIT (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT). The views expressed in this paper are solely those of the authors and do not necessarily represent the views of their employers, the Ambient Networks project, the Commission of the European Union, TEKES, or TIVIT. The comments and ideas from people involved in the Ambient Networks project's mobility research and from the anonymous reviewers are gratefully acknowledged.

REFERENCES

[1] N. Niebert (ed.), A. Schieder (co-ed.), J. Zander (co-ed.), and R. Hancock (co-ed.), *Ambient Networks: Co-operative Mobile Networking for the Wireless World*, Wiley, April 2007.

[2] N. Akhtar, C. Kappler, P. Schefczik, L. Tionardi, and D. Zhou, *Network Composition: A Framework for Dynamic Interworking between Networks*, Second International Conference on Communications and Networking in China (CHINACOM'07), August 2007.

[3] N. Niebert, M. Prytz, A. Schieder, L. Eggert, N. Papadoglou, F. Pittmann, and C. Prehofer, *Ambient Networks: A Framework for Future Wireless Internetworking*, IEEE VTC2005-Spring, June 2005.

[4] B. Ohlman, L. Eggert, M. Smirnov and M. Vorwerk, *The Ambient Networks Control Space Architecture*, 15^{th} World Wireless Research Forum, Paris, December 2005.

[5] R. Agüero Calvo, A. Surtees, J. Eisl, and M. Georgiades, *Mobility Management in Ambient Networks*, IEEE VTC2007-Spring, Dublin, Ireland, April 2007.

[6] A. Surtees, R. Agüero, J. Tenhunen, M. Rossi, and D. Hollos, *Routing Group Formation in Ambient Networks*, 14^{th} IST Mobile & Wireless Communications Summit, Dresden, Germany, June 2005.

[7] M. Eyrich, M. Majanen, E. Perera, R. Toenjes, R. Boreli, and T. Leinmueller, *GSA: An Architecture for Optimising Gateway Selection in Dynamic Routing Groups*, 16^{th} Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Berlin, September 2005.

[8] M. Majanen and K. Pentikousis, *An Evaluation of the Ambient Networks Gateway Selection Architecture*, Third International Conference on Wireless and Mobile Communications (ICWMC 2007), Guadeloupe, French Caribbean, March 2007.

[9] E. Guttman, C. Perkins, J. Veizades, and M. Day, *Service Location Protocol, Version 2*, RFC 2608, June 1999.

[10] C. Perkins (ed.), *IP Mobility Support for IPv4*, RFC 3344, August 2002.

[11] C. Perkins, *IP Encapsulation within IP*, RFC 2003, October 1996.

[12] D. Johnson, C. Perkins, and J. Arkko, *Mobility Support in IPv6*, RFC 3775, June 2004.

[13] Y. Sun, E. Belding-Royer, and C. Perkins, *Internet Connectivity for Ad hoc Mobile Networks*, International Journal of Wireless Information Networks, special issue on Mobile Ad Hoc Networks (MANETs): Standards, Research, Applications, 9(2), April 2002.

[14] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, July 2003.

[15] P. Ratanchandani and R. Kravets, *A hybrid approach to internet connectivity for mobile ad hoc networks*, Proceedings of WCNC 2003, Volume 3, pages 1522-1527, March 2003.

[16] U. Jönsson, F. Alriksson, T. Larsson, P. Johansson, and G.Q. Maguire Jr., *MIPMANET — Mobile IP for Mobile Ad Hoc Networks*, in 2000 First Annual Workshop on Mobile and Ad Hoc Networking and Computing, MobiHoc, 2000.

[17] J. Lee, D. Kim, J.J. Garcia-Luna-Aceves, Y. Choi, J. Choi, and S. Nam, *Hybrid gateway advertisement scheme for connecting mobile ad hoc networks to the internet*, Proceedings of VTC 2003, Volume 1, Pages 191-195, April 2003.

[18] D. Johnson, Y. Hu, and D. Maltz, *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*, RFC 4728, February 2007.

[19] M. Ghassemian, P. Hoffmann, C. Prehofer, V. Friderikos, and H. Aghvami, *Performance Analysis of Internet Gateway Discovery Protocols in Ad Hoc Networks*, WCNC 2004 — IEEE Wireless Communications and Networking Conference, Vol. 5, no. 1, March 2004.

[20] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, *Network Mobility (NEMO) Basic Support Protocol*, RFC 3963, January 2005.

[21] K. Leung, G. Dommety, V. Narayanan, and A. Petrescu, *Network Mobility (NEMO) Extensions for Mobile IPv4*, RFC 5177, April 2008.

[22] M. Bechler, W. Franz, and L. Wolf, *Mobile Internet Access in FleetNet*, 13. Fachtagung Kommunikation in verteilten Systemen, Leipzig, Germany, April 2003.

[23] R. Moskowitz and P. Nikander, *Host Identity Protocol (HIP) Architecture*, RFC 4423, May 2006.

[24] P. Nikander, T. Henderson, C. Vogt, and J. Arkko, *End-Host Mobility and Multihoming with the Host Identity Protocol*, RFC 5206, April 2008.

[25] J. Melen, J. Ylitalo, and P. Salmela, *Host Identity Protocol based Mobile Router (HIPMR)*, Internet-Draft, draft-melen-hip-mr, work in progress, July 2008.

[26] S. Nováczki, L. Bokor, and S. Imre, *A HIP based Network Mobility Protocol*, Proceedings of the 2007 International Symposium on Applications and the Internet Workshops (SAINTW'07), 2007.

[27] J. Mäkelä, R. Agüero, J. Tenhunen, V. Kyllönen, J. Choque, L. Munoz, *Paving the Way for Future Mobility Mechanisms: A Testbed for Mobility Triggering & Moving Network Support*, 2nd International IEEE/CreateNet Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (Tridentcom 2006), Barcelona, 2006.

[28] J. Mäkelä and K. Pentikousis, *Trigger Management Mechanisms*, Proc. of IEEE International Symposium on Wireless Pervasive Computing, San Juan, Puerto Rico, February 2007.

[29] R. Ocampo, L. Cheng, Z. Lai, and A. Galis, *ContextWare Support for Network and Service Composition and Self-Adaptation*, MATA 2005, Montreal, Canada, October 2005.

[30] R. Giaffreda, K. Pentikousis, E. Hepworth, R. Agüero, and A. Galis, *An information service infrastructure for Ambient Networks*, Proc. 25th International Conference on Parallel and Distributed Computing and Networks (PDCN), Innsbruck, Austria, February 2007, pp. 21–27. ACTA Press.

[31] F. Berggren, A. Bria, L. Badia, I. Karla, R. Litjens, P. Magnusson, F. Meago, H. Tang, and R. Veronesi, *Multi-Radio Resource Management for Ambient Networks*, 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Berlin, September 2005.

[32] J. Tenhunen, V. Typpö, and M. Jurvansuu, *Stability-Based Multi-hop Clustering Protocol*, 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Berlin, September 2005.

[33] N. Charkani, E. Hepworth, M. Johnsson, M. Cano, and J. Eisl, *Unbiased Approach to Ambient Control Space Migration*, Proc. First Ambient Networks Workshop on Mobility, Multiaccess, and Network Management (M2NM), Sydney, Australia, October 2007.

[34] S. Deering (ed.), *ICMP Router Discovery Messages*, RFC 1256, September 1991.

[35] R. Moskowitz, P. Nikander, P. Jokela (ed.), and T. Henderson, *Host Identity Protocol*, RFC 5201, April 2008.

[36] J. Laganier and L. Eggert, *Host Identity Protocol (HIP) Rendezvous Extension*, RFC 5204, April 2008.

[37] J. Laganier, T. Koponen, and L. Eggert, *Host Identity Protocol (HIP) Registration Extension*, RFC 5203, April 2008.

[38] P. Jokela, J. Melen, and J. Ylitalo, *HIP Service Discovery*, Internet-Draft, draft-jokela-hip-service-discovery, work in progress, June 2006.

[39] S. McCanne and S. Floyd. ns Network Simulator. http://www.isi.edu/nsnam/ns/.

[40] E. Hernandez and A. Helal, *Examining Mobile-IP Performance in Rapidly Mobile Environments: The Case of a Commuter Train*, LCN 2001, Tampa, FL, November 2001.

Long-Range CAN: to Enhance the Performance of Content-Addressable Networks

Balázs Kovács, Rolland Vida
Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics
Magyar tudosok krt. 2, Budapest, Hungary 1117
{kovacsb, vida}@tmit.bme.hu

Abstract

Distributed Hash Table (DHT) algorithms structure peer-to-peer networks to provide nodes with fast and scalable lookups. In recent DHT solutions, such as Chord and Kademlia, the contacts of a node in the overlay network are determined so as to keep up with a lookup cost of $O(logN)$ in a network of N nodes. As opposed to these, one of the first DHT solutions, called Content Addressable Network (CAN), has the drawback of limiting the lookup cost only in $O(dN^{\frac{1}{d}})$ where d is the number of dimensions in the coordinate space, a fixed network parameter. However, CAN has several merits to exploit, such as its multi-dimensional ID space and its special ID space structure. Thus, in this paper we present an improved algorithm called Long-Range CAN (LR-CAN), able to eliminate the rigidity of the original system and to provide a more scalable and resilient solution, not only compared to the original version, but also to the currently best performing DHTs that we already mentioned.

Keywords: Distributed Hash Table, Content-Addressable Network, small world, lookup-cost limitation, signaling optimization

1. Introduction

Distributed hash table algorithms construct structured P2P networks to control the communication cost of resource lookups. Resource or content lookups in DHT protocols are performed based on a key that is a hash print of the searched content. Nodes and keys are mapped on an identifier (ID) space, and the distance between the key and the node performing the lookup is determined. Usually, the protocols employ a greedy forwarding mechanism to forward the lookup to the owner of the key.

During this greedy forwarding, a node thus scans its contacts to find the closest node to the key. Generally, contacts can be classified into two categories: short-range and long-range contacts. The former category consists of contacts very close to the node in the ID space; they are indispensable for a node to properly participate in a DHT. If some, or all of the short-range contacts fail, a node may be unable to forward lookups. As opposed to these, long-range contacts are not mandatory for nodes to survive; however, they are useful to accelerate the lookups so as to be of a length proportional to the logarithm of the network size. DHT protocols define, set up, and maintain their short- and long-range contacts differently.

One of the first published DHTs was CAN [1]. It maps nodes and keys onto a d-dimensional ID space that wraps to a d-torus. The ID space is split into d-dimensional zones which are assigned to the nodes. A node is responsible for keys which map into its zone. The algorithm defines only short-range contacts for a node, which are its immediate neighbors on the torus. This concept significantly reduces the cost of maintaining contacts and thus signaling, as the number of contacts remains $O(d)$. However, lookup lengths may grow long and present large variations. An answer to this problem might be to increase the number of dimensions of the torus. As a consequence, the number of short-range contacts increases, the ID space is distributed along more dimensions, and thus hop distances decrease. The number of dimensions d can be chosen so as to make lookup lengths proportional to $logN$; however, d is a fixed system-wide parameter, and if one would like to change it on the fly, costly algorithms would be needed to re-hash and reshape the network accordingly. As a result, lookup lengths remain $O(dN^{1/d})$ [1]. If N increases significantly during the lifetime of the system, and if d cannot be modified accordingly, lookup lengths will be notably longer than with an $O(logN)$ system. Another solution to reduce lookup cost is the use of "realities". In each reality, the ID space is distributed randomly; thus, realities assign different zones for nodes and

introduce redundancy in the system, as nodes will probably store different keys in different realities. This method also reduces lookup cost, as lookups can jump between realities. However, realities are less effective in decreasing lookup lengths than the use of an increased number of dimensions.

Chord [2] maps nodes onto a ring as ID space. Short-range contacts of Chord are called successors and predecessors. Besides these, Chord also uses long-range contacts, called fingers, to keep lookup lengths $O(logN)$. In case of m-bit long identifiers, maximum m fingers are maintained, placed at exponentially increasing distances from the node. If we have a look at the basics of the system, Chord and CAN are very similar. A one dimensional CAN is an identifier circle, just as Chord. The main difference is that Chord has a different join and zone assignment strategy than CAN, and it has fingers, while CAN may have multiple dimensions. Unfortunately, the main drawback of Chord is the one-dimensional ID space and the unidirectional circle. Bidirectional Chord [3] eliminates the problem of unidirectionality, but increases the maintenance load of the network to keep fingers precise, although the bidirectional ring makes the system less sensitive to the imprecision of fingers.

Kademlia [4], a system developed from the basics of Pastry [5], is different in many ways. First, it defines a new distance metric, called XOR distance. The ID distance of two nodes is the XOR value of their identifiers. The XOR distance is unidirectional. In order to find short- and long-range contacts, each node examines the senders of messages which flow in the system upon joins and lookups, and decides whether to record a sender as a contact or not. In case of m-bit IDs, a Kademlia node n stores m buckets. Kademlia only defines ID ranges per buckets which should be covered by some nodes in contrast to the deterministic approach of Chord. To enhance routing performance, at most k nodes can be recorded to each bucket. A node stores a key if it cannot forward it to any node closer to the key. The parameter k also introduces redundancy by storing keys at multiple nodes. Kademlia has an iterative lookup strategy that always returns lookups to the initiator after visiting a new hop, until the destination is found. Compared to a recursive strategy it costs more messages, but may provide auxiliary information that may improve the lookup protocol, which is necessary in Kademlia as it learns the topology from lookup traffic.

Overall, most DHTs achieve route lengths proportional to $O(logN)$ if, in some way, they use long-range connections. In 1999 Jon Kleinberg worked out a network model for the small-world phenomenon which sets up requirements for decentralized systems to be able to find the "short paths" in the network. Based on the model of Watts and Strogatz [6], which proposes to have many short-range and a few long-range connections, Kleinberg determined a stochastic model of choosing long-range contacts in a network of arbitrary dimensions [7]. He stated that should the nodes of a 2-dimensional network follow his distribution of placing long-range connections (only one per node), the number of hops will be at most $O((logN)^2)$. Since most of the DHTs suit the Kleinberg model and have stricter rules of placing long-range connections, they obviously provide better path lengths in average than the one mentioned by Kleinberg.

Among the well-known DHT solutions CAN is the most similar to Kleinberg's general model as CAN defines operations in multiple dimensions. Its only shortcoming is not using long-range connections; thus, the variance of lookups yields a high value. An earlier paper presented an algorithm which enhanced CAN to be compliant with Kleinberg's methods, as it installed one "shortcut" per node according to Kleinberg's formula [8]. The authors achieved significant improvements in lookup lengths when compared to the original CAN; however, the approach is still not competitive with the performance of Chord or Kademlia.

The significance of the results presented in this paper is twofold. First, we present Long-Range CAN (LR-CAN), an algorithm that enhances CAN by utilizing long-range contacts, a solution very popular for scalable DHTs. Second, the algorithm introduces adaptivity to network size through a cost-limit function (denoted as $SR(N)$ further on). In contemporary DHTs the number of long-range contacts implicitly changes as network size changes, and their number is proportional to the logarithm of network size. On the contrary, LR-CAN controls the number of long-range contacts explicitly through the above mentioned definite cost-limit function. With the appropriate cost-limit function we can set LR-CAN so as to outperform Chord and Kademlia. Nevertheless, its multi-dimensional ID space, bidirectional along each dimension, and its special node mapping algorithm allow LR-CAN to keep the necessary maintenance traffic lower than in the case of the above mentioned state-of-the-art DHT protocols.

The rest of the paper is organized as follows. Section 3 presents the adaptive algorithm we propose (LR-CAN), while section 4 presents our theoretical expectations concerning its efficiency. Section 5 describes our simulation technique and presents the results by comparing them to the theoretical expectations and to the efficiency of alternative DHT solutions. Then, section 6 presents a method to minimize the signaling of LR-CAN. Finally, in section 7 we conclude the paper.

2. CAN Overview

As mentioned already in the introduction, in CAN each node is responsible for the resources which map into its zone. When joining the system, the new node draws a point

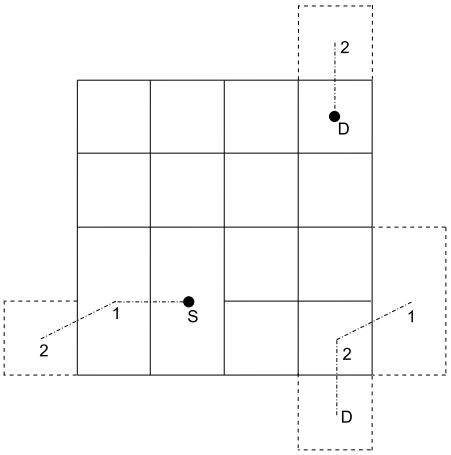


Figure 1. Routing on a 2-torus

P in the d-dimensional coordinate space according to a uniform distribution. The lookup for P is initiated by a gateway node. The node responsible for P will be the host for the newcomer; it will split its zone in two, based on the space splitting rules, it will notify the newcomer about its neighbors, and its neighbors about the newcomer. According to the space splitting rules, always the longer edge needs to be halved, but if more edges are of equal lengths, the edge with the lower-order dimension has to be split. Two nodes are neighbors if the coordinate spans of their zones overlap along $d-1$ dimensions and abut along one dimension. If the ID space is distributed uniformly to zones exactly with the same size, then each node will have $2d$ neighbors.

When a lookup is initiated, key K is deterministically mapped to a d-dimensional point P. Forwarding can happen only between neighbors. Each node checks its Euclidean distance to P and forwards the lookup to its neighbor closest to P in a greedy way. Note that CAN is bidirectional along each dimensions and the ID space wraps. For instance, the distance on a 1-torus between point 0.1 and 0.9 is 0.2 rather than 0.8. Fig. 1 shows a routing scenario in two dimensions on a 2-torus, where a lookup goes from S to D.

In case of a graceful leave, i.e., when a node is able to hand over his tasks before leaving, the leaving node has to choose one of its immediate zone neighbors to merge with its zone. If merging zones is unfeasible in the current state, since the zones would compose an invalid concave zone that contradicts the space splitting rules, the neighbor with the smallest zone will take over; thus, this node will temporarily own two zones, until assigning the zone to a new node or merging it with a third zone. Node failures are handled differently, as in this case the peers sensing the absence of another peer have to arrive to a consensus about the node that takes over the zone of the failed peer. We do not present the way of failure handling because it does not affect our algorithm. For more details about the operation of the original CAN algorithms, please refer to [1] and [9].

3. The LR-CAN Algorithm

LR-CAN uses long-range contacts to reduce lookup cost. The main difference between LR-CAN and other DHTs that employ long-range contacts is that the number of long-range contacts of a node changes adaptively according to a desired lookup cost, expressed by a cost-limit function, as defined in Section 4. In other DHTs the number of long-range contacts is an implicit and not controllable function of network size; thus, the algorithm and its current parameter set up implicitly determines the achievable lookup performance of the DHT.

The main idea behind the LR-CAN algorithm is that a node is able to assess the network size (number of nodes) individually, without the help of an information server. The reason for that is the distribution of the ID space that converges to a distribution with equal zones; this is because on a stochastic basis always the largest zones are split when nodes join the network and draw a random d-dimensional coordinate to determine their host node. In fact, the algorithmic principle behind this feature is that the node ID and the zone assigned to the node in the CAN ID space are independent. In Fig. 2, we can see the difference between the ID space assignment strategy of CAN and Chord. We insert three nodes in the same sequence for both systems. Let us assume that the hash of the IP address in Chord equals the random P coordinate the same node draws in CAN. On the left hand side we can see how a one-dimensional CAN assigns portions of the ID space to these three node. In CAN, the sequence of nodes joining the system affects the mapping. In this example, first, node A owned the whole ID space before node B joined. When B joined, the ID space was split into two equal parts, and the zone closer to the origin was preserved for node A. The same happens when node C draws the point mapping into the zone of node A. In contrast to CAN, in Chord, the sequence of nodes does not count, only their ID; they get the portion of the ID space where they are successors. As a result, the sizes of ID space portions in Chord depend on the hash implementation and the nodes joining the system, whereas in CAN space portions are deterministically balanced.

The aim is to limit the problematic original routing scheme of CAN with a cost-limit function, denoted as $SR(N)$. This function upper bounds the average lookup cost and triggers LR-CAN to increase the number of long-range contacts in the network. Obviously, the cost-limit is a function of the network size N. Since defining their own long-range contacts is the individual responsibility of the overlay nodes, and N varies, the nodes have to be able to

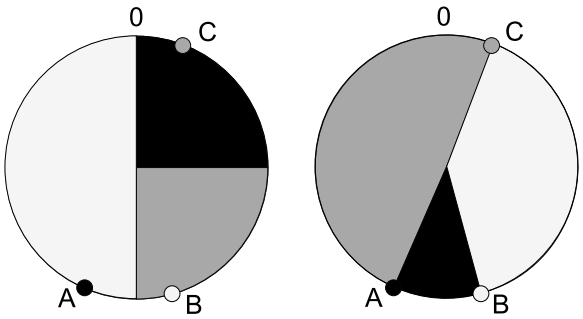


Figure 2. CAN and Chord ID space assignment strategy

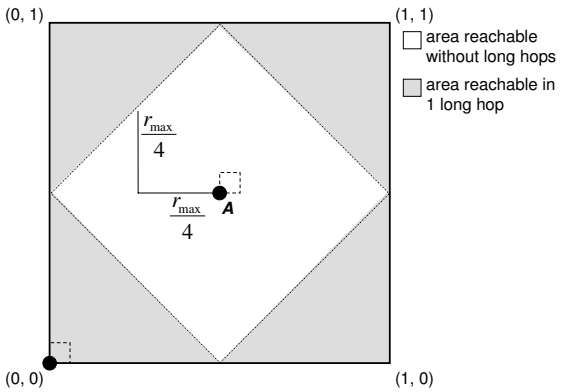


Figure 3. One long-range level is active

assess the network size N. Based on the assumption that the ID space distribution converges to equal zones, network size ($\widehat{N}$) can be assessed in several ways. One technique could be to calculate the sum of the zone sizes of immediate neighbors, and thus infer to network population. Low cost is the primal advantage of this solution as a CAN node always stores information about the zones of its neighbors. The drawback is that the result will reflect a local view of the CAN network; even if we assume a convergence to nearly equal zones, there will be local deviations that may provide inaccurate information about network size. A more accurate solution is by routing to the most distant point in the ID space and measuring the hop distance ($\widehat{R}_{max}$). We can estimate N through a formula that describes the connection between $\widehat{R}_{max}$ and N; just like the first technique, this solution also assumes a CAN ID space with equal zones. As this approach traverses a more significant portion of the ID space, it can provide more accurate information about N than sampling the zone sizes of immediate neighbors. Unfortunately, this second technique has higher cost, because of the necessity to route to the most "distant" point; however, with a careful design this cost remains proportional to $O(logN)$ (explained later in Section 4).

As the nodes are aware of the position of their zones, they can easily determine the most distant point in d dimensions. To do so, we made a simplification and used the bottom-left corners of a zone as reference points to define zone-to-point distances. The bottom-left corner is a good choice since the zone splitting rules of CAN keep this point always belonging to the original owner of the zone, no matter how many times the zone is split because of joins. Note that routing in CAN can only progress horizontally or vertically passing through neighbors, and thus the most distant point is determined accordingly.

