Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

February 2011 **Volume III** **Number 1** **ISSN 2061-2079**

PAPERS

NEWS

CALLS FOR PAPERS

ADDITIONAL

hte

IEEE HUNGARY SECTION

2011.hu

NIH National Innovation Office

FUTURE INTERNET ASSEMBLY

Events of the FUTURE INTERNET WEEK

• ENoLL Event	16 May, 2011	Gerbeaud House	www.fi-budapest.eu
• Internet of Things Conference	16 May, 2011	Hotel InterContinental Budapest	www.iot-budapest.eu
• Fire Concentration Day	16-17 May, 2011	Novotel Budapest Centrum Hotel	www.ict-fire.eu
• Future Internet Conference	17 May, 2011	Hotel InterContinental Budapest	www.fi-budapest.eu
• Future Internet Assembly	18-19 May, 2011	Hotel InterContinental Budapest	www.fi-budapest.eu
• Future Internet Forum	18 May, 2011	Gerbeaud House	www.fi-budapest.eu
• ceFIMS Workshop	18 May, 2011	Gerbeaud House	www.fi-budapest.eu

More information and registration: www.fi-budapest.eu

Indexing information

Infocommunications Journal is covered by INSPEC and Compendex.

The journal is supported by NCA NEMZETI CIVIL ALAPPROGRAM the National Civil Fund.

Infocommunications Journal

Technically co-sponsored by IEEE Hungary Section

Editorial Office (Subscription and Advertisements)**:**
Scientific Association for Infocommunications
H-1055 Budapest, Kossuth Lajos tér 6-8, Room: 422
Mail Address: 1372 Budapest Pf. 451. Hungary
Phone: +36 1 353 1027, Fax: +36 1 353 0451
E-mail: info@hte.hu
Web: www.hte.hu

Articles can be sent also to the following address:
Budapest University of Technology and Economics
Department of Telecommunications
Tel.: +36 1 463 3261, Fax: +36 1 463 3263
E-mail: szabo@hit.bme.hu

Subscription rates for foreign subscribers:
4 issues 50 USD, single copies 15 USD + postage

Publisher: PÉTER NAGY • Manager: ANDRÁS DANKÓ

HU ISSN 2061-2079 • Layout: MATT DTP Bt. • Printed by: FOM Media

www.infocommunications.hu

Light-trains: An Integrated Optical-Wireless Solution for High Bandwidth Applications in High-Speed Metro-Trains

(Invited Paper)

Ashwin Gumaste, Akhil Lodha, Saurabh Mehta, Jianping Wang and Nasir Ghani

Abstract: Moving trains represent a voluminous mass of users moving at high velocities that require bandwidth (on demand). A new solution is proposed that facilitates dynamic provisioning, good scalability and efficient use of available bandwidth. The proposed solution *(light-trains)* seamlessly integrates optical and wireless networking to provide broadband access to users in moving trains. The solution identifies the set of requirements – such as fast hand-off, low-cost of deployment, mature technology and ability to provide dynamic bandwidth provisioning. A protocol that enables end-to-end provisioning across the three domains of technology aka light-trails, WiMax and WiFi is also proposed. The proposed protocol is a cornerstone mechanism for providing inter-domain (technology) connectivity in a pragmatic way. Different aspects of the protocol are considered, amongst which delay and efficiency are analyzed and computed. The protocol and system requirements juxtaposed are simulated extensively. Results pertaining to utilization, delay, efficiency and network wide performance are all showcased. Viability of the model in being able to provide bandwidth to moving users is shown.

Key words: light-trails, moving trains, bandwidth-on-demand, cross medium communications.

I. INTRODUCTION

The growth and spread of the Internet and other web services into the wireless domain has facilitated ubiquity and mobility for end-users. A significant number of the end-users that make use of bandwidth intensive applications are in mass-transportation systems. In order to provide bandwidth to users within the domain of a mass transportation system, it is imperative to have a wireless access mechanism that is further bolstered by a high-speed and dynamic wired backbone solution.

Manuscript received February 6, 2011.
A. Gumaste is James R. Isaac Chair and faculty member in the Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Bombay 400076, India.
Akhil Lodha is Quantative Analyst on the Electronic Options Trading desk at Citigroup, New York.
S. Mehta is a research scholear with the Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Bombay, India.
Jianping Wang is an assistant professor in the Department of Computer Science at City University of Hong Kong.
Nasir Ghani is with the University of New Mexico (UNM), Albuquerque, NM87131 USA.
Digital Object Identifier:

The metro train represents a popular form of mass-transportation system in either its underground or over-bridge manifestation. (1) There are a large number of bandwidth hungry users within the confinement of a metro train. (2) Such a train represents a single moving entity that requires dynamic bandwidth allocation as it moves through a wireless network. (3) Further, several trains can statistically co-exist on the same track. This makes the bandwidth allocation problem along a track complex in lieu of the limited bandwidth-distance product offered by wireless. These needs imply added levels of complexity in the design of hand-off mechanisms and in provisioning the underlying "core" network to facilitate multimedia and emerging broadband applications. We propose a mechanism that enables bandwidth-on-demand to trains over a cross-medium architecture comprising of a wireless overlay supported by a metro-core optical network.

End-users within the train receive and transmit data from access points in each coach, and hence are part of a wireless LAN (IEEE 802.11 type). The access points are further connected to an Ethernet aggregation switch in the train. Provisioning the underlying physical layer which is typically an optical network taking into cognizance this moving entity poses a new design problem.

The light-train approach combines the wireless and optical layers to create an efficient solution for bandwidth allocation in trains. To do so, the light-train solution uses a novel optical technology used in conjunction with wireless end-user technologies. The approach proposed by us takes into consideration end-user profile from both a service requirement as well as equipment asset perspective. The protocol proposed ensures seamless ubiquity while adhering to the norms set by emerging networking services. Our approach has practical assumptions – there is no need to upgrade equipment at the end-users (who continue to use well-disseminated WiFi technology). Equipment is only upgraded at the network side.

This work is an extension of [12] – where we proposed the concept of bandwidth to moving trains. This work further enhances that concept providing a detailed stochastic model in addition to rigorous simulations. We propose the light-trains model as an engineering solution for providing bandwidth on demand to moving trains. In Section II, we discuss the system design of the light-trains model. Section III showcases how the legacy IP network reacts/interfaces to our light-train model. Section IV discusses the principles of the proposed integrated protocol. Section V provides for delay analysis. Section VI compares our model to other related works. Section VII discusses simulation results while Section VIII summarizes the paper.

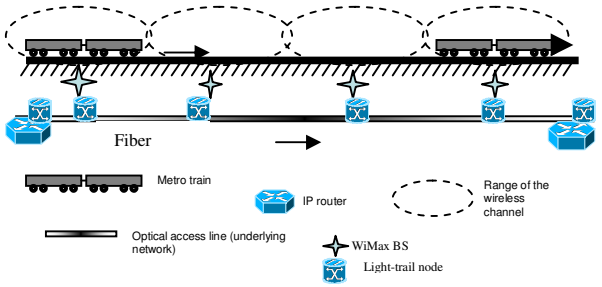


Fig. 1. Conceptual layout of metro trains over a wireless network and underlying optical network.

II. LIGHT-TRAIN SYSTEM AND DEFINITIONS

Consider a train *u* moving along a track. The system is as shown in Fig. 1.

Wireless Gateways: As the train *u* travels, it communicates with the rest of the core network through *wireless gateways* (GW) situated at intervals along the track. The gateways have a limited transmission range and a series of gateways provide bandwidth to a moving train in a concept that is similar to cellular communications. Gateways are positioned one after another separated by a distance that is computed by the transmission range (to prevent Signal-to-Noise Ratio – SNR from degrading below a certain threshold). The distance of coverage of a particular wireless gateway is defined as the *geographic range* of that gateway. In addition to geographic range, we define the concept of *provisioning zone* as the overlap area between the geographic ranges of two adjacent GWs. Trains move from one GW to another in a seamless manner thus maintaining ubiquitous communication. Power based hand-off is the typical technique used for maintaining ubiquity. Hand-off occurs in the provisioning zones. Hand-off and associated methods are defined and explained subsequently in Section IV. B.

The principle requirement on the optical network is of *dynamic provisioning of bandwidth* to wireless GWs. Data paths must be provisioned to the wireless GWs through the optical network on an on-demand basis to meet the BW and delay constraints of services. This time-sensitive requirement, coupled with the fact that multiple trains will be sharing the same track, implies that pre-provisioning will generally lead to an inefficient bandwidth solution. To enable ubiquity it is necessary that the combined optical + wireless network be able to provide *fast hand-off*.

For example cellular hand-off solutions require up to several 100s of milliseconds. Similarly concepts based on *moving tunnel* approach as shown in [8] require up to 100 milliseconds.

We propose an optically-assisted method for hand-off. The underlying principle utilizes the dual concepts of optical multicasting and out-of-band control (described later).

Another characteristic for cross-medium integration is *signaling*: wireless GWs must be able to signal to each other about ongoing communication activities when trains move through their geographic range.