Fig. 3 shows a two-dimensional ID space; for the sake of simplicity, first we explain the LR-CAN algorithm for $d = 2$. For node *A*, located in the middle of the unit square, the most distant points are located in the corners of the ID space, which correspond to one specific point, (0,0), as all the four corners of the square represent the same point on the d-torus. The Euclidean distance between them is $r_{max} = 2 \cdot \frac{1}{2}$, as we need to progress through half of the full length of the first dimension horizontally then again half of the full length of the second dimension vertically. To cover that distance, we will thus need

$$\widehat{R}_{max} = r_{max} \cdot N^{\frac{1}{2}} \quad (1)$$

hops, as along each dimension there are approximately $N^{\frac{1}{2}}$ nodes dividing the coordinate space among them [1]. If *A* initiates a lookup to (0,0), then it can measure the hop distance to the most distant point by a counter in the packet; based on the result and by transforming Equation 1, it can assess N ($\widehat{N}$). Then, we can calculate the value of $SR(\widehat{N})$, which is supposed to be the upper bound for the average route length. If the average lookup cost ($\widehat{R}_{avg}$) that can be calculated from the measured $\widehat{R}_{max}$ (explained in section 4) is higher than the value that the cost-limit function ($SR(N)$) allows, a new level of long-range neighbors is deployed; by doing so, we reduce the size of the ID space where lookups use traditional CAN routing, and we add the first level of long-range contacts, where long-range routing takes over ($L = 0$). Certainly, the distant point we measured our distance to belongs to a zone for which there is a responsible node. This node will be added as a long-range contact, and will be used as any other short-range neighbor in greedy forwarding. Network nodes behave consistently; thus, their individual decisions will be valid in a global scope, improving the performance of the whole network.

If all nodes assessed N and decided to set up their first long-range contact to the most-distant point compared to their own position, the traditional CAN routing will have to be employed only over half of the original ID space for every node; in this reduced space, the most distant points from the nodes will be half as far as the point where a long-range contact points already. For node *A*, the most distant

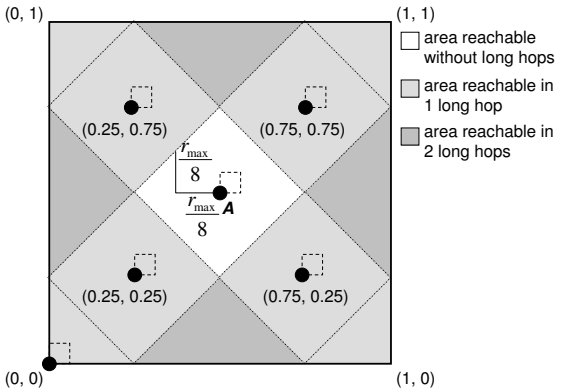


Figure 4. Two long-range levels are active

points (measured in hops) lie on the edges of the square spanned by points $(0.5, 0)$, $(1, 0.5)$, $(0.5, 1)$, and $(0, 0.5)$ (the white square in Fig. 3). If the system intends to further reduce the maximum distances, it needs to distribute the rest of the ID space evenly to have a general improvement that holds for the whole ID space. Among the most distant points, points $(0.25, 0.25)$, $(0.25, 0.75)$, $(0.75, 0.75)$ and $(0.75, 0.25)$ are the most appropriate choices for this goal, as shown in Fig. 4. These points are the next level long-range contact candidates for *A*. In case the hop distance to these new points exceeds $SR(\widehat{N})$ again—which is measured periodically—a new level of long-range contacts is introduced ($L = 1$). In this case we will initiate four new long-range connections; as we do not talk about the entire ID space anymore, these candidate points are not collocated on the torus. By introducing these new long-range contacts, the ID space where the original CAN routing is employed will be reduced again. The whole procedure is repeated until the average hop distance of the traditional CAN routing does not exceed the route length limit dictated by $SR(\widehat{N})$; hence, the LR-CAN algorithm endeavors to maintain the following condition:

$$\widehat{R}_{avg} = \widehat{R}_{max} \cdot m_{\frac{avg}{max}} < SR(\widehat{N}), \tag{2}$$

where $m_{\frac{avg}{max}}$ is a multiplication factor, an empirical ratio of R_{avg} and R_{max} that can be estimated accurately on-line based on the current number of long-range levels and $\widehat{R}_{max}$.

From now on we will call the traditional CAN routing as *short-range routing* because it uses only short-range contacts in forwarding; accordingly, we call *long-range routing* the case when long-range contacts forward a lookup. The routing of LR-CAN can be split in two phases: in the first one a certain key is approached by long-range routing; in the second phase short-range routing leads to the owner of the key. In Fig. 3 and Fig. 4 we can see how the deployment of long-range contacts affects LR-CAN routing. The white square shows the subspace where only short-range routing is employed, in case node *A* starts a query. If the query is initiated to some distant part of the ID space, *A* uses its long-range contacts first, and only afterwards the traditional CAN routing, based only on short-range contacts; thus, the scope of traditional CAN routing is reduced. Certainly, the $SR(N)$ cost-limit function limits the route lengths only partially since the addition of new long-range levels will generate a long-range routing cost.

An LR-CAN node needs to maintain its long-range contacts to review the assignment of coordinate points to nodes. To do so, it pings its contacts periodically and directly. If one of them fails, in the next maintenance period the node initiates a lookup for the corresponding coordinate point again, in order to find out which node is currently responsible for that point. The join procedure of CAN also needs to be modified. To spare the cost of setting up long-range levels, the newcomer can learn the current number of levels from the node through which it joins the network. Nodes also need to periodically route a message to one of the most distant points of the ID space, in order to measure whether a new long-range level has to be installed or not; this is done together with contact maintenance. Since long-range contacts are not necessarily symmetric, a node's long range contacts do not have to be notified neither when the node joins, nor when it leaves the network. As the bottom-left corner of a zone is invariant while a node is online (as explained by Fig. 2), if a coordinate of a long-range point hits the bottom-left corner of the given long-range contact, the point-to-node assignments do not change until the node leaves the network.

4. LR-CAN Cost Analysis

If we assume that routing can progress only horizontally or vertically, the maximum distance in one dimension between two points $\mathbf{p}$ and $\mathbf{q}$ in the ID space, ($\mathbf{p}, \mathbf{q} \in \{\mathbb{R}^d | [0, 1)\}$) can be $\frac{1}{2}$. In d dimensions this maximum distance is $r_{max} = \frac{d}{2}$ (see Fig. 3). In order to obtain the distance in hops, the maximum distance has to be multiplied with the "resolution" ($N^{\frac{1}{d}}$) of the ID space. Therefore, in the original CAN solution the maximum route length in hops is

$$R_{max} = \left\lceil r_{max} \cdot N^{\frac{1}{d}} \right\rceil \tag{3}$$

if we have N nodes in the network [1]. Each new level of long-range contacts halves the maximum distance. Hence the average route length of short-range routing (not including jumps on long-range contacts) is:

$$R_{avg} = \frac{r_{max}}{2^{L+1}} \cdot N^{\frac{1}{d}} \cdot m_{\frac{avg}{max}}. \tag{4}$$

Through simulations we observed that in general $m_{\frac{avg}{max}} \sim 1.4$ when $L \geq 0$, and ~ 2 if no long-range contacts are used

(this latter observation is also present in the original CAN paper [1]).

We want LR-CAN to provide $O(logN)$ lookup cost similarly to other DHT solutions; as a consequence, we propose to use $SR(N) = \frac{1}{c} \cdot log_2 N$ as the upper bound of short-range routing, where c is the cost-limit factor, an arbitrary positive real number. To express lookup cost, we need to deduce how L depends on N. Substituting R_{avg} by $\frac{1}{c} \cdot log_2 N$ in equation 4 yields to the following:

$$\begin{aligned} L &= \log_2 \frac{r_{max} N^{\frac{1}{d}} m_{\frac{avg}{max}} c}{2 \log_2 N} \\ &= \log_2 r_{max} N^{\frac{1}{d}} m_{\frac{avg}{max}} c - \log_2 2 \log_2 N \\ &= O(\log N) - O(\log \log N) = O(\log N) \end{aligned} \quad (5)$$

as d can be considered as constant in our algorithm. The average message cost of lookups can be expressed by the sum of the long-range and short-range routing cost:

$$LR_{avg} + R_{avg} = O(L) + O(\log N) = O(\log N) \quad (6)$$

As mentioned earlier in section 3, probing if R_{max} is over the cost-limit function requires to route a probe message to one of the current most distant points (depending on the number of activated levels) in the ID space by using original short-range routing. Generally, if LR-CAN keeps R_{avg} proportional to $O(logN)$ then R_{max} is also proportional to it. However, in a pathological case when the size of the network grows with orders of magnitude between two long-range maintenance periods, determining R_{max} will cost $O(dN^{\frac{1}{d}}/2^L)$.

As a result, the introduction of L eliminates the significance of d in the lookup cost of LR-CAN. Although in the traditional CAN architecture, as we mentioned earlier, d can be set so as to provide low lookup cost, the signaling load will increase much more than in case of LR-CAN with multiple levels of long-range neighbors. Moreover, d is a fixed network parameter the change of which needs costly rehashing of the whole ID space. It is important to note that the ability to build a multi-dimensional space still has significant benefits, since it vests the system with better fault-tolerance (more short-range contacts); moreover, the imprecision of long-range contacts is less detrimental for lookup performance than in a single-dimensional ID space.

The reduced lookup cost of LR-CAN comes at the price of having more contacts to maintain than in CAN:

$$\begin{aligned} LC_{avg} + SC_{avg} &= 2d + 1 + L2^d \\ &= O(d) + O(L2^d) \\ &= O(L) = O(\log N) \end{aligned} \quad (7)$$

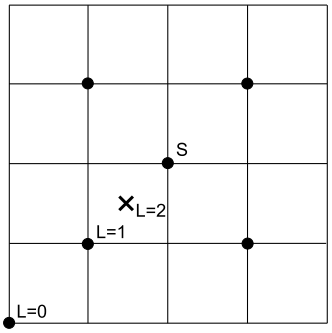


Figure 5. A level-two long-range coordinate maps to the same node as a level-one

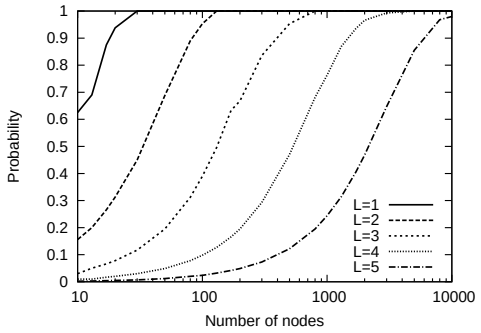


Figure 6. Probability of mapping all long-range coordinates to different nodes

as for level $L = 0$ there is 1 long-range contact, and for each level $L > 0$ there are another 2^d contacts.

Certainly, L has an implicit upper bound; its margin depends on the "resolution" of the ID space. There is no point in defining new long-range coordinates if they are so close to the given node that they map to the same nodes; this case can be seen in Fig. 5: the bottom-left level 2 contact of node A will map to the same node that holds its level 1 contact. Consequently, L should be defined on the $(-1 \leq L \leq \lfloor \log_2 N^{\frac{1}{d}}/2 \rfloor, L \in \mathbb{Z})$ domain; -1 indicates the case when no long-range contacts are used. Obviously, this constraint will be different in real life since the mathematical formula presented here assumes an equally distributed ID space. In Fig. 6, we can see the probability that all long-range coordinates of a given level map to different long-range contacts in function of network size. These probabilities were determined by simulation that used a uniform random number generator to draw the "join coordinates" of CAN nodes. For instance, for 256 nodes the theoretical upper bound yields maximum three levels, while we can be sure about mapping all contacts on the three levels to different nodes with a probability of around ~ 0.88, ac-

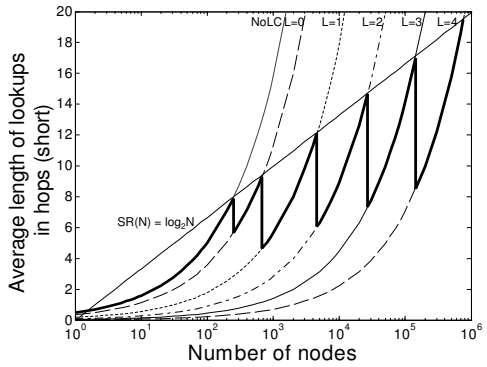


Figure 7. Analytical results on the growth of $\mathbf{R_{avg}}$ in function of $\mathbf{N}$

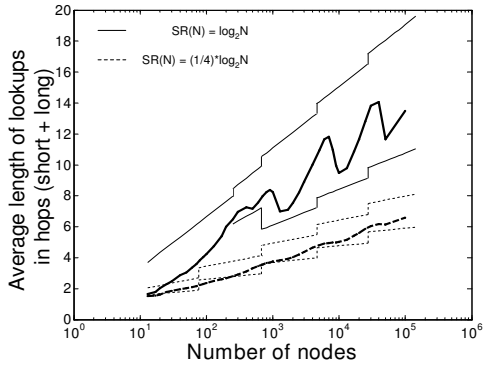


Figure 8. Simulation results on the average route length for different SR functions

cording to the simulation. As we learned, the desired cost-limit function is expressed through the c parameter, but the domain of L implicitly defines also the reasonable domain of c, i.e., there is no point in defining a too high c value. Nevertheless, the protocol presented in Section 3 can safely handle any arbitrarily high c and set up long-range contacts only when beneficial.

5. LR-CAN Performance Simulation

We simulated LR-CAN in a P2P overlay simulator [10, 11] developed in cooperation with the Technical University Darmstadt. The simulator can be used for testing several well known P2P search/lookup algorithms. Currently, Chord, Kademlia, CAN, and Gnutella [12] are all implemented in the simulator. We used the simulator to implement LR-CAN, validate our mathematical formulas in a simulated environment, and to compare the routing performance of CAN, LR-CAN, Chord, and Kademlia.

5.1. Validation of Theoretical Results

In Fig. 7, we can see how short-range route lengths grow in function of network size, in a two-dimensional CAN space, according to the above presented analytical model. A new level is switched on when the $f(N) = dN^{\frac{1}{d}}$ curves cross the desired average of short-range routing, the cost-limit function, namely $SR(N) = \frac{1}{c} \cdot log_2 N$. In Fig. 7, $c = 1$. When $SR(N)$ is reached, route length drops, while the number of neighbors increases. The number of levels corresponding to the different curves is denoted by L. The average, theoretical route length of the short-range routing grows along the thick line.

The mathematical analysis contains assumptions that cannot be fully met in simulations and real life. For the simulation results, we separately constructed LR-CAN networks with several different network sizes, and performed a significant amount of random key lookups to obtain a representative average for lookup message cost. In Fig. 8, we can see how LR-CAN with different c parameters reacts on a growing network. We can see the simulated results on the thick curves with two different values of c ($c_1 = 1$ and $c_2 = 4$) to define two different cost-limit functions. To compare the simulated results to the analytical ones, the thin lines represent a lower and an upper bound for the simulation results, which enclose the respective thick curve. The jumps in the thin curves are due to the long-range routing cost (LR_{avg}) that appears when a new long-range level is introduced; this routing cost depends only on the value of L and d (see Section 6). The theoretical thin curves assume equal-size, 2-dimensional zones and the same L value for all nodes, assumptions which obviously do not hold in real setups.

5.2. LR-CAN Comparison with Chord and Kademlia

Having multiple dimensions is a solution to speed up lookups. Linear search in a one-dimensional space generates $O(N)$ lookup lengths, compared to the $O(dN^{1/d})$ lengths that CAN provides in a d-dimensional space; higher d values yield lower lookup cost. Consequently, when a multi-dimensional solution, such as LR-CAN, has to provide a determined lookup cost, it may have less strict requirements for the precision of long-range contacts than in single-dimension solutions, such as Chord and Kademlia. As a result, LR-CAN may generate less signaling for the maintenance of long-range contacts.

Another DHT solution, Chord, is similar to LR-CAN except for three essential features. First, Chord uses a one-dimensional ID space, which requires to pay more attention to the precision of long-range contacts in order to keep

lookups fast. Second, the ID space is unidirectional, so less alternative routes exist in the system (this drawback was overcome by [3]). And third, the distribution of the ID space is more heavily based on randomness, i.e., the position and hash zone of a node on the ring basically depends on the ID of the node. As opposed to this, CAN splits the ID space into equal parts as much as possible, according to its zone splitting rules operating regardless of the node IDs. A CAN ID space can be easily mapped to a nearly balanced binary tree. Thanks to these features, the peers participating in an LR-CAN network can estimate individually the actual global average of lookup cost, and decide about the introduction of new levels of long-range neighbors accordingly. This is a major advantage of LR-CAN.

Kademlia nodes cover each part of the ID space with k randomly selected neighbors stored in a so called "k-bucket." Kademlia significantly differs from CAN and Chord as it is an on-demand, reactive solution, while the latter ones are proactive. The on-demand nature origins from the contact maintenance procedure. While CAN and Chord are proactive in short-range contact maintenance, Kademlia learns topology changes from its own lookup traffic. If there is no traffic at all, or just moderate one, Kademlia may have imprecise information about the network topology, and thus, it may fail to answer certain lookups or move key-value pairs to the appropriate nodes. As a consequence, Kademlia needs to make some effort to spread the topology information better. To do so, Kademlia nodes generate dummy lookups periodically, and employ an iterative lookup strategy. This means that a lookup message is always returned to its initiator after every newly visited hop, until the destination is found; this results in higher costs for the lookups. Another way to better disseminate topology information is to increase the k parameter. In this case, a node learns about more nodes in the network, and thus knows the topology better; resources are replicated and stored on more nodes, so they are "easier" to find.

In our simulations we focused on performing a *fair comparison* of these DHTs. The scope of the simulations was to investigate how dynamism affects the message cost of lookups, which is basically influenced by the precision of long-range contacts. Since the effect of failures and graceful leaving of nodes are similar in terms of long-range contact maintenance, we did not implement failure handling mechanisms in Chord and LR-CAN. As the on-demand nature of Kademlia enables to handle failures without any additional mechanisms, the comparison in Fig. 9 is only partly relevant for Kademlia because, as mentioned, Chord and LR-CAN lack the corresponding mechanisms. The investigation of failure-tolerance can be another important topic of DHT-related research [13]. The elimination of failure handling means that failure detection and its corresponding mechanisms are not implemented; nodes always "clean up" after they leave the system, i.e., short-range contacts remain correct and precise, and stored key-value pairs are moved to the node that takes over the responsibility of handling the leaving node's zone. By these means, we endeavored to simplify the maze that this comparison with its multi-dimensional problem space frames. We can clearly focus on how the precision of long-range contacts and the stabilization of the DHTs maintaining these contacts affect average lookup costs and the signaling DHTs generate.

Parameters. One of the most influencing parameters is the length of the *stabilization period* DHTs have. All of these three DHTs operate some procedure with a common aim: to set the precision of long-range contacts. The name "stabilization" origins from the procedure of Chord that involves into stabilization a so called "fix-fingers" procedure. As described in Section 3, LR-CAN also reviews its long-range contacts periodically. In Kademlia, a similar periodic task is fulfilled by bucket refreshes. In order to measure how tolerant the algorithm and the overlay structure of these DHTs are for the imprecision of long-range contacts, we varied the stabilization period on a wide range.

LR-CAN and Kademlia have other system parameters that highly affect their performance. As mentioned earlier, the number of dimensions is still an optional parameter that can vest LR-CAN with better failure tolerance. Additionally, dimensions affect when and how long-range contacts need to be deployed. The effect of c has already been deeply discussed; it represents the connection between network size and average lookup cost. Kademlia also has two important parameters: the value k has influence on how widely routing information is disseminated and how redundant the system becomes. Furthermore, parameter α is a concurrency parameter defining the number of asynchronous parallel queries a node can start on a lookup request. We omitted the investigation of α in this comparison as this type of concurrency primarily exploits the underlying link characteristics and could be implemented both in Chord and LR-CAN.

Metrics. A usual way to evaluate the performance of different DHTs is to compare the average number of logical hops needed to find the owner of a key. However, there are other metrics that might come also into focus when logical hops are compared. First, there is a difference between recursive and iterative protocols, as they approach a target differently. A lookup flowing through the same number of nodes means twice as many messages for an iterative protocol (e.g., Kademlia), as the initiator controls the lookup process hop-by-hop. As a result, the *message cost of a lookup* is a better metric than the number of logical hops. Note that in the measurements related to message cost we counted the number of messages needed to reach the owner of a key; the last message, used for returning the searched value, was not added. The amount of *signaling traffic* together with

the achievable message cost of lookups determine the overall lookup performance. Low message cost is worth nothing if its provision requires relatively high signaling traffic. In our signaling traffic measurements we included all the traffic generated during node joins and leaves, the stabilization of long-range contacts, and the lookup traffic. If the intention is to compare the results with Kademlia as well, lookup traffic also needs to be involved into signaling, since Kademlia uses its lookup traffic to maintain its long-range contacts; thus, contact maintenance cannot be differentiated from lookup traffic as in Chord and LR-CAN.

There are also other metrics that might be interesting, but we omitted to present them as they either seemed redundant with the above mentioned metrics, or were non-informative in our simulations. One of these metrics was the *number of contacts* that is only loosely proportional to signaling traffic. For instance, Kademlia may store several times more contacts than Chord or LR-CAN but it generates far less signaling messages. Average *latency of lookups* was a redundant metric in our simulations as non of the implemented DHTs was optimized for link latency; thus, on long-run average the message cost of lookups was directly proportional to the average lookup latency. Nevertheless, the *success ratio of lookups*, which reflects the ratio of the number of successful lookups divided by the number of lookups for a stored key, is an important metric in failure scenarios. However, as already mentioned, in our simulations nodes always cleaned up after leaving; hence, in a proper implementation of Chord and LR-CAN the success ratio of lookups remains 100%. In Kademlia this is not so straightforward: for low k values or rare stabilization steps a node trying to push its key-value pairs to the appropriate nodes before leaving may not be properly aware of the nodes closest to the given keys. This may result in loosing some of its keys and thus the network gradually looses a portion of the key-value pairs it should have saved. This phenomenon can be effectively reduced with increasing k or the stabilization period, which will implicitly also appear in some of the message cost figures we present.

Scenario. As the aim of the simulation was to test how the imprecision of long-range contacts influences routing in the overlays of the different DHT algorithms, we labored a scenario where we replace a significant portion of the participating nodes several times to generate stale or imprecise long-range entries. For the sake of fair comparison between different phases of the simulation run, we split the simulation time up to five phases that are equal in time (1470 sec). In the first phase, 10000 nodes gradually join the network and initiate the storage of some random key-value pairs. These contents are also stored in a globally accessible memory in the simulator so that later the nodes can perform lookups only for content that already had been stored and that should have been preserved by the DHT during the simulation. Note that in the simulator implementation a key stored in the DHTs is not removed when its original owner leaves the network; hence, in an ideal case, DHTs should preserve all the keys that were stored sometime during simulation. Performing lookups for stored content is important especially for Kademlia since in this DHT lookups for a key without a responsible node usually roam around longer than lookups for keys with a responsible node. In the second phase of the simulation, each node performs 10 lookups for different random stored keys. In each of the remaining parts, 32% percent of the nodes is replaced in a 400-sec churn window. In these windows, nodes first leave and then new nodes gradually join the system; the network population always resets to 10000 nodes, so as to preserve the fair lookup and signaling comparison between the phases. Nodes that join the system run the stabilization procedure and fix their long-range contacts according to the current network topology. The next stabilization is called when the timer initiated with the stabilization period elapses; hence, the stabilization periods of nodes are asynchronous, and depend on the time they joined the network.

Analysis of Results. For each simulation run we present two figures. One figure shows the evolution of lookup message cost over time. In order to reduce the number of points, each point represents the average of 1000 consecutive lookups. In order to estimate the value of a result we need to see also the price at which it was obtained. Thus, beside lookup message cost we also present the absolute number of signaling messages that enabled the corresponding performance. One box in these figures depicts the messages sent only in the given simulation phase. Since there are many parameter set combinations, for figures in Fig. 9, we selected a parameter set that adequately represents the protocols, and we altered only the stabilization periods.

In Fig. 9 we can observe several interesting results and phenomena. The most striking one is that the lookup message cost of LR-CAN with the current parameter set is consistently lower in almost all cases. Certainly, with different d and c parameters, we can experience different performance; with higher c value, LR-CAN can further lower lookup message cost. With frequent stabilizations, dynamism has almost no effect on lookups in Chord and LR-CAN; these DHTs keep the same lookup performance over time. In contrast, Kademlia has an interesting behavior on churn. In the second phase, Kademlia has an outstanding lookup cost that deteriorates rapidly in the third phase after the first churn window, which results in many stale entries. After a while, lookup cost converges to the performance experienced before the churn window, but this takes a long time. In the next phase, we can observe a slight shift upward in lookup cost because Kademlia looses some of the keys stored at the beginning of the simulation due to churn and stale entries, and lookups for these keys last longer.

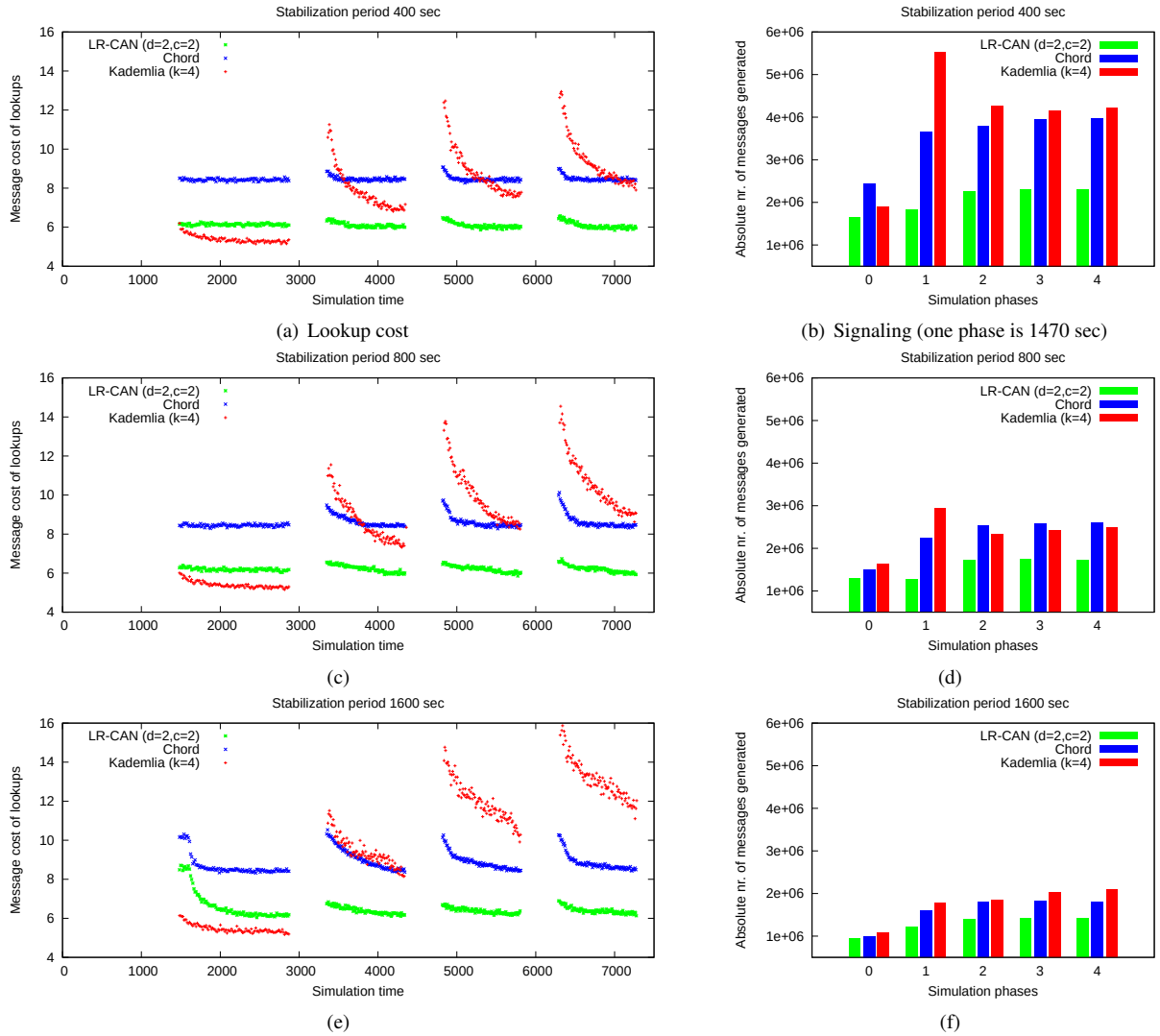


Figure 9. Lookup performance and corresponding signaling traffic in a dynamic network

In Fig. 9(b) we can see the corresponding signaling traffic that contains also the lookup message cost presented by Fig. 9(a). Note again that Kademlia uses its lookup traffic to maintain long-range contacts as well, so maintenance could not be separated from lookups. LR-CAN needs less than half of the signaling of Chord and Kademlia that partly comes from the lower lookup message cost and the more efficient repair of stale long-range entries. In phase 1 we can observe a slightly higher signaling traffic for Kademlia than in later phases, since in this phase there are constantly 10000 nodes present; thus, more bucket refreshes are performed, which are more costly than Chord and LR-CAN stabilizations.

In the rest of the lookup cost figures we can observe that the vertical shift of Kademlia over time becomes more significant, and its ability to converge to the original lookup cost average reduces. Both for Chord and LR-CAN some time is needed until long-range contacts become accurate. With longer stabilization periods, this convergence time also increases. Nevertheless, with its one-dimensional and unidirectional ID space, Chord is more sensitive to network dynamicity; thus, the average lookup cost deteriorates more than in LR-CAN. Regarding the signaling load, the difference between the protocols gets less significant; however, LR-CAN keeps its preliminary advantage.

6. LR-CAN Signaling Optimization

So far the aim of this work was to minimize the message cost of lookups, assuming a reasonable signaling traffic. As

we have seen, the amount of signaling traffic gives a clear view on how much the provided lookup cost is worth. We have also seen that with a well selected stabilization period we may loose some negligible advantage in lookup cost, but we can spare significant signaling traffic load. Besides the stabilization period, the other influential parameter on signaling in LR-CAN is the parameter c that drives the number of long-range levels in a network. Since in a DHT a high portion of the signaling traffic origins from the maintenance of long-range contacts, there might be cases when this kind of network "investment" is not equilibrated by the gains in lookup cost (which is another type of signaling traffic). The low number of lookups injected in the system results in some long-range contacts becoming superfluous; their maintenance costs more than what can be gained on their availability. Overall, fast lookups are desirable in a DHT, but their provision requires a not negligible amount of signaling traffic. Beside the provision of fast lookups, our second aim could be to minimize thus the signaling traffic, i.e., lookup traffic and signaling related to contact maintenance. The rest of this Section describes how most of the signaling traffic can be characterized in LR-CAN by other parameters, and presents a method (together with its specific assumptions and conditions) through which we are able to optimize the number of long-range levels L and minimize signaling traffic.

In LR-CAN, most of the signaling traffic can be well described with some formulas; hence, we can optimize the value of the input parameters in order to minimize signaling traffic accurately. With a few assumptions and modifications to the original LR-CAN algorithm, we can construct an algorithm (SIGMIN) that adapts the number of long-range contacts not only to network size (N) but to lookup rate (λ_l) as well. Here we exclude the cost-limit function from the optimization procedure since in this task we do not care about lookup cost, only about the minimization of aggregated signaling traffic.

The problem is thus to find an optimal value for the number of long-range levels (L) in function of network size (N), long-range maintenance frequency (λ_p), and lookup rate (λ_l). A method for gathering information about N was described earlier in Section 3. The frequency of long-range maintenance, i.e., the stabilization period λ_p is a global network parameter. However, for λ_l we need to implement an aggregated-data-collection technique which collects lookup rate data from the entire network, or at least from a significant part of it. In our solution, data collection is initiated by the bottom-leftmost node of the ID space, which can be a dedicated anchor node in the LR-CAN if it is persistent since the launch of the system; the collection has a cost of $O(N)$. After the data collection, this node can evaluate the input parameters and find the optimal value of L, which is then broadcast to the network. Consequently, the earlier definition of L, which was presented in Section 3 and was based on individual decisions of peers, has to be changed to a method where one node derives the optimal value of L based on aggregated, global information.

By the following equations, we look for the L value that yields the minimal aggregated cost of long-range contact maintenance ($M_{cost}(L)$) and lookup cost ($L_{cost}(L)$) for the whole network, given the parameters described above. $M_{cost}(L)$ includes the message cost of checking one long-range contact, the network size N, the maintenance frequency (λ_p), and the number of long-range contacts when L levels and d dimensions are used (see Equation 7). $L_{cost}(L)$ includes the network size, the lookup rate (λ_l), and the routing cost on long- and short-range contacts (see Section 4).

$$M_{cost}(L) = \begin{cases} 2 \cdot N \cdot \lambda_p \cdot (1+L), & \text{if } -1 \leq L < 0 \\ 2 \cdot N \cdot \lambda_p \cdot (1+L \cdot 2^d), & \text{if } L \geq 0 \end{cases} \quad (8)$$

$$L_{cost}(L) = N \cdot \lambda_l \cdot (LR_{avg}(L) + \frac{\frac{d}{2} N^{\frac{1}{d}}}{2^{L+1}} \cdot m_{\frac{avg}{max}}), \text{if } L \geq -1 \quad (9)$$

$$LR_{avg}(L) \approx \begin{cases} 0.5 \cdot (1+L), & \text{if } -1 \leq L < 0 \\ 0.5 + d \cdot 0.343 \cdot L, & \text{if } L \geq 0, d = 1, 2 \end{cases} \quad (10)$$

The global optimum for L is:

$$L_{opt} \approx \arg \min_{\left(\left[-1, \log_2 N^{\frac{1}{d}}/2\right], L \in \mathbb{R}\right)} (M_{cost}(L) + L_{cost}(L)) \quad (11)$$

For the long-range cost $LR_{avg}(L)$, we gave a close approximation in equation 10, derived from the average cost resulting from the following formulas:

$$\begin{aligned} c(L) =& 2^L \cdot \left(\frac{1}{2} + \frac{L}{2}\right) - \\ & -2^L \cdot \left(\sum_{i=2}^{L} \left(\frac{1}{2^i} + \sum_{j=1}^{\lfloor \log_2 (i-1) \rfloor} \frac{1}{8^j}\right)\right), \text{if } L \geq 1 \end{aligned} \quad (12)$$

$$LR_{avg}(L) = \begin{cases} \frac{1}{2}, & \text{if } L = 0 \\ \frac{(c(L)-1)}{2^L} + \frac{1}{2^{L+1}}, & \text{if } d = 1, L \geq 1 \\ \frac{(c(L)-1)}{2^{L-1}} + \frac{1}{2^{2L+1}} - & \\ -4 \cdot \sum_{i=1}^{\lfloor \log_2 (L+1) \rfloor} \frac{1}{4^i} \cdot \sum_{i=1}^{\lfloor \log_2 (L) \rfloor} \frac{1}{4^i} & \text{if } d = 2, L \geq 1 \end{cases} \quad (13)$$

The validity of these formulas can be checked against the weighted average of routing zones, as presented in Fig. 3 and Fig. 4. Equations 12 and 13 prove the validity of equation 10 only for maximum two dimensions; however, two dimensions are adequate for LR-CAN and SIGMIN, since the lookup performance does not depend on d in these two algorithms, as mentioned in Section 4.

Solving equation 11 yields the optimal value of L, which will be a real value. However, on a given node, L must have

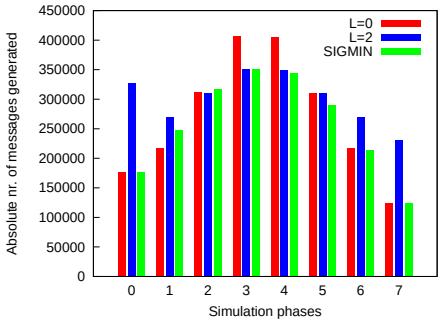


Figure 10. Simulation of lookup rate influence on signaling

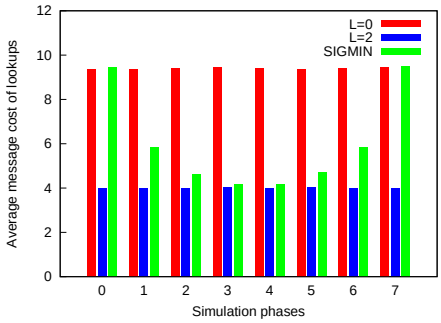


Figure 11. Corresponding lookup cost of the same simulation

an integer value. Thus, the nodes will individually choose the integer value either immediately below or immediately above the optimal value, using a simple stochastic method. As a result, the global average of L will be close to the optimal value.

We need to optimize the performance of the network in the future based on past data. Hence, peers should provide detailed information about their lookup activity on a regular basis (e.g., daily), not only a simple average of a certain larger timeframe. On the other hand, the decision made in advance is only valid if network population does not decrease or increase significantly in the timeframe the decision is made for, or the change has to be predictable.

The simulation that produced the graphs in Fig. 10 consisted of equal timeframes with constant node population. Only lookup rate changed in each phase, linearly increasing from phase 0 to 3, being constant in phase 3 to 4, and decreasing from phase 4 to 7. In one case L was set to 0, in another case L was set to 2 in a static manner for the entire network, while in the third case nodes used the SIGMIN algorithm to determine the value of L. SIGMIN was started with no long-range contacts. Note that setting up new levels of long-range contacts has some non-negligible cost, which is clearly shown in the first phase for the $L = 2$ variant (boxes in the middle); if $L = 0$, there are much less contacts to set up and maintain (boxes on the left). SIGMIN (boxes on the right) might also be affected by this additional signaling cost, as an increasing lookup rate might trigger the introduction of new long-range contacts, according to equations 8–11. This effect can be seen in phases 1 and 2; hence, in phase 2 SIGMIN generates a higher signaling load than the other solutions, although only by a slight margin (this casual cost is not expressed by the formulas 8–11). The remarkable ability of the SIGMIN algorithm is to always converge to the more efficient variant in terms of signaling (Fig. 10). The average message cost of lookups corresponding to the variants is depicted in Fig. 11.

7. Conclusion

In this paper, we proposed an enhanced algorithm for CAN, which reduces the scope of the original routing method in the ID space by deploying long-range contacts. These contacts are deployed adaptively, so as to limit the maximum possible route length in the original greedy forwarding step of CAN. If the lookup cost exceeds a defined limit as network grows, LR-CAN deploys new levels of long-range contacts. As a consequence, the lookup cost becomes proportional to $O(logN)$ for the right $SR(N)$. Our simulations prove our theoretical performance evaluation to be appropriate. Compared to popular DHT algorithms (e.g., Chord and Kademlia), LR-CAN has an outstanding performance in networks of large-scale and even in dynamic scenarios.

We also presented a method to describe the signaling traffic of LR-CAN originating from message cost of lookups and contact maintenance. By a few measurable parameters we can optimize the number of long-range contacts in the network to minimize the signaling traffic of LR-CAN.

Acknowledgment

The authors would like to the Hungarian National Information Infrastructure Development Programme for the supercomputer cluster we were allowed to use for our time and resource consuming simulations.

References

[1] S. Ratnasamy et al. A scalable content-addressable network. In *Proc. of the ACM SIGCOMM*, pages 161–172, San Diego, CA, 2001.

[2] I. Stoica et al. Chord: Scalable peer-to-peer lookup service for internet applications. In *Proc. of the ACM SIGCOMM*, pages 149–160, San Diego, CA, 2001.

[3] J. Jiang et al. Bichord: An improved approach for lookup routing in chord. *Lecture Notes in Computer Science, ADBIS*, 3631:338–348, 2005.

[4] P. Maymounkov and D. Maziéres. Kademlia: A peer-to-peer information system based on the xor metric. In *Proc. of the 1st IPTPS'02*, pages 53–65, Cambridge, MA, 2002.

[5] A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Middleware 2001*, pages 329–350, Heidelberg, Germany, 2001.

[6] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393(6684):440–442, 1998.

[7] J. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proc. 32nd ACM Symposium on Theory of Computing*, pages 163–170, Portland, OR, 2000.

[8] A. Ohtsubo et al. The power of shortcuts in greedy routing in content-addressable networks. In *Lecture Notes in Computer Science, EUC*, volume 3207, pages 994–1003. Springer Berlin, 2004.

[9] S. Ratnasamy et al. A scalable content-addressable network. In *TR–00–010*, Berkeley, CA, 2001.

[10] V. Darlagiannis, A. Mauthe, N. Liebau, and R. Steinmetz. An adaptable, role-based simulator for p2p networks. In *Proc. of Int. Conference on Modeling, Simulation, and Visualization Methods*, pages 52–59, Las Vegas, NV, 2004.

[11] Peerfactsim.kom. http://www.peerfact.org, 2007.

[12] Gnutella. http://www.gnu.org, 2007.

[13] D. Liben-Nowell et al. Analysis of the evolution of p2p systems. pages 233–242, 2002.

[14] B. Kovács and R. Vida. An Adaptive Approach to Enhance the Performance of Content-Addressable Networks. In *Proc. of ICNS 2007*, pages 93–98, Athens, Greece, 2007.

Real-time Data Flow Scheduling for Distributed Control

Anca Hangan, Ramona Marfievici, Gheorghe Sebestyen
Department of Computer Science, Faculty of Automation and Computer Science
Technical University of Cluj-Napoca, Romania
E-Mail: {Anca.Hangan, Ramona.Marfievici, Gheorghe.Sebestyen}@cs.utcluj.ro

Abstract

Remote process control and supervision applications developed over the TCP/IP networks require special communication models and techniques, which can guarantee the real-time and safety restrictions inherent to automation systems. This paper presents a reservation-based communication system architecture and a communication model based on data flow analysis that offer a good control over the transmission time of critical data. As part of the model, an analytical method is proposed that allows a-priory evaluation of the required minimum bandwidth necessary to assure the satisfaction of real-time transmission restrictions.

Keywords: *distributed control, reservation-based scheduling, real-time traffic, communication model, quality of service*

1. Introduction

Remote process supervision and control systems require a communication infrastructure that supports reliable and safe real-time data transmission. These requirements are usually solved by using dedicated networks and industrial protocols, as presented in [1] and [2]. But these special purpose protocols are incompatible with general-purpose protocols used in local area networks and company intranets. If there are interoperability requirements between distributed control applications and organizational software, the incompatibility of industrial protocols and the TCP/IP stack may be a problem. Local computer networks and TCP/IP protocols, on the other hand, fail to satisfy real-time requirements because they apply the best-effort principle in supplying communication services, and it is quite difficult to use them as infrastructure for real-time applications.

During the last years, a significant research trend emerged for using best-effort (non-deterministic) networks for real-time communication. This trend was caused by technical developments such as the increasing network bandwidth (Gbps) and the development of new Quality of Service (QoS) mechanisms. The need for the integration of distributed control systems with other information systems that do not require special communication services was another cause.

To satisfy real-time communication requirements on TCP/IP networks, solutions that enable a predictable network behavior and that also provide end-to-end delivery time guarantees have to be developed.

In this paper we propose a reservation-based communication system architecture for distributed control applications on TCP/IP networks. As part of the solution, we define a control traffic model and a network bandwidth estimation method.

1.1 Related work

Several authors investigated the problem of accommodating real-time traffic on TCP/IP networks and proposed some solutions.

Wijnants and Lamotte present in [3] a method for managing the network bandwidth for multiple client applications. Their communication middleware, the NIProxy, is able to partition available client bandwidth between real-time and non real-time traffic flows by arranging them in a stream hierarchy. This solution gives good results in improving client Quality of Experience, but it does not guarantee any end-to-end timing requirements for real-time traffic.

Another approach is presented in [4], where the authors propose introducing a prioritization mechanism in the TCP/UDP/IP protocol stack, a mechanism which complies with the IEEE 802.1D standard. They evaluate the solution through simulation, using the OPNET simulator, by measuring

end-to-end latency of real-time packets in the presence of FTP traffic on the same network. Their conclusion is that a large part of the end-to-end message latency occurs at the end nodes, assuming that the network bandwidth is large enough to support all the traffic. In the referred paper the authors do not provide a solution for evaluating network bandwidth requirements, they just assume that the bandwidth is large enough.

In [5], Martinez et al. present Earliest Deadline First (EDF) communication scheduler implementation adapted for high-performance networks. The characteristics of high-performance networks enabled them to simplify the calculus of packet deadline, taking into consideration only the previous packet's deadline, packet size and average bandwidth.

A good technique which enables the provision of real-time guarantees is the reservation of resources for each task, in accordance to its requirements. In [6] and [7] reservation techniques are combined with feedback techniques to provide delay and execution time guarantees for tasks that coexists in a shared environment.

Schantz et al. describe two approaches, priority-based and reservation-based, in developing distributed real-time middleware [8]. For both solutions, the communication infrastructure is an IP network. In the priority-based middleware the standard Differentiated Services (DiffServ) mechanism is implemented for network resource management. In the reservation-based middleware the Integrated Services (IntServ) mechanism is implemented. Their main contribution is in the area of middleware implementation. But there are two quite important issues not addressed by this paper: (1) the provision of instruments for evaluating the resource requirements of real-time tasks and (2) the differentiation between hard and soft tasks.

IntServ [9] [10] provides end-to-end per-flow QoS by means of hop-by-hop resource reservation within the IP network but impose a significant burden on the core routers. To reduce the complexity within each core router, alternative schemes, referred to as Measurement Based Admission Control Schemes (MBAC) have been proposed [11]. These schemes replace per-flow states with run-time link load estimates performed in each router. However, MBAC solutions still require significant modification of the existing Internet architecture, as core routers must support load estimation algorithms, and still need to be explicitly involved in per flow signaling exchange.