To illustrate this, consider the example of a train that moves through the geographic range of one GW to the next (adjacent) GW. Assume that a user in the train is trying to download a large file. Assume that the time required to download the file exceeds the geographic range of the GW; and hence it cannot be downloaded within the range of one GW. In such a case, the file must be made available to the next adjacent GW, when the train arrives in its geographic range, so that the user in the train can successfully download the entire file. The time lost between tearing down the connection between the train and the first GW and the setting up the connection (making the file available) at the second GW is of critical importance. Another factor that also leads to an efficient cross-medium solution is that the second GW must be aware of the fraction of the file that was successfully transmitted by the first GW, so that it has to only transmit the remaining fraction. While TCP sequencing would take care of this issue, we do require a more robust and far more dynamically adjustable solution. This necessitates the need for efficient signaling both at the optical and the wireless layer amongst nodes as well as between the nodes and the train.

An optical networking solution that enables dynamic provisioning, optical multicasting and provides for effective signaling is the *light-trails* approach originally proposed in [1-4]. As opposed to conventional end-to-end circuit or lightpath (optical wavelength circuit), a light-trail is a generalization of a lightpath [5] such that multiple nodes along the path can communicate with each other. A light-trail is analogous to an optical bus with the added advantage that communication and arbitration of the bus is done through an *out-of-band* (OOB) control channel. The OOB control channel is optically dropped, electronically processed and then reinserted into the network. The differentiation between the OEO control channel and the all-optical (OOO) bus based data channel is the key to some of the functionalities of light-trail based networks. Multiple light-trails, each on a unique wavelength, use a common control channel.

In order to create a light-trail, nodes must be based on an architecture that supports three functionalities: (1) The ability to drop-and-continue incoming optical signal. (2) Ability to passively add optical signal and (3) To support the OOB control. The first two properties, that of drop and continue as well as passive addition lead to formation of the optical bus, while the third property enables efficient (dynamic) provisioning within the bus. A light-trail is defined between two extreme nodes (a start node called the convener node and a stop node called end-node) that regulate signal flow within the optical wavelength bus. Signal flow is in the direction from the convener to the end-node.

Light-trail node architecture is shown in Fig. 2. A fiber that carries composite WDM signal enters the node premises from the left-side. The composite signal is de-multiplexed into constituent wavelengths by an optical de-multiplexer (typically an Arrayed Wave-Guide – AWG). The

control channel is also extracted and electronically processed by a control card. The data channels that are de-multiplexed are fed individually to a *local access section*. This section enables the node to access (TX/RX) data on individual light-trails. To do so, the section consists of two passive couplers separated by an optical shutter. The two couplers are in 1x2 and 2x1 configurations. The first coupler (called Drop Coupler – DC) drops and continues incoming optical signal while the second coupler (called Add Coupler – AC) enables passive addition of the optical signal. The optical shutter is a slow moving switch that is in the OFF position at the convener and end-nodes of a light-trail, while is in the ON position for all the intermediate nodes. Switching is required only when light-trails are set up or torn down, which for the case of providing bandwidth to moving trains is a semi-permanent feature and seldom used.

Nodes along a light-trail communicate with each other by setting up *connections*. Connection setup and tear-down over a light-trail requires no optical switching and is done entirely through signaling in the OOB control channel using a protocol that we define in Section III.

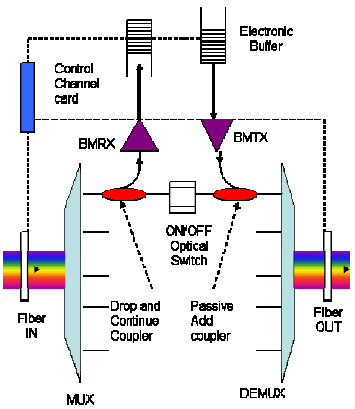


Fig. 2. Light-trail node specifics.

II. A. High Level System Description:

An IP/MPLS backbone network interacts with the railway-grid at select gateways to provide backbone connectivity with the rest of the Internet. Typically, the selection of IP/MPLS routers along the grid implies points at which a light-trail terminates or commences as shown in the Fig. 3. Each gateway communicates with a passing train through a wireless link.

We propose WiMax as a wireless candidate technology for the wireless link between a GW and a moving train. The choice of IEEE 802.16 WiMax is due to the support of the large distance-bandwidth product and ability to coexist with other wireless last inch technologies that users may in addition subscribe to (for e.g. cellular/GSM or WiFI).