A completely different approach is provided by DiffServ [12]. In DiffServ, core routers are stateless and unaware of any signaling. While DiffServ easily enables resource provisioning performed in a management plane for permanent connections, their widely recognized limit is the lack of support per-flow resource management and admission control, resulting in the lack of strict per flow QoS guarantees. A number of proposals, presented in the literature, have shown that per flow Distributed Admission Control schemes can be deployed over DiffServ architectures [13] [14]. Although significantly different in implementation, they share the common idea that accept/reject decisions are taken by the network endpoints and are based on the processing of "probing" packets, injected in the network at setup to verify the network congestion status. A "pure" Extended Admission Control (EAC) scheme, called Phantom Circuit Protocol-Delay Variation (PCP-DV) is proposed in [15]. The scheme determines whether a new connection request can be accepted based on delay variation measurements taken on the probing packet at the edge nodes.

Reinemo et al. [16] propose and evaluate three different admission control schemes for virtual cut-through networks, each one suitable for use in combination with DiffServ based QoS scheme to deliver soft real-time guarantees. Two of the schemes assume pre-knowledge of the network's performance behavior without admission control and are both implemented with bandwidth broker. The third is based on endpoint/egress admission control and relies on measurements to assess the load situation. Due to the way the flow control affects latency and the nature of cut-through networks, latency and jitter properties are hard to achieve.

An approach to quantify the impact of end-to-end QoS provisioning through a combination of both intra and inter-autonomous system (AS) traffic engineering (TE) is proposed in [17]. Two offline QoS-aware systems are deployed for this and a direct relationship between intra-AS and inter-AS TE is then established. The interaction between them is analyzed and both the decoupled and integrated approaches are presented.

In [18], several possible algorithms for routing and scheduling which allow coexistence of QoS and best-effort flows are presented. The network algorithm takes into account state imprecision in routers, maxmin bandwidth allocation, and existing link state information.

1.2 Our contribution

This paper introduces a communication model, which is based on control data flow analysis and communication scheduling that uses a rate-monotonic algorithm. The communication model solves efficient delivery for short control messages over TCP/IP

networks and facilitates a-priori estimation of required network bandwidth for the reservation mechanism.

A reservation-based communication system architecture is also proposed. Our solution uses Integrated Services/RSVP and the facilities of IPv6 protocol as support for real-time communication. The implemented middleware translates real-time requirements to existing network services and is used to validate both the communication model and the system architecture.

The remainder of the paper is organized as follows: Section 2 presents some specific communication-related issues about distributed industrial control systems. In section 3 we discuss the details of the proposed system architecture. A new data flow-based traffic model is introduced in Section 4. Section 5 describes the method for bandwidth estimation, based on communication scheduling. Section 6 covers some important implementation aspects. The experiments and results analysis are described in Section 7. Section 8 concludes the paper.

2. Problem description

Distributed control systems are an integral part of the industrial automation domain. Their functionalities include data acquisition, monitoring and control of industrial processes. While the majority of the control systems are usually located within more confined area (e.g. plant area, company local network) and communications are usually performed using local area network (LAN) technologies that are typically reliable and high-speed, other are geographically distributed (e.g. SCADA systems) and need long-distance communication systems such as the Internet.

Our research has the objective of solving the communication issues of distributed control systems which are deployed in the companies' local TCP/IP networks. Usually, these networks are managed by the companies and the nodes (hosts, switches, routers, servers) are configured and administered by a company internal authority. Such a network has to accommodate two broad categories of traffic: non-real-time and real-time. Non-real-time traffic can adjust to changes in delay and throughput and is generated by applications that include common Internet-based applications, such as file transfer, electronic email, remote logon, network management, and Web access. Real-time traffic does not easily adapt, if at all, to changes in delay and throughput and have requirements that include beside delay and throughput, delay variation and packet loss.

In addition, these corporate networks may be connected to strategic partner networks and to the Internet, thus, making more use of Wide Area Networks (WANs) and Internet to transmit their data to remote stations.

Most control applications must satisfy real-time and safety constraints. A very important parameter in real-time environments is the system response time, defined as the time between the occurrence of an event and the corresponding response. In distributed systems, message delivery time, has a large influence on the system's response time. Network protocols must incorporate message delivery time control mechanisms in order to guarantee maximum delivery time for control messages. These mechanisms assume a deterministic network behavior, which permit a-priori evaluation of maximum message delivery time.

When measuring the performance of a real-time communication system, the following parameters are taken into consideration [19]:

- Deadline miss rate (fraction of all messages that are delivered to late at the destination)
- Delay jitter (the variation of message delays)
- Loss rate (fraction of all messages that are dropped on the route from source to destination)

For hard real-time applications deadline misses are not acceptable, moreover message response time must be guaranteed a-priori. But delay jitter may not cause serious problems as long as deadlines are satisfied. In the case of soft real-time applications deadline misses are tolerable to some extent, but, under some particular conditions, delay jitter may have negative effects. In the case of distributed control systems traffic, one has to deal with both hard and soft real-time requirements. For these reasons, the goal is to minimize both message response time (end-to-end delay) and delay jitter.

In the case of using a TCP/IP network for real-time communication, some important issues may arise. First of all, a maximum response time for packets has to be guaranteed. This can be a serious problem knowing that TCP/IP networks function on a best-effort basis.

Another communication issue is message delivery efficiency. Data transmitted through the network in distributed control systems are quite different compared to data transmitted by usual applications which generate traffic in TCP/IP networks. Control applications use short, unstructured data (e.g. digital signal values). Process control data is generated, mostly, at well determined periods of time. The majority of supervision and control functions involve data acquisition, processing and storage, visualization

of process status and command issuing, which require a short reaction time.

Control applications include different automation and computing devices, which are interconnected. In order to assure interoperability, the communication protocol must allow uniform and transparent access to system's resources and it must be simple enough to allow implementation on devices with limited computing resources.

Last but not least is the issue of real-time and non-real-time traffic coexistence. In the case of remote process control it is quite possible to have both traffic (real-time) generated by the control system and traffic (best-effort) generated by other applications (e.g. office automation) that run in the same network. Network bandwidth has to be managed in order to assure real-time requirements for control traffic and also to assure fair treatment for the best-effort traffic.

The objective of this paper is to take a new approach in solving some of the following communication issues:

- Guarantee packet delivery time in TCP/IP networks in the presence of both real-time and non-real-time traffic
- Assure predictability of network behavior
- Assure transmission efficiency of process control data
- Provide device interoperability and uniform access to process control data

3. System architecture

In order to provide a comprehensive communication system architecture based on IP infrastructure so that to meet the challenges of quality of service provisioning for industrial control application we integrate in three major components: (1) industrial control applications and processes; (2) a middleware system (service manager) between the application and the protocol driver; this middleware closely interacts with Internet protocol stack; (3) network infrastructure based on IPv4 or IPv6 protocol.

The first component of the system architecture represents quality of service demanding applications that use a QoS API to send requests to the service manager. These applications generate periodic and aperiodic traffic. The traffic is characterized by packet size, transmitting data rates, priority, and accepted latency.

The middleware bridges the industrial applications and the underlying network systems by dispatching the application requests and returning status and feedback from the underlying system to the application. Examining the application requests and the available network resources, the middleware selects a provisioning service or service level, maps the application QoS to network-specific quality of service, and initiates resource allocation or renegotiates the parameters with the application before the flows' source starts to generate any packets.

The following components were integrated in the proposed middleware:

- Traffic QoS specification
- QoS negotiation
- Traffic and QoS monitoring
- Resource reservation
- Data transfer

3.1 Traffic QoS Specification and QoS Negotiation modules

An application which wants to set up a connection in order to transmit packets to another application in the network uses the means of the traffic QoS specification to set up a reservation request first. This module is a generic API so that an application demanding quality of service is isolated from the complexity of the provisioning services.

The application defines its generic QoS specification in terms of traffic profile which is composed of parameters that characterize the traffic stream or session (source IP address and port number, destination address, transport protocol) and parameters that define quantitatively the network performance requirements (transmitting rates, message size, transmission deadlines, latency), which can be specified using maximal, average and minimal values.

The traffic QoS specification module contains a set of rules for converting the traffic characteristics to parameters in the underlying message model.

Based on the input from the QoS specification module, the QoS negotiation module is responsible for authorizing the request and check if the network is able to support the new connection interacting with the resource reservation module for resource allocation. The goal of this module is to provide optimal quality of service with respect to critical parameters and previous requests.

Application's requests for quality of service parameters can be solved in two ways: positive, in case

the resource reservation module sends a positive acknowledge to the QoS negotiation module that there are enough resources in the network to satisfy the request and the reservation is set along the path, and negative. In case of a negative notification, the application may invoke the QoS negotiation module in order to find what resources and services are available in the network and to adjust the quality of service requirements and start a new negotiating procedure.

3.2 Traffic and QoS Monitoring module

In this module components are included for monitoring network resources (available bandwidth, average utilization of a link, delay, jitter) and quality of service related statistics from routers (queue length, number of conforming/exceeding packets in bytes, number of dropped packets, CPU utilization). It also signals significant changes in resource availability.

When an application establishes a network traffic stream, this module starts collecting its performance. It collects data from traffic stream, including quality of service specification, connection times, transmission rates and delays, and communicate the quality of service parameters to the QoS negotiation module in order to determine if there is any quality of service violation. All collected data is stored into a management information base.

3.3 Resource Reservation module

The resource reservation module is the ultimate authority for the resource handling in the proposed architecture (Fig. 1). Its main building blocks are admission control and reservation setup. Admission control implements request authorization by checking if the network is able to support the flow and the decision algorithm that nodes use to determine whether a new flow can be granted the requested quality of service with/without impacting earlier guarantees. For these tasks it closely interacts with the main entity, the resource reservation protocol.

Resource ReSerVation Protocol (RSVP) [9] is used for resource reservation signaling. It is designed to enable the senders, receivers and routers of communications sessions to communicate with each other to reserve resources for new flows at a given level of QoS. On the other hand, the reservation protocol is responsible for maintaining flow specific state information at the end nodes and at the nodes along the path of the flow.

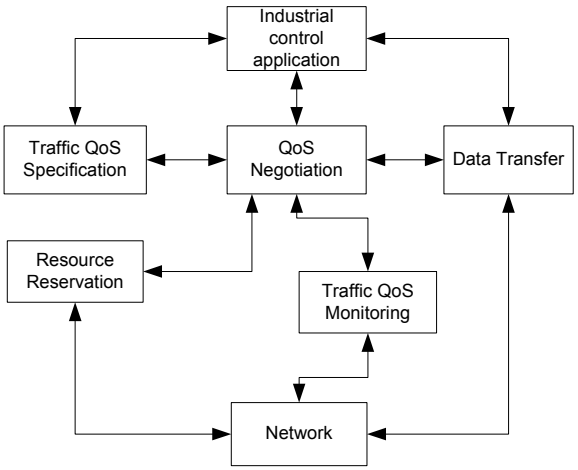


Figure 1. System architecture – components

RSVP requests result in resources being reserved in each node along the data path. Given below are the main attributes of this protocol: it requests resources in only one direction (it treats a sender separately from a receiver, although the same application might be running at both the sender and the receiver); is receiver-oriented (the receiver of a data flow initiates and maintains the resource reservation used for that flow); RSVP itself is not a routing protocol, but it is designed to work with the existing routing protocols; RSVP supports both IPv4 and IPv6.

To make a resource reservation at a node [20], our RSVP daemon uses the admission control mechanism. If check fails, the RSVP returns an error notification to the QoS negotiation module that originated the request. If checks succeed, the RSVP daemon sets the parameters.

4. The data flow model

In order to make an analytical evaluation of the traffic generated by the distributed control system, it is required to classify this traffic and then, based on the identified types, to define the traffic model.

4.1 Traffic classification

Control traffic is generated by data exchanged between the control applications and industrial devices, such as:

- Values obtained through data acquisition, with a well defined frequency (e.g. temperature in an oven, liquid level in a tank, engine state – started/stopped, etc.)
- Commands generated at known periods of time

- Operator commands (e.g. start/stop engine, increase oven temperature to 200 degrees, etc.)
- Process events, alarms, alerts, etc.

The previous categories of data generate periodic and aperiodic network traffic, with real-time constraints.

4.2 Model definition

To model the control traffic, we introduce *data flows* [21]. A data flow is the sum of all packets sent through the network that have the same source, destination, content and periodicity. Traffic between control applications and devices connected to the process is a sum of periodic and aperiodic data flows. As an example, consider an application that monitors the temperature in a room. Temperature sensors measure the temperature in the room at the same time, with the periodicity of five minutes and send the data to the process computer. This computer packs the temperature values into packets and sends them to the monitoring application. All these packets containing temperature values create a data flow.

Fig. 2 shows the data flows established between two control applications connected to remote industrial processes, through a TCP/IP network.

Periodic data flows include values obtained through data acquisition, control commands, which occur at well defined periods of time. Aperiodic data flows include commands issued by the application operator, high priority alerts and event signals. A number of parameters are identified for each data flow type. The parameters for periodic data flows are: inter-release period, priority (importance), content (process control data included in the flow), required packet delay (or response time), transmission deadline, source, destination and packet size. The parameters for aperiodic data flows are: priority, content, required packet delay, transmission deadline, source, destination and packet size.

In real-time task modeling, it is a common practice to assume that aperiodic tasks have a minimum inter-release period, which is given by process related parameters. Because all tasks are considered periodic, scheduling and feasibility analysis are simplified. For the same reasons, we choose to make the same assumption (minimum inter-release period) for aperiodic data flows.

A data flow is formally defined as an n-tuple:

$$DF = (T, P, r, D, l, Src, Dest, c) \quad (1)$$

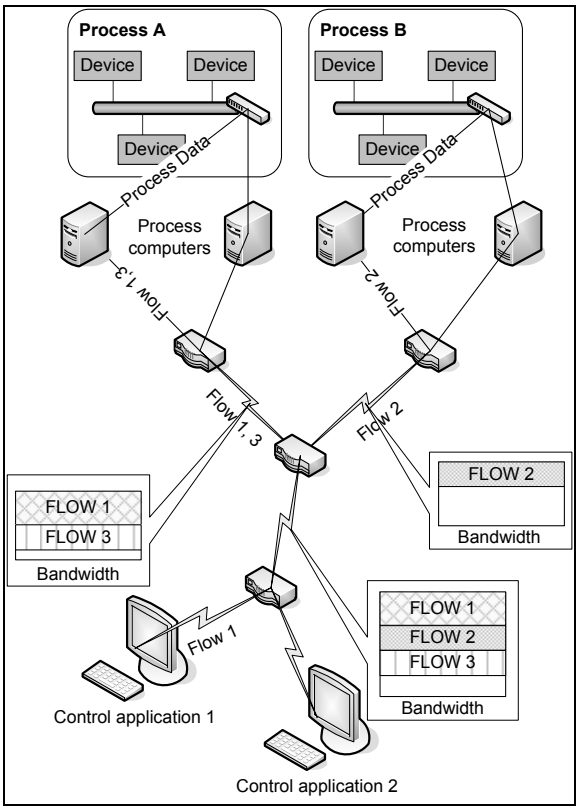


Figure 2. Data flows between control applications

The components of the n-tuple are the data flow parameters. T is the inter-release period for periodic data flows and the minimum inter-release period for aperiodic data flows. P is the priority, r is the required packet delay or response time, D is the transmission deadline and l is the size of a data flow packet. The last three components are the source (*Src*), destination (*Dest*) and content (c) of the data flow.

4.3 Communication optimization

The protocols from the TCP/IP stack are optimized to deliver large packets of data in a best-effort manner. Process control messages, on the other hand, are very short (from a few bytes to hundreds of bytes); they have periodic occurrence and real-time constraints. If very short periodic messages are packed and released in a TCP/IP network, the protocol overhead is very large compared with the payload data. Because, messages with short period of occurrence (e.g. seconds, milliseconds) can create large amount of traffic although few effective data is transmitted, it can be said that the network is inefficiently utilized for data transmission.

```
count = 0;
Foreach process_data _value
{
    DF[count] = Create_data_flow ();
    Set_data_flow_parameters ( DF[count] );
    count++;
}
Sort_data_flows_by_period ();
For ( i=0; i < count-1; i++ )
{
    If ( Periodic ( DF[i] ) &&
          ! Marked_for_delete ( DF [i]))
    {
        j = i+1;
        While ( DF[i].period == DF[j].period )
        {
            If ( Periodic ( DF[j] ) )
            {
                If ( DF[i].src == DF[j].src )
                   and ( DF[i].dest == DF[j].dest )
                {
                    Aggregate ( DF[i], DF[j] );
                    Mark_for_delete ( DF[j] )
                }
            }
            j++;
        }
    }
}
Foreach DF
{
    If (Marked_for_delete ( DF ))
    {
        Delete ( DF );
    }
}
```

Figure 3. The algorithm pseudocode for the aggregation of data flows

To optimize the transmission of control packets, we can adopt the strategy of aggregating control data from different process devices into the same data flow.

To organize control data into larger data flows, the following parameters must be considered:

- Data acquisition periodicity, assuming that devices which perform data acquisition with the same period read the sensor values virtually at the same time
- Data priority
- Data source and destination

By performing data aggregation, the data flow packets contain larger amounts of effective data, hence increasing the efficiency of control data transmission.

The specification of data flows for a control system is obtained using the algorithm presented in Fig. 3. First, an array of data flows (*DF*) is created by defining a data flow for each piece of process data. The parameters are set for each data flow. The array of data flows is then sorted by data flow period. To aggregate data flows that have similar characteristics, each periodic data flow is compared to all other periodic data flows that have the same period (*DF[i].period == DF[j].period*). If the data flows have the same source (*DF[i].src == DF[j].src*) and destination (*DF[i].dest == DF[j].dest*), they are put together in the same data flow. The aggregated data flow will contain data from both initial data flows, will have the highest priority (*max(DF[i].priority,DF[j].priority*)) and the smallest deadline (*min(DF[i].deadline,DF[j].deadline*)) of the two data flows. The first data flow will be substituted by the aggregated data flow and the second data flow will be deleted.

For aperiodic data flows, which cannot merge with other flows, transmission efficiency is reduced. A solution for these flows, if their frequency of occurrence is high, is to reserve space for aperiodic data in periodic flows. This space (e.g. a few bytes) is used only if the aperiodic event takes place right before a periodic data flow packet is sent.

5. Communication scheduling

Solving the issue of message transmission time control is critical in a distributed control system. The system architecture proposed in this paper uses a TCP/IP network as a communication infrastructure. The main challenge in this case is guaranteeing real-time constraints on a best-effort communication infrastructure. For solving this problem, a bandwidth reservation mechanism is used.

The bandwidth reservation mechanism requires the estimation of network bandwidth for all traffic generated in the system. For this purpose we adopted a method commonly used in the case of real-time tasks, the worst case scenario analysis using the rate-monotonic scheduling algorithm [22].

In our approach, each data flow is a “task” and the network is the “processor”, which has to be shared between all “tasks” in the system. Priorities are assigned to all data flows in a rate-monotonic manner. This way, a data flow which has a lower period will have a higher priority. Periodic and aperiodic data flows are taken into consideration for scheduling.

Aperiodic data flows are considered to have a minimum inter-release period. The next step is to compute the response time for each data flow. The data flow system is feasible if, for each data flow, the response time is less than its transmission deadline (r < D). In this work we consider that transmission deadline for a data flow is equal to its inter-release period (T = D). By obtaining the appropriate values for the response time of all data flows, in the worst case, a maximum value for the required network bandwidth can be derived. The maximum bandwidth value obtained is used to make resource reservations. In this way, it can be guaranteed that actual response time for each data flow will be less or equal than the computed response times.

It is considered that the worst case response time for a data flow happens when a packet has to wait for the transmission of packets that belong to all data flows with higher priority and for one packet with lower priority, but with the largest transmission time. It is also assumed that all data flows start at the same time.

In order to compute the data flow response time (r_i) the following variables are taken into consideration:

- The delay caused by the devices found on the network path (t_{delay})
- The transmission time of packets that belong to the data flow (C_i)
- The data flow's inter-release period (T_i)
- The data flow's priority (Pi)
- The transmission time of packets that belong to data flows with higher priority
- Maximum transmission time of packets that belong to data flows with lower priority
- The number of hops from source to destination ($nHops$)

Response time for each data flow is computed using the following set of equations:

$$\begin{cases} r_i^{\,0} = t_{delay} + nHops * (C_i + \sum_{P_j > P_i} C_j) \\ r_i^{t+1} = t_{delay} + nHops * (C_i + \sum_{P_j > P_i} \left\lceil \frac{r_i^t - C_i}{T_j} \right\rceil * C_j + \\ \quad + Max\{C_k \mid P_k < P_i\}) \end{cases} \quad (2)$$

The response time is obtained through iteration, until $r_i^{t+1} = r_i^t$. There are two important conditions which have to be imposed:

- All response times have to be less than the corresponding deadlines
- Network utilization has to be less than 100%

The bandwidth value is gradually increased (with 10%) while the response time and utilization are computed, until these requirements are met.

In the first iteration, the response time of a data flow is computed by summing up the transmission time, the delay caused by the network devices such as switches and routers, and the transmission time of all other data flows which have higher priority. In subsequent iterations, the response time equation has two new components, the maximum transmission time of data flows which have lower priority ($Max\{C_k \mid P_k < P_i\}$) and the sum of transmission time of all packets of higher priority that are likely to be released while the packet is being transmitted through the network. The number of packets with higher priority that influence the response time of the data flow equals the number of packets that are released in the response time of a packet from the data flow. To decrease the number of iterations, the time interval when the actual packet is being transmitted (C_i) is subtracted from the response time, as in that time interval packets from other data flows can not be transmitted.

The transmission time for packets which are included in a data flow is computed as follows:

$$C = \frac{Packet\_length}{Bandwidth} \quad (3)$$

The delay caused by network devices can be approximated by using a mean round-trip time of a probe packet sent on the same route on which the bandwidth reservation will be made.

As the response time is computed recursively, the computation time could be a problem. In our case, because bandwidth requirements are assessed and reservations are made before starting the control system, a larger computation time is not an issue.

6. Implementation details

To validate the proposed system architecture, the data flow model and the method of estimating network bandwidth, a prototype of a distributed control system, was developed. The prototype includes the

communication middleware, the industrial application and the industrial device simulator (for the provision of industrial process data).

The communication middleware runs on a network infrastructure based on TCP/IP stack, with IPv6 as a network protocol. The solution adopted [23] is to use the IPv6 Traffic Class and Flow Label fields. The Traffic Class field enables a source to identify desired traffic-handling characteristics of each packet relative to other packet from the same source. The intent is to support various forms of services. In case of IPv6 standard [24], a flow is defined as a sequence of packets sent from a particular source to a particular destination for which the source desires some special handling by the intervening routers. From the source's point of view, a flow is just a sequence of packets that are generated from a single application instance at that source and have the same transfer service requirements. From the router's point of view, a flow is a sequence of packets that share attributes that affect how these packets are handled by the router. In principle, all of a user's requirements for a particular flow could be defined in an extension header and included in each packet, but for our implementation, we leave the concept of flow open to include a wide variety of requirements and adopt the flow label, in which the flow requirements are defined before traffic generation and a unique flow label is assigned to the flow.