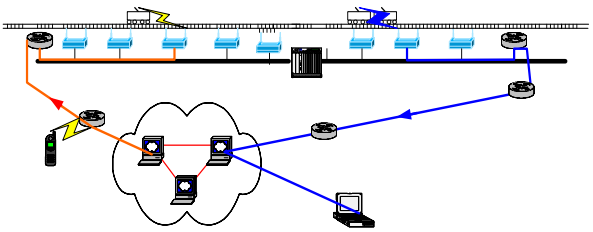


Fig. 3 Connection establishment in light-trains.

Information sent by the GW to the train is collected at a single platform within the train (the WiMax SS). This platform has a dual role: in one direction of communication it is used for communicating with the GW and in the other direction of communication it acts as an Access Point (AP) used to communicate with the end-users (residing in the train).

II. B: Duplex Communication in the Light-train System:

Let us now consider how a real-time application is supported at an end-user. A duplex communication application operates through our system using two communication modes – Forward Communication (FC) and Reverse Communication (RC) depending on the direction of communication *from* or *to* the end-user. We first describe how FC works. The application at the end-user sends information through the intra-train WiFi network to the access point. In our model, we assume that the WiFi network is configured to implement PCF mode of operation, so as to enhance throughput.

The AP/SS collects this data (in form of Ethernet frames) and attempts to send it to a gateway through whose geographic range the train is passing. A scheduling protocol in the WiMax domain and a fast hand-off mechanism is assumed (and described in Section IV.B) that enables establishment of communication and ubiquity between the SS and the GW. The WiMax frequency allocation is described in Section IV. The railway tracks are assumed to be divided into cells in succession, that are partially overlapped power-threshold defined regions supporting wireless communication. When a train passes from the range of one cell to another cell, a *hard hand-off* is assumed to happen.

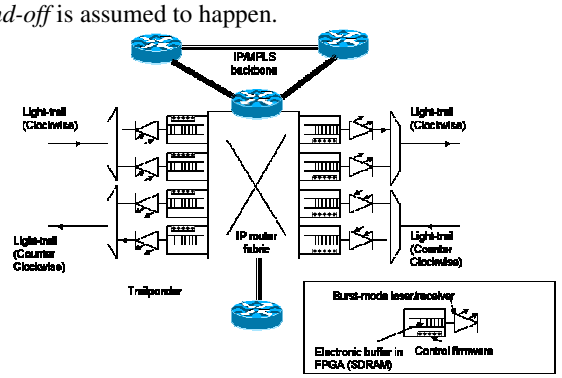


Fig. 4. Light-trail architecture : Data collection at the trailponder.

The packets that are sent to the GW, should now be transported into the core network. This is done by the optical (light-trail) solution. Since a light-trail is an all-optical time-

shared medium (bus), the packets are buffered in light-trail *trailponders* (electronic queues with added fast ON/OFF using burst-mode capability) [9] until a successful *connection* (in the light-trail) is established. A connection in a light-trail is when two nodes communicate over a light-trail. This is a software (control plane) defined operation that involves no optical switch configuration resulting in dynamic bandwidth provisioning. Once a connection at a gateway is established, the information is sent through the light-trail to the nearest IP/MPLS router that is subsequently assumed to be connected to the rest of the Internet (core network).

In the reverse direction of communication (RC), i.e. from the IP/MPLS router to the end-users, information for a particular end-user in form of packets arrives at a specific IP/MPLS router. This router is connected to a light-trail that contains a GW in whose range the train is moving. The data to be sent to the users in the train is collated in a *trailponder* [9] at the IP/MPLS router (as shown in Fig. 4). The trailponder is responsible for sending this data into the light-trail by forming a connection. Since the light-trail is a time-shared medium, the trailponder at the IP/MPLS router competes with trailponders at GWs in connection provisioning. Multiple trailponders may exist at an IP/MPLS router – each supporting a different light-trail (on one of the many WDM channels). A single trailponder at an IP/MPLS router supports communication to all the nodes in the light-trail using the optical multicast property.(Fig. 4).

The information is sent into an *appropriate* light-trail (the one that contains the GW that has in its range the train that further contains the end-user which requested for the information).