The RSVP module is designed as a state machine. The objects defined in this module represent:

- RSVP sessions
- State information extracted from PATH message and information from RESV message
- Reservations installed in an outgoing interface
- Information about a previous hop in a session, i.e. the last reservation that has been sent to this hop

For each RSVP session all relevant information is bundled and the destination address and port is saved. From each PATH message all relevant information is held, i.e., the sender's address and traffic specification, routing information. For each reservation requested from a next hop, reservation specification is held, i.e., the FlowSpec, which determines the amount of resources that are requested, depending on the service class.

The industrial control application uses all the facilities offered by the communication middleware and implements the following functionalities:

- Remote process control and visualization
- Input and output data flow definitions for devices participating in the industrial process
- Control data flow through commands sent to devices connected to processes
- Specification and negotiation of resources needed for communication with other devices
- Receive and process data flows from industrial devices
- Register data flow delay time

The operator can visually create the diagram of the industrial process, by dragging the symbols of different types of devices on the control board. Next, the operator has to specify input data flows (data received from devices connected to the process) and output data flows (commands sent to devices) in order to establish communication parameters.

After the definition of data flows, the negotiation process for resources starts. Input and output data flows are analyzed and, as a result, bandwidth needed to satisfy real-time communication constraints is computed. The application sends a query asking for the available bandwidth and round-trip time to destination process. The response time can be guaranteed only for the data flows having the period less than the delay caused by the network devices (e.g. switches, routers). If the available bandwidth is insufficient, data flows having the smallest period are deleted, data is recomputed and application begins the resource reservation process. After the negotiation and reservation process, the application can start to send and receive data flows.

The industrial device simulator sends periodical data flows (requested by the control application) containing process values randomly generated from a predetermined range and receives periodical data flows representing commands from the control application for devices connected to the process. Devices cannot negotiate resource reservations for generated data flows nor to specify quality of service parameters. The control application connected to these devices is responsible for the negotiation and bandwidth reservation.

An important issue encountered during the implementation of both the industrial control application and the device simulator is the specification of data flows. In order to assure the device and application interoperability, data flow parameters and content are specified using XML. In this way messages between applications and devices are interpreted easier and the access to process and control data is uniform.

```
<Flow>
    <ID> data_flow_ID </ID>
    <SrcIP> source_IP </SrcIP>
    <DestIP> destination_IP </DestIP>
    <Per> data_flow_period </Per>
    <Pri> data_flow_priority </Pri>
    <Name> data_flow_symbolic_name </Name>
    <Content>
        XML_content_specification
    </Content>
</Flow>
```

Figure 4. Data flow specification in XML

Fig. 4 shows an example of a periodic data flow specification in XML.

7. Experiments

We conducted two sets of experiments. First, we used a simulator to validate the proposed method for network bandwidth estimation. Second, we performed some tests using the implemented control system prototype, which was deployed on our experimental infrastructure.

7.1 Estimation of network bandwidth

For the first set of experiments we measured the response time, jitter and packet loss for multiple periodic real-time data flows, which were released in a simulated TCP/IP network. The main objective of these experiments was to check if the computed network bandwidth value guarantees the required response time and low jitter for all data flows.

The simulation study was performed on Network Simulator (NS-2) [25], version 2.33. The simulation results were evaluated for different scenarios using the topology depicted in Fig. 5.

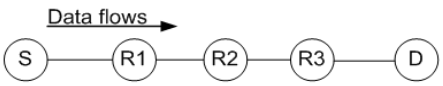


Figure 5. The topology used for simulations

Table 1. Parameter settings for periodic real-time data flows

Flow	*Period (ms)*	*Packet size (B)*
F1	10	300
F2	120	300
F3	50	300
F4	75	300
F5	520	300

Table 2. Response times (1st scenario)

Flow	*Response time (ms)*		
	Measured maximum	*Measured average*	*Measured minimum*
F1	50.66	28.87	25.33
F2	50.66	33.01	25.33
F3	50.66	32.98	25.33
F4	50.66	30.48	25.33
F5	50.66	33.03	25.33

The topology consists of 5 nodes. These nodes are connected with full-duplex bidirectional links. All links have the same available bandwidth and propagation delay. In this paper it is assumed that per link delay is negligible. Constant-Bit-Rate (CBR) agents were attached to the source node (S) and used to generate periodic, fixed size packet traffic in the network. User Datagram Protocol (UDP) was used as transport layer protocol to minimize the overhead of establishing a connection. Five periodic data flows were defined, having the same source (S) and destination (D). The parameter settings are summarized in Table 1.

Two scenarios were simulated. Measurements were made to compare data flows' response times with the corresponding deadlines and to observe to what extent the jitter affects the response time of packets.

In the first scenario the network bandwidth was set to the minimum value which can accommodate all defined data flows (379 Kbps). Results analysis revealed that the average measured response times were acceptable in the case of data flows which had larger periods, but for the other data flows response times were very often greater than the corresponding deadlines. Jitter measurements showed that even if the average value was quite small, the maximum value was very large, approximately equal to the measured minimum response time. For this scenario no packets were lost.

For the second scenario, equations (2) were used to derive the maximum bandwidth needed by the set of data flows in order to satisfy the deadlines. The computed maximum bandwidth was 2106 Kbps. As expected, a considerable difference can be observed between the measured maximum response time and the maximum computed response time. This difference is due to the fact that the worst-case scenario does not occur during simulation time, thus the resulting network utilization is low. All the deadlines were satisfied and the average delay jitter is very small for the flow with the largest period. There was no packet loss.

For both scenarios, measured values can be found in Tables 2-5 and the comparison between data flows

in terms of response time and jitter are shown in Fig. 6-9.

7.2 Tests performed using the prototype

To test the distributed control system prototype, two PCs connected in a local network were configured as traffic source and destination. A static route consisting of another two PCs which played the role of routers was established between these nodes. The network infrastructure was based on IPv6 protocol. The communication middleware, the control application and the device simulators were deployed on the test infrastructure.

A process schema containing monitoring elements connected to two data flows was specified in the control application. The first data flow (Flow 1) has a 2 seconds period and 270 byte packet size. The second data flow (Flow 2) has a 0.5 second period and the same packet size. After starting the remote control application and the device simulators, response time for all packets was measured.

Table 3. Response times jitter (1st scenario)

Flow	*Response time jitter (ms)*		
	Measured maximum	*Measured average*	*Measured minimum*
F1	25.33	3.22	0
F2	18.68	3.58	0
F3	19	4.33	0
F4	24	4.41	0
F5	25.33	3.59	0

Table 4. Response times (2nd scenario)

Flow	*Response time (ms)*			
	Measured maximum	*Measured average*	*Measured minimum*	*Computed*
F1	6.078	3.103	3.039	9.12
F2	6.078	3.839	3.039	68.4
F3	6.078	3.870	3.039	27.36
F4	6.078	3.447	3.039	41.04
F5	6.078	3.724	3.039	86.64

Table 5. Response times jitter (2nd scenario)

Flow	*Response time jitter (ms)*		
	Measured maximum	*Measured average*	*Measured minimum*
F1	3.039	0.114	0
F2	2.279	0.547	0
F3	2.279	0.484	0
F4	3.039	0.815	0
F5	3.039	0.002	0

Measurements were made in two cases. In the first case, the communication middleware was used to make network bandwidth reservations before starting the traffic. In the second case, no reservations were made for the real-time traffic. For both data flows, response time measured during tests was less than the maximum allowed response time, in the case of reservations, presented in Table 6.

The measurements showed that the proposed system architecture, traffic model and method of data flow scheduling are able to satisfy the control system's requirements and guarantee a maximum delivery time. They also showed that the analytical evaluation of the response time is an upper limit to the measured time parameters.

If no reservations were made, for both data flows, measured response time fluctuated between a minimum of 0.367 seconds and a maximum of 0.617 seconds, as can be observed in Table 7. Packets of Flow 1 have the same priority on the network as packets of Flow 2, even though Flow 2 requires a better response time. Real-time requirements were not satisfied, because for Flow 2 the maximum measured response time was greater than the computed maximum response time.

Fig.10 and Fig. 11 show charts that compare the computed response time for the two data flows with the measured response time, on both test scenarios.

Table 6. Measured response time for experimental data flows with reservations

	Flow 1	*Flow 2*	*Flow 1*	*Flow 2*
Computed bandwidth	9 kbps			
Available bandwidth	100 kbps		64 Mbps	
Measured RTT	1.5 ms		0.65 ms	
Computed maximum response time	0.745 s	0.497 s	0.744 s	0.496 s
Measured response time	0.685 s	0.372 s	0.677 s	0.367 s

Table 7. Measured response time for experimental data flows without reservations

	Flow 1	*Flow2*
Computed bandwidth	9 kbps	
Available bandwidth	100 Mbps	
Measured RTT	0.4 ms	0.4 ms
Computed maximum response time	0.744 s	0.496 s
Maximum measured response time	0.617 s	0.617 s
Minimum measured response time	0.367 s	0.367 s

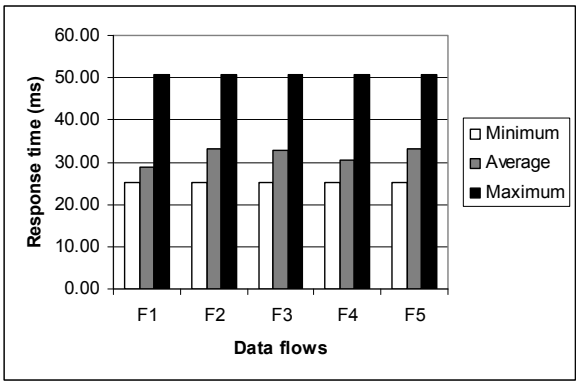


Figure 6. Response times (1st scenario)

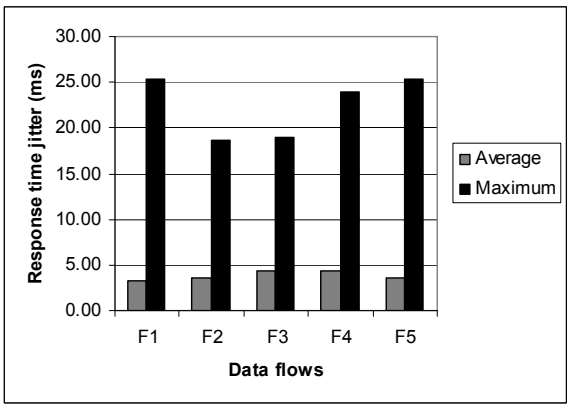


Figure 7. Response times jitter (1st scenario)

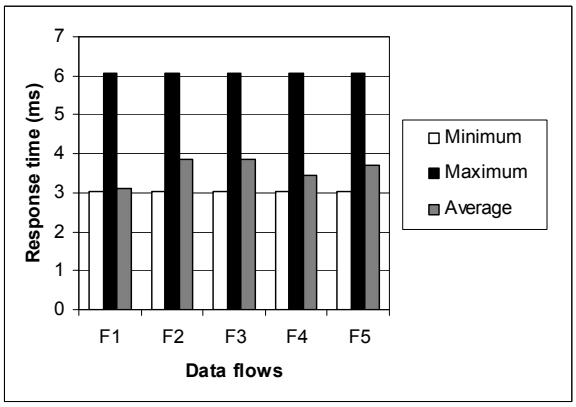


Figure 8. Response times (2nd scenario)

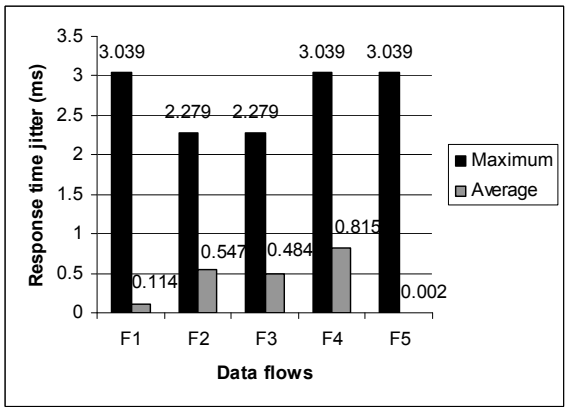
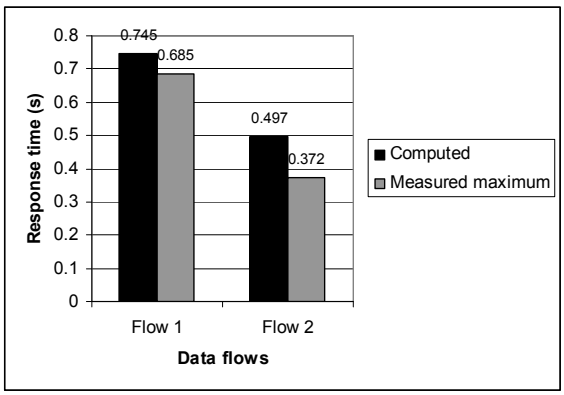
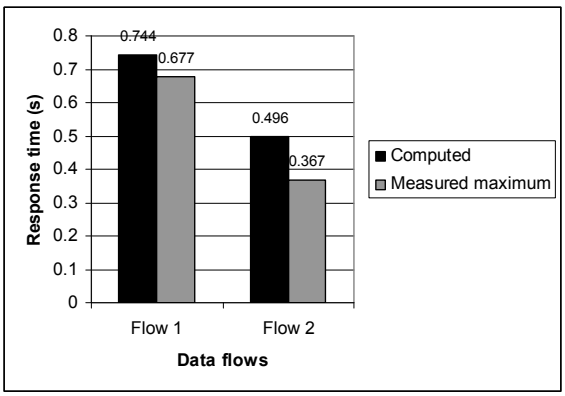


Figure 9. Response times jitter (2nd scenario)



(a) Available bandwidth = 100kbps



(b) Available bandwidth = 64Mbps

Figure 10. Response time measurements using reservations

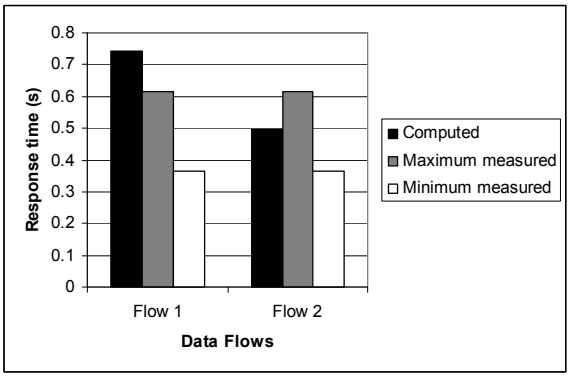


Figure 11. Response time measurements without using reservations

8. Conclusion

This paper presents a new approach in solving network delivery time control and data delivery efficiency in distributed control systems, on TCP/IP infrastructures. We introduce the data flow traffic model which provides the basis for communication scheduling, and a reservation-based communication system architecture. Our solution uses Integrated Services/RSVP and the facilities of IPv6 protocol as support for real-time communication.

The experimental results show that the implemented prototype satisfies the real-time constraints. This proves the validity of the proposed communication model and the method for computing the required network bandwidth. Moreover, the analytical evaluation of response time demonstrated to be an upper limit for measured delivery time.

9. References

[1] L. B. Fredriksson, "Controller area networks and the protocol CAN for machine control systems", Mechatronics, vol. 4, no. 2, pp. 159-192, 1994

[2] S. Koubias and G. Papadoupoulous, "Modern fieldbus communication architectures for real-time industrial applications", Computers in Industry, vol. 26, no.3, pp. 243-252, Aug. 1995

[3] M. Wijnants, W. Lamotte, "Managing client bandwidth in the presence of both real-time and non real-time network traffic", 3rd International Conference on Communication Systems Software and Middleware and Workshops, (COMSWARE), Bangalore, Jan. 2008, pp. 442-450

[4] T. Skeie, S. Johannessen, O. Holmeide, "Timeliness of real-time IP communication in switched industrial Ethernet networks", IEEE Transactions on Industrial Informatics, Volume 2, Issue 1, Feb. 2006, pp. 25 – 39

[5] A. Martínez Vicente, G. Apostolopoulos, F. J. Alfaro, J. L. Sánchez, J. Duato, "Efficient Deadline-Based QoS Algorithms for High-Performance Networks," IEEE Transactions on Computers, vol. 57, no. 7, Jul., 2008, pp. 928-939

[6] C. Lu, Y. Lu, T. Abdelzaher, J. Stankovic, S. Hyuk Son, "Feedback Control Architecture and Design Methodology for Service Delay Guarantees in Web Servers", IEEE Transactions on Parallel and Distributed Systems, vol. 17, no. 9, Sep. 2006, pp. 1014-1027

[7] L. Abeni, L. Palopoli, G. Lipari, J. Walpole, "Analysis of a Reservation-Based Feedback Scheduler", IEEE Real-Time System Symposium (RTSS), Austin, Texas, 2002, pp. 71

[8] R.E. Schantz, J.P. Loyall, C. Rodriguez, D.C. Schmidt, Y. Krishnamurthy, I. Pyarali, "Flexible and Adaptive QoS Control for Distributed Real-time and Embedded Middleware", Proceedings of Middleware 2003, ACM/IFIP/USENIX international middleware conference, Rio de Janeiro , Brazil, 2003, pp. 374-393

[9] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification", RFC2205, September 1997

[10] J. Wroclawski, "The use of RSVP with IETF Integrated Services", RFC2210, Sept. 1997

[11] S. Georgoula.; P. Trimintzios, G. Pavlou, K.H. Ho, **"**Heterogeneous real-time traffic admission control in differentiated services domains**",** IEEE Global Telecommunication Conference, (GLOBECOM), Volume 1, 28 Nov.-2 Dec. 2005

[12] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, "An Architecture for Differentiated Services", RFC2475, Dec. 1998

[13] F. Kelly, R. Key, S. Zachary, "Distributed Admission Control", IEEE Journal on Selected Areas in Communications, Vol. 18, Dec. 2000, pp. 2617-2628

[14] L. Breslau, E. Knightly, S. Shenker, I. Stoica, H. Zhang, "Endpoint Admission Control: Architectural Issues and Performance", in Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication 2000, Stockholm Sweden, pp. 57-69

[15] G. Bianchi, F. Borgonova, A. Capone, L. Fratta, C. Petrioli**, "**Endpoint admission control with delay variation measurements for QoS in IP networks", ACM SIGCOMM Computer Communication Review, vol. 32, no. 2, April 2002, pp. 61 - 69

[16] S.-A. Reinemo, F. O. Sem-Jacobsen, T. Skeie, O. Lysne, "Admission Control for diffServ Based Quality of Service in Cut-through Networks", 10th International Conference on High Performance Computing (HiPC 2003), ed. by Timothy Mark Pinkston and Viktor K. Prasanna, pp. 118-129, Heidelberg, Springer. Lecture Notes in Computer Science

[17] K.-H Ho, M. Howarth, N. Wang, G. Pavlou, S. Georgoulas, **"**Two Approaches to Internet Traffic Engineering for End-to-End Quality of Service Provisioning", 1st EuroNGI Conference on Next Generation Internet Networks - Traffic Engineering, Rome, Italy, 18-20 April 2005, pp. 135 – 142

[18] K. Nahrstedt, S. Chen, "Coexistence of QoS and Best Effort Flows - Routing and Scheduling", Proceedings of 10th Tyrrhenian International Workshop on Digital Communications: Multimedia Communications, Ischia, Italy, Sept. 1998

[19] Liu, J. W.S., *Real-Time Systems*, Prentice Hall, 2000

[20] S. Shenker and L. Breslau, "Two Issues in Reservation Establishment", ACM SIGCOMM '95 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Cambridge, 1995, pp. 14-26

[21] A. Hangan, R. Marfievici, Gh. Sebestyen, "Reservation-Based Data Flow Scheduling in Distributed Control Applications" – The Third International Conference on Networking and Services, ICNS 2007, 19-25 June 2007, Athens, Greece, in Proceedings of the Third International Conference on Networking and Services, IEEE Computer Society Washington, DC, USA, 2007, ISBN: 0-7695-2858-9, pp. 10-15

[22] C.L. Liu, J. W. Layland, "Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment", Journal of the ACM, Vol. 20, No. 1, January 1973, pp.46-61

[23] R. Marfievici, Gh. Sebestyen, A. Pop-Bidian, "Industrial Control Communication Framework based on an IPv6 Infrastructure" – International Multi-Conference on Computing in the Global Information Technology – Challenges for the Next Generation of IT&C, ICCGI 2006, Bucharest, Romania, 2006

[24] R. Banerjee, The Internet Protocol version 6 (IPv6): issues and challenges, Technical Report, Computing Science Laboratory, Oxford University, Feb. 2002

[25] ***, The Network Simulator - ns-2, http://www.isi.edu/nsnam/ns/

Service Triggering in MVNO & Multi-Country environments

Marc Cheboldaeff
Alcatel-Lucent
cheboldaeff@alcatel-lucent.com

Abstract

Traditional mobile operators invested huge amounts of money in the 1990s to build the current 2G wireless networks, like GSM networks in Europe. Those networks have proven to be stable. In most cases, their capacity has not yet been exhausted. Furthermore, the marketing departments of mobile operators now consider the commercial possibility of selling mobile subscriptions through new channels, like supermarkets. The concept of Mobile Virtual Network Operator (MVNO) arose. Some of them are just new brands and do not own any tele-communications equipment. Other companies do own part of the network: these are the Mobile Virtual Network Enablers (MVNE). They provide part of the network infrastructure, while the Mobile Virtual Network Operators (MVNO) serve end-customers. Furthermore, in the context of international globalization, it appears more and more meaningful that the same platform in one dedicated country serves end-subscribers from several operators in different countries. When it comes to service triggering, the interactions between three kinds of network are critical. These three kinds of networks are: the home network, which owns the services' platform, the host network, which actually triggers the platform in the home country, and visited networks, where the subscriber may roam abroad. The goal of the present paper is to study these interactions based on an actual implementation example.

Keywords: IN, Service Trigger, MVNO, Multi-Country, IMS

1. Introduction

We shall focus on signaling issues for triggering a Value Added Service (VAS) such as an Intelligent Networks (IN) service. The signaling to establish the bearer trunk will not be the central focus of this paper; instead, we shall concentrate on the signaling for the exchanges with the services' platform.

We shall assume that the owner of the services' platform like the MVNE has its own Network Sub-System (NSS), which represents the true *Home Network*. First, it owns the register of its subscribers or Home Location Register (HLR). That is, it has the freedom for provisioning subscribers. In addition, it owns a VAS platform. Finally, it owns a core network, including Mobile Switching Centers (MSC) and Signaling Transfer Points (STP). For the Radio Access Network (RAN) or Base Sub-System (BSS), the MVNE relies on a traditional mobile operator defined as the *Host Network*. The fact that the MVNE owns a VAS platform means that it has the flexibility to offer differentiated services in comparison with other mobile networks including the host network.

A IN prepaid service is a good example of a Value Added Service because it contains both:

1. Originating triggers: a subscriber pays to make a call; thus the prepaid service is triggered for an outgoing call.
2. Terminating triggers: a subscriber might pay to receive calls: for example when he/she is roaming abroad in a *Visited Network*. For such incoming calls, the service is triggered too.

Like in [1], we shall study first the outgoing and terminating case for a single MVNO. To illustrate this, we give an implementation example. Then, we tackle the Multi-Country topic, which may apply when MVNOs are located in different countries, as well as when an international operator has affiliates in various countries. Finally, we shall outline what the interconnections between the home network and external networks could look like in the IMS architecture.

Since many acronyms are used, a terminology can be found at the end of the paper.

2. Making outgoing calls from the Host network

The MVNE does not own its Radio Access Network (RAN). Therefore, in the home country a mobile subscriber of the virtual operator is first detected by the RAN of the host network. The Host Network needs to retrieve the subscriber information from the HLR of the MVNE. Based on that information, it will trigger the VAS platform of the MVNE.

Through which protocol? The two usual protocols for value-added services such as IN services are INAP and CAMEL. For interoperability issues, especially in the roaming case (see chapter 3) the CAMEL protocol might be preferred. CAMEL, sometimes called CAMEL Application Part (CAP), as INAP, is built over TCAP [2], which is built itself over SS7, and therefore CAMEL resp. INAP messages transit through the SS7 network by means of Signaling Transfer Points (STP).

If CAMEL is used, the IN platform is triggered as illustrated in Figure 1. This applies to voice calls, as well as to SMS, in case the latter are charged through an SS7-based protocol.

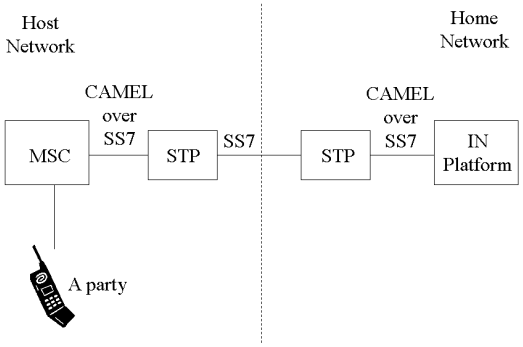


Figure 1. *„Home" Mobile Originating Call*

The A-party is known in the host network. At registration time, the VLR gets the information from the HLR of the MVNE about the Originating CAMEL Subscription Information (O-CSI) for that subscriber. The O-CSI contains, among other things, the SS7 address or Global Title (GT) of the IN platform belonging to the MVNE. The MSC retrieves that information in turn from the VLR. Based on that information, the MSC of the host network builds and forwards the service triggering message to the IN platform in the home network. In fact, the message is built by the Service Switching Functionality contained in the MSC. That functionality is called SSF or gsmSSF as described in [3]. The message later transits through the STP nodes of the SS7 network.

In the case of GPRS resp. UMTS networks, the role of the MSC consisting in triggering the Value Added Service would be played by the SGSN, which would contain also an SSF, called this time gprsSSF instead of gsmSSF. For more details, the reader can refer to [4].

We might call this scenario a "home" call since the subscriber is in its home country. However, the call is handled partly by the host network. So in reality, it is not completely a "home" call.

When the subscriber moves within the host network from one area to another, the VLR corresponding to the current area for the subscriber is updated with the service triggering information related to the A-party. The local MSC retrieves the updated information from the VLR and is still able to contact the IN platform in order to trigger the Value Added Service.

The reader who wishes to get more information on that topic can refer to the chapter 'Mobility Procedures' in [5].

3. Making outgoing calls from a visited network

The main reason why the CAMEL protocol is preferred is that it is fully standardized. You cannot find so many various implementations as with INAP. Consequently, while a subscriber is visiting a mobile network abroad, the Visited MSC (V-MSC), which monitors the call legs, can talk directly through CAMEL with the IN Platform of the MVNE. That latter provides the instructions for handling the call.

In CAMEL, there are different subsets of capabilities or *Phases*, where the next phase is a superset of the previous one. The actual CAMEL phase to handle the call should be the minimal phase supported by both the visited MSC and the IN platform of the MVNE. For example, if the IN platform supports Phase 2, but the visited MSC only supports Phase 1, then Phase 1 will be the relevant phase to handle the call.

The IN platform is triggered as illustrated in Figure 2. In case data traffic was charged through an SS7-based protocol like CAMEL Phase 3, the scenario described would still apply.

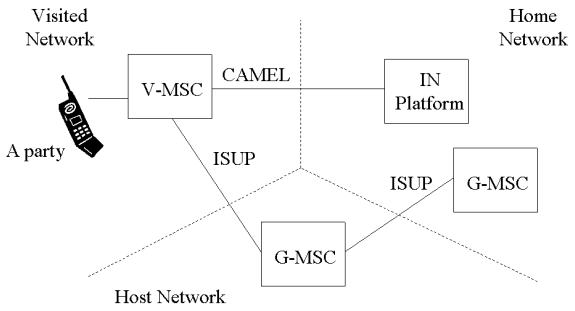


Figure 2. *Roaming Mobile Originating Call*

The IN platform can only be triggered once the A-party has been identified as a subscriber of the MVNE. When the MVNE subscriber roams abroad, the VLR is updated with the O-CSI of that subscriber. From the visited VLR, the visited MSC retrieves the GT of the IN Platform belonging to the MVNE, so that it can get instructions from the IN platform on how to handle the call. Once that information is retrieved, the V-MSC can monitor the call. The bearer trunk is established through the ISUP protocol, from the visited network to the host network, which forwards the call to the G-MSC of the home network. In case the MVNE is officially registered at regulatory authorities, and does not operate under the umbrella of a well-established operator, it can happen that the bearer trunk transits directly from the visited network to the home network through an international carrier, thus bypassing the host network.

Having a CAMEL dialogue between the visited and the home networks means of course that the visited MSC has CAMEL capabilities. However, CAMEL is not necessarily implemented by all roaming partners, and even if it is, it may not be in the right phase. For example, in order to play announcements, CAMEL Phase 2 is required. And in order to charge data traffic, CAMEL Phase 3 is required. Consequently, if the visited MSC has only Phase 1 at its disposal, then the home network might better control the call itself. For that purpose, the control of the call needs to be passed over to the home network. We will study how this could work in Chapter 4.

4. CAMEL Rerouting

Let us consider the case in which the V-MSC offers CAMEL Phase 1, and the control of the call is passed over to an MSC in the home network.

When a subscriber is abroad, the V-MSC takes care of all outgoing calls. The V-MSC knows which IN platform to trigger when the subscriber tries to make a call since the VLR is updated with the O-CSI from the HLR of the MVNE. Once triggered, the IN platform of the MVNE can instruct the V-MSC to connect the call to a dummy destination number belonging to the home network. That dummy number could contain a sequence number in order to identify the call later. When the ISUP 'Initial Address Message' (IAM) to set up the voice call reaches the G-MSC of the home network, the G-MSC could trigger again the IN platform of the MVNE. Based on the sequence number defined earlier, the IN platform would be able to correlate the incoming voice call with the service triggering message previously received, and would thus take control of the call in this fashion.

This is illustrated in Figure 3.

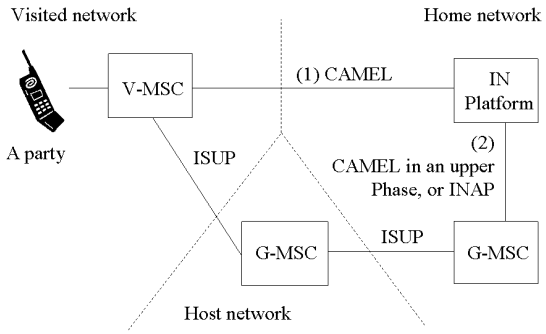


Figure 3. *CAMEL Rerouting*

When the G-MSC of the home network triggers the IN platform the second time, it can use INAP. It is not necessary to use CAMEL anymore since the MSC and the IN platform belong to the same network. Therefore, the rerouting scenario is also a way to control an originating call using INAP.

There could be also a rerouting scenario for home calls between the host network and the home network. There would be a first trigger to the IN platform from the host network. Later, a second trigger would come from the G-MSC of the home network. As the first trigger comes necessarily from the host network, INAP could be used also in order to trigger the IN platform the first time from the host MSC. It would only be a matter of agreement between the host and the home networks, not with other partners.

5. Receiving calls

In the terminating case, the called number or B-Party is one of the subscribers of the MVNE, which is

not necessarily the case for the A-Party. The network, where the A-party is located, is called the *Interrogating Network*. For more information on that topic, the reader might refer to [3], especially the chapter 'Architecture'.

As in the originating case, the G-MSC of the interrogating network needs to set up the connection to the B-party and get the CAMEL subscription information from the HLR of the MVNE. In that case, it is not the Originating but the Terminating CAMEL Subscription Information (T-CSI). That piece of information contains again the GT of the IN platform belonging to the MVNE. Consequently, there can be a CAMEL dialogue between the G-MSC of the interrogating network and the IN platform, as illustrated in Figure 4.

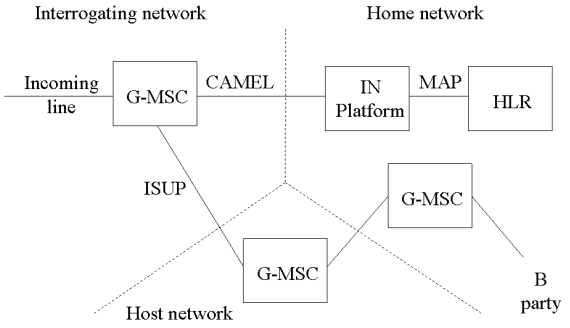


Figure 4. *Mobile Terminating Call*

Again, this assumes that the G-MSC has CAMEL capabilities and that the supported CAMEL phase is sufficient to provide the desired service to the end-subscriber.

The host network is involved in the terminating scenario, too, since the B-Party, which is a subscriber of the MVNE, is seen as belonging to the host network by any foreign network.

In order to avoid the problem with CAMEL compatibility, the T-CSI can be disabled in the HLR, and the G-MSC of the MVNE can trigger the IN platform itself based on the incoming ISUP message to set up the voice call, similar to the second trigger in the CAMEL re-routing concept. It is possible because the bearer trunk will always reach an MSC of the home network in the terminating scenario. This is illustrated in Figure 5.

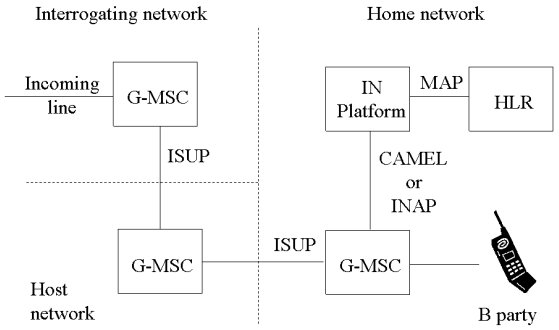


Figure 5. *Mobile Terminating Call*

In that case, the terminating call is monitored by the G-MSC of the MVNE itself, with instruction from the IN platform.

6. Implementation

The present chapter describes an actual implementation of an IN prepaid platform at the MVNE *Vistream* in Germany in 2006.

6.1 Making outgoing calls

For outgoing calls from the host network, CAMEL Phase 2 has been used in order to have the capability to play announcements to the end-subscriber.

Here is the log for an Initial DP message, which is the initial CAMEL message in order to trigger the IN platform:

```
TRACE: cml!initial_dp_received(
transaction_id= 1576,
[...],
calling_party_number=cml_calling_party_number(
present=true,nature_of_address=anoa_internatio
nal_number,number_incomplete=false,numbering_p
lan_indicator=cmlnp_isdn,presentation_indicato
r=iapi_restricted,screening_indicator=isi_netw
ork_provided,address_signals=4915701234518),
[...],
called_party_bcd_number=cml_called_party_bcd_n
umber(present=true,type_of_number=cmlcs_unknow
n,numbering_plan_indicator=cmlbnp_isdn,address
_signals=015701234593),
[...],
event_type_bcsm=cml_p2_event_type_bcsm(present
=true,value=cmlp2etb_collected_info)
[...],
msc_address=cml_isdn_address_string(present=tr
ue,nature_of_address=cmnt_international_number
,numbering_plan_indictator=cmlbnp_isdn,address
_signals=491770381000),
[...]
)
```

Please note that only the transaction identifier, the calling party number, called party number, event type and MSC address have been kept from the original message log. The other parameters have been removed for the sake of clarity.

In the present log, a MVNE subscriber, whose number (without Country Code) is 015701234518, calls another subscriber from the MVNE, whose number is 015701234593. The 01570 prefix is characteristic for the MVNE. The present trigger relates to the A-party of the call i.e. 015701234518 since the IN platform has been triggered in Detection Point (DP) 'Collected Info' in accordance with the Basic Call State Model (BCSM) defined in [3].

The GT of the MSC, which sent the CAMEL message, is an address in the host network: 491770381000. That piece of information needs to be passed over to the IN platform since it is relevant in order to rate the call to know whether the call comes from abroad or not. If the subscriber were roaming in a foreign network, the MSC address would have another Country Code other than 49 for Germany, and the call would be more expensive for the calling party.

6.2 Receiving calls

For mobile terminating calls, the decision has been taken to have the G-MSC of the MVNE triggering the IN platform using CAMEL Phase 2.

Here is the log for an initial DP message:

```
TRACE:   CAMEL_PROT_FSM[85874]:   call_id   =
'553658458'   state  =  'CML_IDLE'   event  =
'LLS_cml_initial_dp_receivedTyp'

TRACE: cml!initial_dp_received(
transaction_id=1605,
[...]
calling_party_number=cml_calling_party_number(
present=true,nature_of_address=anoa_national_s
ignificant_number,number_incomplete=false,numb
ering_plan_indicator=cmlnp_isdn,presentation_i
ndicator=iapi_allowed,screening_indicator=isi_
network_provided,address_signals=01638080080),
[...],
called_party_number=cml_called_party_number(pr
esent=true,nature_of_address=anoa_national_sig
nificant_number,internal_network_number_indica
tor=iini_route_to_number_allowed,numbering_pla
n_indicator=cmlnp_isdn,address_signals=0157012
34518),
[...],
event_type_bcsm=cml_p2_event_type_bcsm(present
=true,value=cmlp2etb_term_attempt_authorized),
[...],
msc_address=cml_isdn_address_string(present=tr
ue,nature_of_address=cmnt_international_number
,numbering_plan_indictator=cmlbnp_isdn,address
_signals=491570012360),
[...]
)
```

In this example, the GT of the MSC, which sent the CAMEL message, is an address from the MVNE: 49**1570**012360. Again, we recognize the prefix (0)1570. The present trigger relates to the incoming call for the B-Party: 015701234518 since the DP is 'Terminating Attempt Authorized'. The A-Party number, 01638080080, is not a number of the MVNE.

In the case of a Mobile Terminating call, the IN platform needs to know the location of the B-Party. If the latter were in the home country, the call would most probably be free of charge: the subscriber then would not need to pay to receive calls in the home country. It would certainly not be the case if the B-Party were abroad.

In order to know the location of the B-party, the IN platform sends a MAP Any Time Interrogation (ATI) message to the HLR. The reader might have a look again at Figure 5, and refer to [5] for further information.

Here is an example of an ATI message:

```
TRACE: mapss7!send_anytime_interrogation(
[...],
msisdn=map_v2_isdn_address(present=true,noa=mn
t_international_number,np=mnp_isdn_telephony,d
igits=4915701234518)),
requested_info=map_requested_info(location_inf
ormation=true,subscriber_state=true,current_lo
cation=false),
[...]
) call_id=553658458
```

This way, the IN platform knows in which country the called party is and can rate the call accordingly.

7. Multi-country MVNO

Given that the aim of an MVNE is offering services to multiple MVNOs, it can happen that these MVNO have different partnerships with mobile network operators, thus leading the MVNE home network to interface with more than one host network. It is especially the case in the context of international globalization, where MVNO can be located in different countries. One example would be when an international supermarket chain with subsidiaries in different countries wants to launch parallel MVNO offers.

For an MVNO located in another country than the MVNE, it makes sense that the MVNO relies on a local host network, in order to avoid some interconnection costs. This makes it necessary for an MVNE to interface with multiple host networks. It is represented in Figure 6.

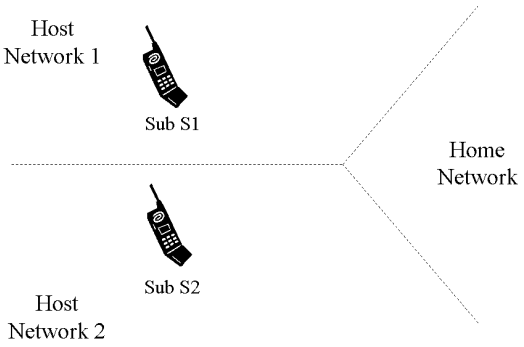


Figure 6. *Multiple Host Networks*

In the case that two host networks are not located in the same country, it can also happen that S2, a subscriber of an MVNO relying on Host Network 2 is roaming to Host Network 1. This is represented in Figure 7.

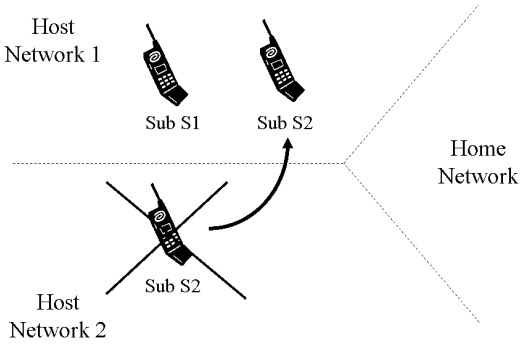


Figure 7. *Subscriber roaming to another Host Network*

Will the two triggering messages for S1 and S2 be different given that they may come from the same SSP in the same Host Network, possibly using the same protocol?

7.1 MVNO Differentiation

Regarding the originating SSP in the Host Network, the SCP is indeed able to identify it.

The incoming triggering message, which is for whatever IN triggering protocol a TCAP message [2], provides the following information:

SS7 Protocol	Available parameter
TCAP	Application Context
SCCP	Originating GT
MTP	Originating PC

Nevertheless, this information depends on the originating SSP only, not on the MVNO. Consequently, the MVNO distinction cannot be done by the "network", unless it is done at an upper level like CAMEL i.e. while the SSP retrieves the subscriber-specific application information, which is the O-CSI in the case of an outgoing CAMEL trigger.

Please note that this MVNO differentiation at Application level is valid not only when two MVNO rely on two hosts networks located in two different countries, but also if there are two MVNO relying on the same host network in the same country because in this case, the triggering information coming from the SSP might be the same also.

If we look at the O-CSI, the following parameters are available according to [3]: gsmSCF Address, Service Key, Default Call Handling, TDP List, DP Criteria, CAMEL Capability Handling. If we do not use different gsmSCF addresses (Global Titles) to identify in reality the same IN platform, the relevant parameter for MVNO differentiation would be the Service Key. In other words, MVNO1 would use Service Key 1, while MVNO2 would use Service Key 2 when triggering the IN platform.

This means that two service instances, triggered by two different service keys, would coexist on the SCP. Attached to each of the instance, there would be a different data set: this can make data configuration complex!

Another option is to have the subscriber differentiation not coming from the network, but from the SCP: subscriber S1 would be stored in the SCP with an MVNO "Identifier" (ID), or "Community" ID, or "Class of Service", or whatever the name is, equal to 'MVNO1', while subscriber S2 would be stored with MVNO_ID 'MVNO2'. Based on the subscriber data on the SCP, the service logic, especially the rating, could be different.

Whether data separation relies on a Service Key or on an MVNO ID, it is recommended for a service's platform to support data segmentation: MVNO1 should have the capability to access or modify its data in a secure way, without the risk of access or modification by MVNO2.

7.2 Roaming consideration

Nevertheless, the MVNO ID information stored at SCP level does not tell whether the subscriber is currently roaming or not! If we take the case of multiple host networks located in different countries but belonging to the same international group, it could be an asset to define the same service logic i.e., the

same service key with variations depending only on the originating country and whether the subscriber is currently roaming or not. For example, some menu options like the call history could be barred to roaming subscribers to avoid interconnection fees with the host network, but allowed for calls made in the same country as the local host network.

In this case, the MVNO ID would not be sufficient anymore, but the Country Code of the Country "where the subscriber is not considered as roaming" would be required at SCP level too, so that the service logic, by comparing this Country Code with the Country Code of the originating SSP, knows in the end whether the subscriber is indeed roaming. This means that it can be an asset for a services' platform to store the "home" country of any subscriber as part of the subscriber data.

Otherwise, it could make sense to add an additional parameter, a kind of "roaming flag", in the service triggering message coming from the network, but this would need extending IN protocols.

Of course, another way to avoid this issue is to physically assign a different SCP per MVNO. Therefore, if MVNO1 is routed to SCP1, while MVNO2 is routed to SCP2 with two different GT, GT1 and GT2, it is possible to define the "home" country on SCP1 as the country of Host Network 1, and differently on SCP2 as the country of Host Network 2. It can make maintenance easier, since a downtime for MVNO1 on the SCP1 machine would have no impact on MVNO2 which runs on SCP2. However, it means additional hardware investment for the MVNO, and this needs to be considered in the business case.

Please note that all these considerations regarding the multi-country topic are not relevant only to MVNO, but also to any international operator, with affiliates in various countries, wishing to store subscribers from different countries on the same platform centrally located in a specific country.

8. Service triggering in IMS networks

Let us see what the MVNE scenario could look like in the IMS architecture. The reader might refer to [6] and [7] to get information on the IMS concepts.

The User Equipment (UE) of the A-party first contacts the Proxy Call State Control Function (P-CSCF) of the visited network, as described in [7] in the paragraph 'Roles of Session Control Functions'. Similarly to the V-MSC in the GSM case, the P-CSCF sees the subscriber as belonging to the host network (instead of the home network) and therefore sends a SIP message to the Interrogating Call State Control Function (I-CSCF) of the host network. The latter is able to make the distinction between its own subscribers and the ones from the MVNE. Consequently, the SIP message is sent finally to the home network.

It makes sense that the MVNE owns an I-CSCF capability. This way, the host network does not need to know how the MVNE subscribers are dispatched onto the different S-CSCF of the home network. Otherwise, the host network needs to be informed every time the distribution changes.

If the I-CSCF belongs to the MVNE, as well as the S-CSCF, which corresponds to option 3 in [8], the service triggering scenario is described in Figure 8.

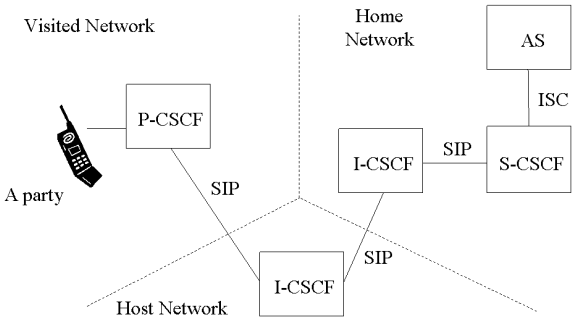


Figure 8. *MO Call by an MVNO Subscriber in IMS*

This means that as soon as the MVNE owns equipment at Session layer, an originating call resp. session is always controlled by an S-CSCF in the home network. The Application Server (AS) will never control a CSCF in the visited network like a Service Control Point (SCP) could do against a visited MSC in traditional GSM networks.

Another option for the MVNE would be that the I-CSCF and S-CSCF belong to the host network. In other words, the MVNE does not own any equipment at Session layer, only at Application layer. It corresponds to option 1 or 2b or 2c in [8].

The service triggering scenario in that case is described in Figure 9.

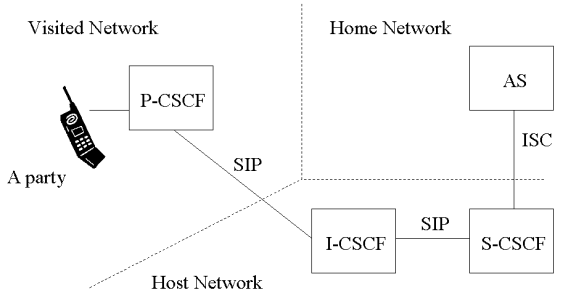


Figure 9. *MO Call when the MVNO only owns the Service Layer*

Please note that in case SMS or more generally data traffic is not charged by means of an SS7-based protocol, but by means of an IP-based protocol, like Diameter, the triggering scenario would be similar. The SMS-C resp. GGSN would forward the charging request to an SMS-C resp. GGSN located either in the home network or in the host network, depending on whether the home network has its own SMS-C resp. GGSN. Finally, the home or host SMS-C resp. GGSN would start a dialogue with the Application Server.

9. Conclusion

In traditional GSM networks, the interaction between a home network and a visited network is a known field. It corresponds to the roaming scenario. At this point, it has been standardized and implemented for years. Through the CAMEL protocol, there can be a direct control of a network component in the visited network by a services' platform in the home network.

With the irruption of the MVNO, which own part of the network equipments, the concept of host network has been introduced between the visited and the home network. However, there can still be control of the visited network by the home network with regard to value-added services. If there can be multiple host networks between the visited and the home network, MVNO differentiation must be done at Application level.

In the IMS architecture, which relies on standard IP protocols like SIP or Diameter, the component that monitors the call resp. session and the services' platform are always within the home network. This is valid even in the roaming case, unless the MVNO owns only the services' platform and no equipment at all at Session level. Consequently, the interactions between the home network and external networks are less critical in IMS when it comes to service triggering.

Terminology

2G	Second Generation
3G	Third Generation
3GPP	3G Partnership Project
AS	Application Server
ATI	Any Time Interrogation
BCSM	Basic Call State Model
BSS	Base Sub-System
CAMEL	Customized Applications for Mobile networks Enhanced Logic
CAP	CAMEL Application Part
CSCF	Call State Control Function
DP	Detection Point
G-MSC	Gateway MSC
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
GT	Global Title
HLR	Home Location Register
IAM	Initial Address Message
I-CSCF	Interrogating-CSCF
ID	Identifier
IMS	IP Multimedia Subsystem
IN	Intelligent Networks
INAP	Intelligent Network Application Part
IP	Internet Protocol
ISC	IMS reference point between CSCF and AS
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
MAP	Mobile Application Part
MO	Mobile Originating
MSC	Mobile Switching Center
MTP	Message Transfer Part
MVNE	Mobile Virtual Network Enabler
MVNO	Mobile Virtual Network Operator
NSS	Network Sub-System
O-CSI	Originating CAMEL Subscription Information
P-CSCF	Proxy-CSCF
PC	Point Code
RAN	Radio Access Network
SCCP	Signaling Connection Control Part
SCF	Service Control Function
SCP	Service Control Point
SIP	Session Initiation Protocol
SMS	Short Message Service
SMS-C	SMS Center
SMPP	Short Message Peer-to-Peer protocol
SS7	Signaling System No 7
SSF	Service Switching Function

STP	Signaling Transfer Point
TCAP	Transaction Capabilities Application Part
TDP	Trigger Detection Point
T-CSI	Terminating CAMEL Subscription Information
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
VAS	Value Added Service
VLR	Visited Location Register
V-MSC	Visited MSC

Acknowledgements

Much of the source material used for this paper derives from work accomplished together with the technical teams of Alcatel-Lucent, E-Plus and Vistream. The author would like to especially thank Robert Bakker, Angelo Lattuada, Ian Logan, Carlos Martinez Garcia, Kun Pang, Wolfgang Pein, from Alcatel-Lucent; Dietmar Kohnenmergen from E-Plus; Jan Geiger, Peter Mros from Vistream; Jon Hill, Ulrich Bellmann, Martin Löffler.

The author would like to thank Michelle Gansle for her review.

References

[1] M. Cheboldaeff, *Interactions between a Mobile Virtual Network Operator and External Networks with regard to Service Triggering*, Proceedings of the Sixth International Conference on Networking (ICN 2007)

[2] International Telecommunication Union, *Specifications of Transaction Capabilities Application Part (TCAP)*, ITU-T Q.771

[3] 3rd Generation Partnership Project, Technical Specification Group Core Network, *Customized Applications for Mobile network Enhanced Logic (CAMEL) Phase 2*, 3GPP TS 03.78

[4] 3rd Generation Partnership Project, Technical Specification Group Core Network, *Customized Applications for Mobile network Enhanced Logic (CAMEL) Phase 3*, 3GPP TS 29.078

[5] 3rd Generation Partnership Project, Technical Specification Group Core Network and Terminals, *Mobile Application Part (MAP) Specification*, 3GPP TS 29.002

[6] 3rd Generation Partnership Project, Technical Specification Group Services and System Aspects, *IP Multimedia Session Handling*, 3GPP TS 23.218

[7] 3rd Generation Partnership Project, Technical Specification Group Services and System Aspects, *IP Multimedia Subsystem (IMS)*, 3GPP TS 23.228

[8] N. Blum, K. Knuettel, T. Magedanz, *Convergence in Services, Media and Networks – Basic requirements for Virtual Network Operators*, Fraunhofer FOKUS / Technical University of Berlin, Germany, International Conference on Intelligent Networks, May 29th – June 1st 2006

How to Achieve and Measure the Benefits of Fault Tolerant Production Infrastructures

Emmanouil Serrelis, Nikos Alexandris
Department of Informatics,
University of Piraeus,
80 Karaoli & Dimitriou,
18534, Piraeus, Greece
serrelis@unipi.gr, alexandr@unipi.gr

Abstract

Disaster Recovery Infrastructures, which have become a common aspect of all major IT infrastructures, could transform to Fault Tolerant Infrastructures in order to increase productivity, effectiveness and availability. This paper suggests a methodology for the transformation of High Availability Systems on which Disaster Recovery Infrastructures are based, to Fault Tolerant Production Infrastructures and establishes some Key Performance Indicators (KPIs) as a means to measure the effectiveness of the approach, adopting the principles of the Information Technology Infrastructure Library (ITIL) framework to a cost cutting, ecological and security-aware environment.

Keywords: *Fault Tolerance, Disaster Recovery, Availability, Change Management, Key Performance Indicators, ITIL*

1. Introduction

Events, such as the recent global economic crisis, have stressed the need to reduce the expenses in every aspect of the effected organizations, including IT expenses. Additionally, Green IT has become equally important being nowadays a major strategic objective. As stated in "Gartner's 10 Strategic Trends for 2009" *"For IT, green is everything, and that includes anything that can help cut the energy bill and reduce fuel use."* [1].

Towards this direction, IT should consider, among others, the change of its existing infrastructures and services. Some of the most eminent expenses of IT-related infrastructures are the costs related to Disaster Recovery Infrastructures. How could these infrastructures be optimized in both operational and financial terms? What could be the effect of transforming a disaster recovery infrastructure to a more cost-effective infrastructure? These are the basic questions this paper addresses.

In a "disaster avoidance" rather than a "disaster recovery" approach, the high availability solutions aim to proactively protect business continuity by monitoring the key business functions and mission critical applications that are predetermined as business priorities. In a situation where an IT component fails, it can be dealt (manually or automatically) well before its failure impacts the business. Designing IT system components with the ability to remain operational in the event of a failure has an additional benefit; that is to increase IT efficiency through continuous architecture. Moreover, such "preserve and protect" measures can facilitate maintenance projects when malfunctioning or low performing components can be upgraded or repaired during a planned downtime.

The current paper, based on [2], suggests the transformation of the existing "cold-standby" Disaster Recovery Infrastructures, based on High Availability Systems, to Fault Tolerant Production Infrastructures, presenting the various differences of the two approaches. Using the theory of change management adding the necessary technical aspects, a specialized transformation strategy is proposed. The results of both the transformation methodology and the the adoption of Fault Tolerant Production Infrastructures are examined using the concepts of Information Technology Infrastructure Library (ITIL) framework and especially through the use of Key Performance Indicators (KPIs).

The remainder of this paper is organized as follows. Section 2 gives an overview of the technical and terms related to availability. Section 3 highlights the benefits of migrating to a Fault Tolerant Infrastructure, whereas Section 4 introduces some basic principles of transition strategies. The proposed transformation strategy is presented in Section 5 and the measurement methodology of the related benefits is addressed in

Section 6. This approach is critically evaluated in Section 7 and the paper concludes in Section 8.

2. Availability terms

Currently, there are several approaches for developing Disaster Recovery Infrastructures. All of them aim to protect organizations' most valuable assets: Data and Services. In order to be able to understand the differences between them, it is essential to define several terms related to availability.

Production infrastructures should be distinguished from Backup (or Disaster Recovery) infrastructures. The **Production infrastructures** aim to serve all daily services of the organization, whereas **Backup infrastructures** operate only if a disaster occurs. This fact would classify Backup infrastructures as a rather luxury solution which would justify their existence only in the case of an extreme, disastrous event.

Availability is the proportion of time that an application can be used for productive work, measured against the time that it must be functional. The time that the application must be functional or available to users is called "mission time," which may be quite different than 7 days per week - 24 hours per day (24x7) or 5 days per week - 8 hours per day (usual working hours) [3].

There are two factors that determine application availability. The first is the reliability of the components that comprise the application: namely, how often any of the consisting components fail. The second is how long it takes for the application to be restored once a failure has occurred. The components that comprise an application minimally include the server hardware, operating system and the application itself. The application may also be dependent on data storage devices, network access devices, databases, file systems, and other hardware and software components. The amount of time it takes to bring up an application after a failure depends on what it was that caused the application to fail. This time period is called Recovery Time Objective (RTO). If the application itself failed, recovery may be as simple as restarting the application on the same system. If, on the other hand, the application has failed due to a hardware failure, recovery can take a significantly longer time since it could involve [3]:

- Notifying the service provider of the failure
- Waiting for the arrival of the service technician
- Determining what component failed
- Replacing the failed component
- Rebooting the operating system
- Recovering the file system
- Recovering the database
- Restarting the networking software
- Restarting the application

Fault tolerance differs from high availability by providing additional resources that allow an application to "ride through" a failure without interruption [3]. Many of the high-availability solutions on the market today actually provide fault tolerance for a particular application component. Disk mirroring, where there are two disk drives with identical copies of the data, is an example of a fault-tolerant component. If one of the disk drives fails, there is another copy of the data that is instantly available so that the application can continue execution. However, once such a failure occurs, the system becomes vulnerable to the failure of the single remaining disk drive, which now has the only copy of the data and represents a single point of failure. Action should be taken as soon as possible to create a mirror of the remaining disk drive. However, this process may have a negative impact on system performance, depending on where the processing to re-mirror the drive takes place.

A fully fault-tolerant solution requires that all the resources that the application is dependent on are replicated, including the application process itself. This requires an independent processor (not part of the same – probably – symmetrical multiprocessing system) and a copy of the memory that the application uses. In the worst-case failure scenario, one in which the processor or memory fails, the replicated version of the application continues to execute. Other failures simply require the application to use alternate resources (disk drives, disk adapters, communications devices). As a result of this complete hardware and process replication, fault-tolerant systems are significantly more expensive than highly available systems. A fault-tolerant system would be used in a situation where no downtime can be tolerated at all, such as an air-traffic-control system, an emergency-response system or financial trading systems (during trade hours).

In evaluating a fault-tolerant system, particular attention should be paid to the repair process. While the system may be capable of proper operation through a failure, to ensure that a subsequent failure will not bring the system down, the failed component must be immediately repaired.

Load balancing is a technique (usually performed by special software or hardware mechanisms called load balancers) to distribute work between many computers, processes, hard disks or other resources in order to get optimal resource utilization and decrease computing time. It is also the ability to make several servers participate in the same service, performing the same tasks or supporting the same service [4]. Load

balancing can also be used to increase the capacity of a server farm beyond that of a single server.

This technique is seen as complementary of fault tolerant services since it frequently provides the ability to maintain unaffected services during a certain predefined number of simultaneous failures [5]. Also, traditional implementations of fault-tolerant platforms often involve duplicate proprietary hardware and software with complex binding and mechanisms. This causes higher implementation costs and longer periods of inactivity, which could not make such solutions attractive to short term investment and productivity management. The challenge is to provide fault tolerant infrastructures that would contribute to the daily business operations as well as to the failure or disaster situations.

3. The benefits of fault tolerant approach

Having examined the background information of availability, fault tolerance as well as the load balancing techniques, it is necessary to present the benefits of migrating to a Fault Tolerant Infrastructure that could be used for production purposes as well.

One of the fundamental advantages of High Availability Systems that are based on load balancing techniques is the protection of systems operation. In addition to that their presence can vastly improve the overall performance. "*Capacity on demand, load balancing, offline maintenance capacity and zero-point backup windows are all examples of the added value [that] a continuous architecture can produce*". [6]

And where there is added value, there could be Return On Investment (ROI). Still, the quantification of ROI in that situation is not straightforward. Increases in efficiency - unless they result tangible savings like staff reductions or other avoided bottom line expenses - are often elusive to measurement. Nevertheless, they should be examined for any possible ROI contribution.

As the frequency of planned downtime is rapidly escalating due to the increasing number of applications development and the corresponding increase in upgrades and patches, the need to compress the downtime as much as possible has become even more pressing. For some companies, downtimes or even slow downs of 5-10 minutes can have a substantial affect on revenues. [7]

Other sources, such as [8], have shown that there can be a parallel use of a segment of a disaster recovery infrastructure in order to tackle extreme attacks. Expanding this idea, the benefits described above are multiplied when segments of the primary site work together their equivalent ones in the disaster recovery infrastructure, in load balancing mode.

It is, therefore, evident that organizations have more that one reasons to transform their existing implementations and select the fault tolerant production infrastructure solution. The very same tools that are used for high availability such as clustering, volume management and load balancing, can automate key procedures that would decrease the length of the downtime window as well as the cost of downtime administration. Savings from these types of value-adding features are very real and can help reduce or eliminate costs associated with planned downtime. In

	High Availability	Fault Tolerance	Fault Tolerant Production Infrastructures
Purpose / Impact	To enable faster recovery of lost data and stalled business operations in the event of a disaster.	To proactively avoid some types of disasters before they occur.	Proactively avoid most types of disasters before they occur. Increase the productivity of the organisation's IT infrastructures.
Cost	Tangible IT investment.	Tangible IT investment. ROI can be measured in most of the cases.	Tangible IT investment with measurable ROI.
Benefits	Faster time to recovery, lower lost revenues/productivity, reduced recovery costs.	Reduced probability of disaster occurrence, improved operational efficiencies, reduced planned downtime windows and costs.	Minimal probability of disaster occurrence, improved operational efficiencies, reduced planned downtime windows and costs.
Return On Investment (ROI)	Soft since the benefits are only realized in the event of a disaster.	- Reduction of disaster probability is soft. - Reduced planned downtime generates real savings in IT costs through automation of procedures that can reduce the need for IT resources, eliminate human error and save in lost business from shorter downtime.	- Reduction of disaster probability is soft. - Improved operational productivity can have direct impact on revenues and expenses and could contribute tangible cash through sales and savings. - Reduced planned downtime generates real savings in IT costs through automation of procedures that can reduce the need for IT resources, eliminate human error and save in lost business from shorter downtime. These are hard, tangible benefits driven by avoided revenue loss and reduced operational expenses.

Table 1 – Comparison of Availability Solutions

addition, the outage window is compressed so that business functions can continue with little or no interruption. Table 1 concentrates the above points.

As far as the environmental requirements and international directives are concerned, Fault Tolerant Production Infrastructures could greatly contribute to Green IT by reducing the power demands needed for operation to one site distributing power (as well as the related CO2 emissions) to multiple geographically dispersed IT sites which were consuming power anyway.

4. Transition strategies principles

Before presenting any transition strategies, some basic transition questions should be asked:

- How the transition should be planned and implemented?
- Which parts of the organization should be integrated into the fault tolerant production infrastructure solution?
- Who should be involved in the transition project?
- What this transition will cost in terms of money? Will the final outcome worth the transition costs?
- When is the right time to perform such a transition?

4.1 The change management theory

"*The concept of **change management** describes a structured approach to transitions in individuals, teams, organizations and societies that moves the target from a current state to a desired state*" [9]. This is exactly the situation one deals when transforming one IT Infrastructure to another, so it is considered very useful to see which points are suitable and applicable the situations presented in previous paragraphs. There are several theories regarding change management. The most popular ones are presented in [10] and [11]. However, as [12] points out the first question of someone diagnosing a problem is "what changed?" With a change management process in place, that question is far easier to answer. Change management is a process made up of people, software, and procedures. When properly followed the process results in many benefits including increased staff efficiency, reduced server and network device downtime and reduced Mean Time To Recover (MTTR). Change management can also bring about positive impacts on security, providing trusted audit data and increased control over ad-hoc changes, all of which lead to reduced IT costs.

Change management is critical for maintaining highly reliable systems that meet the defined service levels of the organization. To this end, best practice organizations are pushing all changes back into the build and test phases such that only rare emergency changes are actually performed on production systems. The whole network device change process must become formalized and incorporate security, testing, and documentation. The organization must ensure that appropriate preventive, detective and corrective controls exist in order to meet the challenges of regulatory frameworks, such as SOX, as well as to improve operational efficiencies.

Forrester's "Best Practices For Infrastructure Change Management: Regain Control of Runaway IT Infrastructures,"[13] boldly states "In *IT, change is an engine of progress, as well as a source of doom... While application software change control is a relatively mature process, many organizations implement infrastructure change manually, relying primarily on the IT staff's knowledge and expertise. This ad hoc process is nearing its limits in today's complex environment, where the risks inherent to changes multiply*".

Automating the change management process means addressing the six steps in an effective change management process:

1. A change is requested
2. Requested changes are reviewed, the impact assessed, and resources estimated and assigned
3. Changes are either approved or rejected
4. If approved, changes are developed and tested in a preproduction environment
5. Changes are implemented into production
6. Changes are verified and reconciled by someone else in the organization

The last step is the critical missing piece in most organizations. In order to effectively manage change, it is needed to complete the change process circle. This can be done by conducting a final verification that the requested change was implemented properly, verifying that change was implemented on all target systems and finally to have the ability to see if the change control process was bypassed. Without this step, the change management loop remains open ended, and it impossible to tell the difference between authorized, successful changes and unauthorized (or unsuccessful) changes.

The results are in and the experts agree that reducing service outages from human error through automated processes provide IT savings and a more efficient business. Eighty percent of IT budgets is used to maintain the status quo. By implementing

enforceable change management process, IT gains control of the infrastructure. By gaining visibility in what changed, IT closes the loop on change management and improves availability, improves audit performance, and lowers IT operational costs.

4.2 The technical experience

This section includes industrial experience as referenced in [14] and [15]. In today's IT infrastructures, applications are interrelated and integrated with others more than ever. At the same time, shared infrastructure elements are more common, while managing a maintenance window for each application can be exceedingly complex. However, a common maintenance window for infrastructure activity can be beneficial.

The technical experience of the current status, as highlighted above, has taught some basic lessons. The first one is that an organization should always aim to reduce unplanned downtime, since it costs on money and reputation.

The second lesson comes from [13] which states that "*80% or more of unplanned downtime is the result of People and Processes, not hardware or O/S failures*". This means that this percentage is caused things like data corruption, application failures, software failures, errors in configurations, scheduling errors, operator errors, delayed batch jobs etc. So, in order to deal with these causes of downtime, an organization should provide funds and time in people (Proper stuffing and training), problem management, event management, job scheduling, test and time recovery scenarios (in the form of production readiness reviews), Application and capacity planning and last, change management which is the area that is discussed in this paper.

The third lesson is more technology-related and mentions that an organization should minimize single points of failure, take care of environmental, facilities and network threats, make use of load balancers, redundant dispatchers, replication, cloning, RAID technologies, such as mirroring, striping and hot swap availability. Additionally an organization should plan to operate using High Availability, or even better Fault Tolerant solutions with clustering and auto fail over capabilities.

In order to implement infrastructures that could deal with the issues above and make use of the technologies mentioned, the organization should understand the application architecture and constraints as well as to understand all application dependencies and interrelationships to needed components, whereas they should reduce any batch interference (delays, lockups etc.).

Furthermore, they should manage other planned changes, by developing suitable infrastructure and facility work and performing appropriate hardware, operating system, database, application changes and upgrades. Another need that should be covered is the need for proper infrastructure test environments. Within this framework the organization should aim to common maintenance windows, expecting increased coordination as well as staff overhead.

Taking all these lessons into account an organization should try to follow the following rules within the plans for implementing Fault Tolerant Production Infrastructures. Firstly, they must integrate application availability in their design, since this can be hardly be improved in later phases. Secondly, there should be a well planned transaction queuing as well as a highly optimized batch processing. The third rule is to set the requirements for scheduling and availability early in the design phase. Fourthly, an organization should choose to serve only business-critical functions with high-cost Fault Tolerant infrastructures, having in mind that these kind of infrastructures cost about 3.5 times as much as a standard infrastructure. [16]

5. The proposed transformation strategy

Taking into account all above sources, the **proposed transition strategy** combines the change management theory and the technical experience. There are seven phases to complete the transformation from the High Availability Standby Systems to the Fault Tolerant Production Infrastructures. These are:

Phase 1: Definition of the transformation scope
Phase 2: Categorization of System groups
Phase 3: Application Analysis
Phase 4: Process Analysis
Phase 5: Cost Analysis
Phase 6: Business Decision
Phase 7: Execution of Transformation

5.1 Definition of the transformation scope

As can it be easily understood, a problem well defined is a problem that can be solved more easily. During the first phase of the transformation, the organization should decide which systems are candidates to transform. Thus, for each application area, it should be determined what the transformation scope is, with the correct user representative(s). At the same time, the schedule goal and the availability goal should also be agreed. Since it is more costly to re-change any infrastructures, it is important to determine and design

schedule and availability up front, just like any other application functional requirement.

5.2 Categorization of System groups

The second phase aims to categorize the system groups. For example an organization could distinguish between Business Support Systems, Operational Support Systems, Self Service / e-Commerce, Management Support Systems. This categorization will give the organization a rough idea on how these systems should be implemented in terms of availability, enriching the decisions taken in the first phase.

5.3 Application Analysis

During the third phase, the organization should understand each application's architecture, special constraints, "release tolerance" and flexibility to change. Additionally, the applications dependencies on other applications and components should be gathered, along with architecture diagrams and data flows. Finally, decisions on the whether the applications' modification for Fault Tolerance should be in-house or outsourced should be made.

5.4 Process Analysis

In this phase questions such what is the current Standing Operating and Testing Procedure should be answered with respect to technology. The current availability of each function/application should also be identified. Furthermore, what can the organization expect with existing budget. In order to answer these questions more easily, metrics related to availability, efficiency and performance have to be established. The Final of this phase is to identify root causes of unplanned downtime.

5.5 Cost Analysis

The most important phases are phases 5 and 6. This is where is actual decision on whether the transformation should be executed or not is taken. In the cost analysis phase, the basic question that the executive level will pose is what improvements can the organization make from existing budget. In order to answer this properly, the organization has to consider to invest in the right areas to expand schedule and availability. Additionally, the organization has to know costs to expand schedule beyond baseline to meet goals as well as the costs to increase availability beyond baseline to meet goals. At this phase involvement from all areas of the organization should be encouraged.

5.6 Business Decision

This is the last phase before the actual execution of the transformation. During this phase, the organization should develop a consistent approach to weigh the business benefits against the cost, while maintaining focus on the business problem, which is to increase the availability and the usability of its systems. Towards that decision, a Steering Committee or the business owners of the applications need to determine the business need. Since it is difficult to cost and plan for applications individually an accurate categorization would be very useful. At all times, the decision committee should be aware of the transformation sponsor capabilities and wills that would also be effected by any potential future expenses that a Fault Tolerant Production Infrastructure may imply.

5.7 Transformation Execution

The final phase of the suggested transformation strategy is actual execution of the transformation. In order to achieve this, the organization, and especially the people involved and affected, should be committed to the project. A detailed and realistic definition of the resources in terms of people and budget is necessary. Another very important issue is to define the owner of the new infrastructure in order to establish a common communication point that could manage, adjust, develop, document the transformation plan, with goals, activities, responsibilities, dates, etc. Finally, the organization should measure the actual benefits against the initial goal, for use in future or parallel transformation projects.

6. Measuring the benefits

The application of the approach, as described above, has been demonstrated in past [17], resulting a rather successful outcome. However, as pointed out in that attempt, in order to provide more concrete evidence of the applicability of the methodology, some formal metrics of the methodology should be established. These metrics should enable consistent measurement of resources, time and cost.

Towards that direction, the Information Technology Infrastructure Library (ITIL) framework has been examined for suitability. ITIL is a globally accepted set of best practices used for the management of IT environments. In order to improve the level of IT services provided in an organization, the ITIL

framework suggests the adoption and combination of methodologies, tools, metrics and roles.

As it can be understood, the adaption of ITILs metrics to serve the needs of the transformation methodology described above, would strongly support the applicability of the methodology. Additionally, such an adaptation could also provide a known interface for people who are familiar with the ITIL framework as well as ITIL's measuring tools.

The following paragraph presents the foundations for the application of ITIL-based metrics to the transformation approach.

6.1 Key Performance Indicators (KPIs)

In ITIL terminology, KPIs are "*financial and non-financial metrics which help organizations to define and measure progress toward organizational goals*" [18]. KPIs main goal is to review the current state of an organization and provide the basis for the prescription of a course of improving actions. In order to obtain a more solid view of the organization's state, KPIs should be monitored in real-time, a process otherwise known as Business Activity Monitoring (BAM). Common uses of KPIs include the measurement of intangible benefits or values such as leadership development, engagement level, service delivery, and satisfaction rates. Being able to grasp such aspects, managers typically tie KPIs to organization's strategic management.

The selected KPIs may differ depending on the nature of the organization and the organization's objectives. In any case, their proper usage could assist an organization to measure progress towards their organizational goals, especially goals which include difficult to quantify knowledge-based processes.

Any KPI is a part of a "measurable objective" which is made up of a direction, KPI, benchmark, target and time frame. For example: "Increase Average Storage Utilization per Server from 20% to 60% by the end of the year 2010". In this case, "Average Storage Utilization per Server" is the KPI.

KPIs should not be confused with Marketing-related Critical Success Factors. For the example above, a critical success factor would be something that needs to be in place to achieve that objective; for the previous example, a file archiving software tool.

Performance indicators should also differ from business drivers & aims (or goals). A financial institution may consider the "increase rate of deposits" as a Key Performance Indicator which might help the institution understand its position in the market, whereas a telecommunications company could consider the "percentage of successful call attempts from its customers" as a potential Market-related KPI.

Nevertheless it is necessary for an organization to at least identify its KPIs. The basic rules for identifying KPIs are:

- To have pre-defined business processes.
- To have clear requirements for the aims and the performance for the business processes.
- To have a measurement that could quantify and qualify its results and compare these with the previously set goals.
- To examine the variances and adjust any processes or resources needed to achieve short-term goals.

The definition of any KPI should apply all of the following characteristics:

- Specific, so that it should not be confusing with other KPIs
- Measurable, so that it should be feasible to measure it or calculate using a specific measurement unit
- Achievable, so that it should be easy to obtain the necessary information
- Relevant, so that it should directly connect to the business objective
- Time-bound, so that it should take into account time constrains, in order to be able to tackle any issues related to time depended results and filter them.

Key Performance Indicators in practical and strategic development terms are objectives to be targeted that will add value to the business.

Having seen how KPIs are defined and used within a generic organization, it is now possible to use these principles within an IT infrastructure environment, where some more specific KPIs could be defined. [19] These KPIs will be used to represent the benefits from the adoption of Fault Tolerant Production Infrastructures as well as the benefits from the usage of the suggested transformation approach.

6.2 Generic KPIs for the adoption of Fault Tolerant Production Infrastructures

Although the transformation approach is quite specific as far as the transformation steps are concerned, each IT infrastructure involves different modules, processes and systems. Thus, the KPIs chosen to be presented in this paper could only be considered as a first, generic, set of KPIs. Additional KPIs can and should be considered in order to match the specific needs of an organization. However, the total number of the KPIs used to measure the success of the adoption of Fault Tolerant Production

Infrastructures should not be too large since this may affect the performance levels of the infrastructure.

The generic set of KPIs for measuring the success of the adoption of Fault Tolerant Production Infrastructures could be divided into the technical and business related KPIs. These two KPI categories are not are not directly related to each other. They are aim to point up different aspects of the Fault Tolerant Production Infrastructures and measure the technical and business benefits of its adoption. It should be made clear that there is no need for conciliation, combination or synchronization between the results of these two categories.

Nevertheless, the measurement of the following KPIs should take place before and after the transformation, so that the comparison can confirm the benefits of the Fault Tolerant Production Infrastructures.

Technical KPIs

1. **Usable Storage in IT Site(s)**: This KPI is the storage that can be used to store data in an IT site after any technical overhead, such as RAID configurations of storage boxes. The IT site can be Primary, Secondary, Other, Disaster Recovery or any IT Site. The measuring unit is in MBs or GBs.
2. **Average utilization of total Processing Power capacity Average in IT Site(s)**: This KPI is the average percentage of utilization of processing power of all systems in a specific (or all) IT Site(s) during the measurement period. The IT site can be Primary, Secondary, Other, Disaster Recovery or any IT Site. The measuring unit of utilization is a percentage. The measuring unit of processing power is Million Instructions per Second.
3. **Average network throughput between servers and clients**: The term "Throughput" refers to the performance of data transmission, and is measured by characters actually transmitted or received during a certain period of time. Throughput is usually measured in bps (bits per second). A better (higher) throughput to the clients could signify the existence of a better infrastructure.
4. **Average Disks I/O in central storage in IT Site(s)**: This particular KPI reveals the average percentage of storage disks utilization of all systems in a specific (or all) IT Site(s) during the measurement period. The IT site can be Primary, Secondary, Other, Disaster Recovery or any IT Site. The measuring unit is bps (bits per second).
5. **Average Memory utilization in in IT Site(s)**: The memory utilization expose the average percentage of memory utilization of all systems in a specific (or all) IT Site(s) during the measurement period. The IT site can be Primary, Secondary, Other, Disaster Recovery or any IT Site. The measuring unit is a percentage.
6. **IT Site power usage effectiveness**: This KPI is calculated by dividing the total power usage of an IT Site by the power usage of IT equipment (computer, storage, and network equipment as well as switches, monitors, and workstations to control the IT Site). The IT site can be Primary, Secondary, Other, Disaster Recovery or any IT Site.
7. **Systems Footprint in IT Site(s)**: The footprint represents the physical area that the systems occupy and is measured in square meters or square feet. A change in the measurement of this KPI would support the benefits earned by the adoption of the Fault Tolerant Production Infrastructures. Similarly to other KPIs the IT site can be Primary, Secondary, Other, Disaster Recovery or any IT Site.
8. **% of production servers located in Primary / Secondary IT Site(s)**: This is one of the most profound benefits of Fault Tolerant Production Infrastructures. It appears as a percentage of production servers located in a particular IT Site (Primary or Secondary) over the total number of servers in all IT Sites.

Business KPIs

1. **Planned Downtime of offered business services**: Planned downtime is downtime of any business service caused by scheduled for system or application maintenance. It is measured in minutes or hours per year.
2. **Unplanned Downtime of offered business services**: This is the amount of downtime of any business service arising from reasons other than maintenance. It is measured in minutes or hours per year.
3. **Recovery time of business critical services**: This KPI presumes that there has been decided which are the business critical services. The recovery time is the duration of time within which the business critical services can be restored after a disaster in order to avoid unacceptable business consequences. It is measured in minutes, hours or days.
4. **Operational Expenses of IT Division**: The Operational Expenses, measured in any currency, are the yearly running costs of any organization, or in parts of the organization, such as IT Division. A decrease in these could signify a better usage or management of the existing resources.
5. **Capital Expenses of IT Division**: Opposite to the previous KPI, Capital Expenses are the one-off costs of products and non-consumable parts. It is measured in any currency and could relate to the financial benefits

of the adoption of Fault Tolerant Production Infrastructures.

6. **Cost of Recovery of new business services**: This is a very important KPI since it could depict the low cost expansion capabilities of Fault Tolerant Production Infrastructures. The measuring unit is any currency.

7. **Satisfaction rate by IT staff (System owners)**: This is a qualitative measurement of the satisfaction of the IT staff. The staff's satisfaction rate could be based on periodic surveys of employees after a reasonable period of infrastructure maturity time such as 6 months. The maturity time could minimize non in type negative reactions, caused by staff's natural resistance to change [20]. This KPI is measured as a percentage of positive reactions.

8. **Percentage of satisfaction by Business staff (Business owners)**: In the same way as in the previous KPI, this measurement is related to the satisfaction of the Business Staff which may (or may not) have a different opinion on the benefits of the implemented infrastructure. This KPI is measured as a percentage of positive reactions.

9. **Average frequency of updates of disaster recovery plans**: This KPI should portray staff awareness on the updating the Disaster Recovery plans. Since the Fault Tolerant Production Infrastructures amplify the role of Disaster Recovery IT Sites, it should be expected that this update frequency should be increased. It is measured in days.

10. **% of growth of IT budget**: An unusual growth of the IT budget may entail some form of relation to the new Infrastructure architecture.

11. **New Systems Procurement rate (as % of existing systems)**: This KPI should confirm that the procurement of new systems should be less frequent since extra resources and capacities would be freed up (mainly in the Primary IT Site) after such a transformation.

12. **Average time to provision new systems**: This is the average time needed to provide a new system to an application or system owner. The time starts counting when the request is send and ends when the system is handed over. The measurement time is in minutes, hours or days. A more dynamic infrastructure, as Fault Tolerant Production Infrastructures aim to be, should decrease that time.

13. **Average time to provision new business services**: This KPI differs from the previous one for the reason that it also includes processes and people needed to provide the new business services. It is measured in minutes, hours or days.

6.3 KPIs for the usage of the suggested transformation methodology

The KPIs that could be used for measuring the benefits from using the suggested approach are less dependent on the IT infrastructure and business services of the organization that has chosen to use this approach, than the KPIs described in the previous paragraph. Again, the number of the selected KPIs should be limited to a level that would not effect the actual progress and effectiveness of the methodology.

Since the core of the transformation approach is a change management set of processes, the consequent KPIs for measuring the success of the suggested transformation methodology are solely related to project management metrics. The measurement of the following KPIs should take place during the transformation, and be compared to similar projects that have been (or will use) different transformation methodologies. These projects may also originate from outside the implementing organization.

Project Management KPIs

1. **Return on the transformation process Investment**: This KPI illustrates the main idea behind this paper. It is a predominantly hard KPI to measure since the actual Return cannot be directly calculated. However all other KPIs mentioned in this paragraph could be used as input to its calculation. It could be measured in any currency or in time units such as days, weeks or months. When measured in time units the Return represents the time gained by using the proposed transformation methodology.

2. **Total Time of transformation process**: This is the time period the transformation project runs and includes all seven phases of the proposed transition strategy described in paragraph 5. It is measured in days, weeks or months.

3. **Utilization rate of human resource for project purposes**: This is the percentage of the time that a worker will dedicate to the transformation project in relation to its total time. It is similar to Full-time equivalent (FTE) which is a way to measure a worker's involvement in a project and is used by many organizations worldwide.

4. **Downtime of Business Services due to transformation project**: Some of the transformation phases described before could effect the operation of some Business Services and thus their availability. Less production outage time for each Business Service means less lost income of the organization and more value for the transformation methodology. It is measured in minutes, hours or days.

5. **Total cost of transformation project**: This is very important since the transformation project should be significantly less expensive than the expected earnings. It is measured in any currency.
6. **Number of people involved in transformation project**: This is also important in order to be able to appreciate the staffing needs of the project.
7. **Percentage of administrative activities related to the transformation project**: This is a project management quality KPI. It presents the number of administrative activities for the transformation project in relation to the total activities of the project which also include implementation activities. It is measured as a percentage.
8. **Budgeted Cost of Work Scheduled**: This is the sum of the budgets of the activities that were planned or scheduled to be completed, otherwise known as "planned value". It is measured in any currency.
9. **Budgeted Cost of Work Performed**: This KPI, measured in any currency, is the planned or scheduled cost of activities that were completed, also known as "earned value". It is measured in any currency.
10. **Actual Cost of Work Performed**: It is the sum of actual costs of activities that are completed. It is measured in any currency.
11. **Schedule Performance Index**: This is calculated by the use of the previous KPIs. It is the division of "Budgeted Cost of Work Performed" by the "Budgeted Cost of Work Scheduled".
12. **Cost Performance Index**: This is calculated by combining two of the previous KPIs. It is the "Budgeted Cost of Work Performed" divided by the "Actual cost of Work Performed".
12. **Cost Schedule Index**: This is the "Cost Performance Index" multiplied by the "Schedule Performance Index". The Cost Schedule Index measures the likelihood of recovery for any project that is late and/or over budget. The closer the index is to 1, the more likely the project's can be recovered from it's deviation to the original baseline. This can be useful for any organization that would decide to apply the proposed transformation methodology.
13. **% of time coordinating project**: This is an efficiency related KPI for the methodology and is represented as a percentage of time (in man hours) used to coordinate project relative to over the total time used to implement (and coordinate) the project.
14. **% of milestones missed**: Percentage of milestones recorded in all processes and phases as missed.
15. **Number of incidents due to transformation project**: In theory the transformation like any other planned change should not cause any incidents. However, a more practical evaluation of the methodology should also measure also the number of incidents caused by the methodology in relation to the total number of incidents. In any case the changes should not cause more than a upper percentage of all the incidents.
16. **Average rework per phase after implementation of each phase**: This is a significant measurement of the quality of the Analysis and Design processes that is methodology involves. If the rework per phase is low then this could be an indication that the methodology is providing a solid foundation for the transformation to Fault Tolerant Production Infrastructures. It is measured in man-days, man-weeks or man-months.

7. Evaluation and Future Improvements

The suggested transformation is not a new idea. However the actual application of the change management theory to the specific transformation tasks which are based more on practical Management experience than Information Technology theory is a new addition to the arsenal of an IT manager.

The presented benefits range from low-level technical benefits to high level financial benefits as well as the contribution to Green IT Infrastructures.

The theory has been supported by establishing some ITIL-based metrics (KPIs) in order to challenge and prove its applicability. These metrics aim to measure, test and evaluate practically the proposal in a formal, accurate and consistent manner.

Using KPIs, such as the ones proposed, the IT managers are able to evaluate the outcomes of the transformation of High Availability Systems to Fault Tolerant Production Infrastructures. Furthermore, the transformation methodology itself can also be evaluated in terms of technical and business benefits.

As a more general remark, it should be pointed out that the transformation methodology can also be seen as part of ITIL's set of concepts and policies for managing change in infrastructures and services.

Following that way of thinking, future research should include a more thorough analysis of the relationship with ITIL's change management process, aligning the transformation of High Availability Systems to Fault Tolerant Production Infrastructures with the related ITIL's core components, namely the Service Strategy, Service Design and Service Transition.

A more detailed analysis of the selected KPIs and their usage can also offer more information on the prospective users of the methodology. Usage-related factors could include supplementary information on the measuring time, measuring frequency as well as number of measuring repetitions.

There are also other extensions to the proposed methodology, which can enforce the relationship with

ITIL's practices. These extensions might engage the use of Balanced Scorecard as well as the adjustment of IT organization Service Catalog. The Balanced Scorecard suggests that an organization should be viewed from four different perspectives (the Learning & Growth Perspective, the Business Process Perspective, the Customer Perspective and the Financial Perspective). Additionally, the Balanced Scorecard suggests the development of some other metrics, the collection of related data and the proper analysis of the perspectives' relations. The Service Catalog is a list of services that an organization provides to its employees or customers. Each service within the catalog may well include:

- A description of the service
- Timeframes or service level agreement for fulfilling the service
- Who is entitled to request/view the service
- Costs (if any)
- How to fulfill the service

Yet, this paper is to be perceived as a packaged proposal that includes a proposition for the target infrastructure, the transformation methodology as well as the metrics for the efficiency of both the infrastructure and the methodology.

8. Conclusion

This paper has suggested the transformation of the existing "cold-standby" Disaster Recovery Infrastructures, based on High Availability Systems, to Fault Tolerant Production Infrastructures. The various differences of the two approaches have been presented and there are clear indications that organizations can benefit from transforming their existing implementations and selecting the Fault Tolerant Production Infrastructure solution.

The business and technical results of the transformation as well as the effectiveness of the methodology are measured through the use of relevant KPIs using the ITIL framework. Savings from these types of value-adding features vary from case to case but the use of this methodology makes possible the reduction or elimination of costs associated with planned (and most unplanned) downtime. In addition, the outage window is compressed so that business functions can continue with little or no interruption. The suggested transformation strategy has used the theory of change management adding several technical aspects. This transformation strategy can be considered as a strong support tool in order to make the transformation less costly, less time consuming as well as to effectively integrate people, systems and procedures.

9. References

[1] P. Thibodeau, "Virtualization Tops Gartner's 10 Strategic Technologies for 2009", *Computerworld*, http://www.cio.com/article/print/454906

[2] Em. Serrelis, N. Alexandris, "From High Availability Systems to Fault Tolerant Infrastructures", *IEEE Computer Society Press*, ICNS, Athens, 2007

[3] "High Availability: A perspective", *Gartner Research*, ID Number: DPRO-90193

[4] W. Tarreau, "Making applications scalable with Load Balancing", http://1wt.eu/articles/2006_lb/index.html

[5] "Highly Available Embedded Computer Platforms Become Reality", *International Engineering Consortium*, http://www.iec.org/online/tutorials/ha_embed/topic01.html

[6] K. Miller, "Don't Recover-Failover", *DM Direct*, Oct 2004

[7] S. Atwood, "Planned Downtime", *DM Direct*, Veritas Software, October 2004

[8] Em. Serrelis, N. Alexandris, "Disaster Recovery Sites as a Tool of Managing Extreme Attacks", *IEEE Computer Society Press*, ICISP, Cap Esterel, 2006.

[9] "Change Management", *Wikipedia*, http://en.wikipedia.org/wiki/Change_management

[10] J. Martin, "Organisational Behaviour", *Thomson Business Press*, 1998, ISBN 1-86152-180-4, pages 575-600

[11] L. J. Mullins, "Management and Organisational Behaviour", 5th Edition, *Pitman Publishing*, 1999, ISBN 0-273-63552-2, pages 821-830

[12] "Five basic principles of Change Management", http://www.teamtechnology.co.uk/changemanagement.html

[13] J. P. Garbani, "Best Practices For Infrastructure Change Management: Regain Control Of Runaway IT Infrastructures", *Forrester Research*, 25-3-2004, http://www.forrester.com/Research/Document/Excerpt/0,7211,34048,00.html

[14] "Microsoft Operations Framework 4.0", http://www.microsoft.com/technet/solutionaccelerators/cits/mo/smf/smfchgmg.mspx

[15] "Change Management (ITSM)", *Wikipedia*, http://en.wikipedia.org/wiki/Change_Management_%28ITSM%29

[16] D. Scott, Y. Natis, "Building Continuous Availability Into E-Applications", *GartnerGroup*, COM-12-1325, 29/9/2000

[17] Em. Serrelis, N. Alexandris, "Fault Tolerant Production Infrastructures in Practice", *IEEE Computer Society Press*, PIMRC, Athens, 2007

[18] F. John Reh, "Key Performance Indicators – What are Key Performance Indicators or KPI", *About.com*, http://management.about.com/cs/generalmanagement/a/keyperfindic.htm

[19] "KPI Library", http://kpilibrary.com/

[20] A. J. Schuler, "Overcoming Resistance to Change: Top Ten Reasons for Change Resistance", http://www.schulersolutions.com/resistance_to_change.html

Preliminary 2009 Conference Schedule

http://www.iaria.org/conferences.html

NetWare 2009: June 14-19, 2009 - Athens, Greece

- SENSORCOMM 2009, The Third International Conference on Sensor Technologies and Applications
- SECURWARE 2009, The Third International Conference on Emerging Security Information, Systems and Technologies
- MESH 2009, The Second International Conference on Advances in Mesh Networks
- AFIN 2009, The First International Conference on Advances in Future Internet
- DEPEND 2009, The Second International Conference on Dependability

NexComm 2009: July 19-24, 2009 - Colmar, France

- CTRQ 2009, The Second International Conference on Communication Theory, Reliability, and Quality of Service
- ICDT 2009, The Fourth International Conference on Digital Telecommunications
- SPACOMM 2009, The First International Conference on Advances in Satellite and Space Communications
- MMEDIA 2009, The First International Conferences on Advances in Multimedia

InfoWare 2009: August 25-31, 2009 – Cannes, French Riviera, France

- ICCGI 2009, The Fourth International Multi-Conference on Computing in the Global Information Technology
- ICWMC 2009, The Fifth International Conference on Wireless and Mobile Communications
- INTERNET 2009, The First International Conference on Evolving Internet

SoftNet 2009: September 20-25, 2009 - Porto, Portugal

- ICSEA 2009, The Fourth International Conference on Software Engineering Advances
  - SEDES 2009: Simpósio para Estudantes de Doutoramento em Engenharia de Software
- ICSNC 2009, The Fourth International Conference on Systems and Networks Communications
- CENTRIC 2009, The Second International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services
- VALID 2009, The First International Conference on Advances in System Testing and Validation Lifecycle
- SIMUL 2009, The First International Conference on Advances in System Simulation

NexTech 2009: October 11-16, 2009 - Sliema, Malta

- UBICOMM 2009, The Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies
- ADVCOMP 2009, The Third International Conference on Advanced Engineering Computing and Applications in Sciences
- CENICS 2009, The Second International Conference on Advances in Circuits, Electronics and Micro-electronics
- AP2PS 2009, The First International Conference on Advances in P2P Systems
- EMERGING 2009, The First International Conference on Emerging Network Intelligence
- SEMAPRO 2009, The Third International Conference on Advances in Semantic Processing