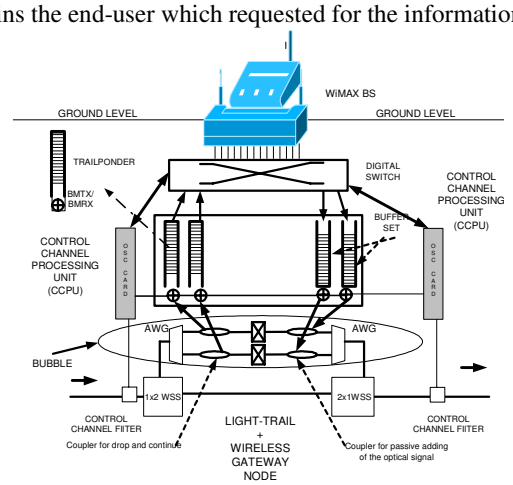


Fig. 5. Cross-medium node architecture supporting a WiMax BS and light-trail based optical network. Note that the 1x2 WSS conserves power and cost by processing only select WDM channels at a site.

Gateway nomenclature: The GW through whose range the train is presently passing is called a *live* gateway; while the GWs through whose range the train has already passed are called *dead* gateways (for that particular train); finally, the gateways in whose range a train would soon arrive (downstream of the live gateway) are called *dormant* gateways.

It may happen that the entire information content (say a file) is not transmitted by the live gateway while the train was in its range. The train may have moved to the next (dormant) gateway and begun communicating with this gateway (turning its status to a live gateway). The status change in GWs is dependent on a hand-off mechanism that is principally based on a power-threshold algorithm as defined in the IEEE 802.16e WiMax mobility standard in conjunction with our optical out-of-band control-channel. Hence, when the train undergoes a hand-off and begins to communicate with the next GW, the remainder of the file has to be transmitted by this new "live" GW. In such an event, the new live GW should know the exact fraction of the file that has already been transmitted and also have access/contain the remaining fraction that is to be transmitted into the wireless channel. *Due to optical multicast property of the light-trail all the nodes have access to the data that was sent by the IP/MPLS router, and hence the problem of availability of data at the new live gateway does not arise*. However to avoid replication of transmission and thereby efficiently utilize the already scarce wireless bandwidth, it is necessary that the new live node begin transmission from the frame/packet (within the file), where the previous gateway ceased transmission. This information pertaining to fractional file transfer is sent from one GW to the next through the out-of-band Optical Supervisory Channel (OSC)/control channel.

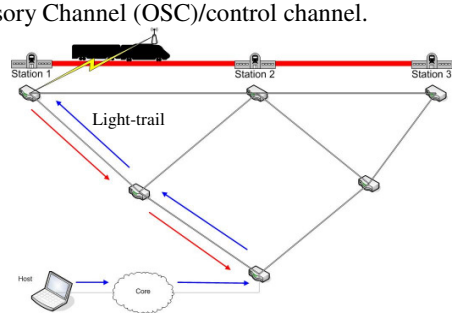


Fig. 6. IP Routing when the train has registered at station 1.

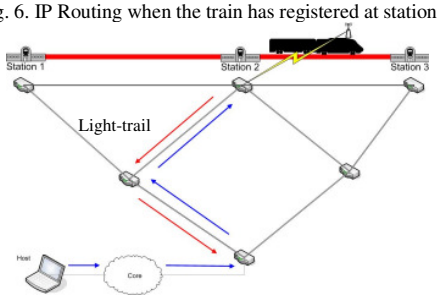


Fig. 7. IP Routing when the train has registered at station 2.

III. IP ROUTING WITH CONTROLLED OPTICAL MULTICASTING

In this subsection, we describe how the IP layer reacts to the moving train.

When a train passes over a router, it registers itself at the router (which is the convener node of a light-trail). The router advertises to the rest of the core network that the train is reachable through itself. To support a large number of IP

addresses we use Network Address Translation (NAT) at the AP.

Consider Fig. 6 in which an end-user in the train desires to communicate with a host in the core network. When the train crosses *station 1* and moves towards *station 2*, the router at *station 1* advertises the train to be in its range (first hop path). Correspondingly, the routing advertisements for the train flow along the direction of the red lines (to the core network) and the data packets sent by a host in the core network flow along the direction of the blue lines (as shown in Fig. 7). It is assumed that the train itself is an edge router, and the router at station 1 is simply advertising a shortest route (of one hop) to this edge router.

In Fig. 7, the train has moved to *station 2*, where a light-trail is terminated and a new light-trail begins (the terminating/end-node for the previous light-trail is the convener node for the new light-trail). It is assumed that between the previous and the next light-trail there is an IP router using the connection configuration as shown in Fig. 7. The train now registers itself with this IP-router (at *station 2*). The router at *station 2* begins to advertise the train to be in its one-hop range. The routing advertisements for the train flow along the direction of the red line in Fig. 7 and the data packets sent by a host in the core network flow along the blue lines.

When the train registers at *station 2*, router 1 stops route advertisements, while router 2 starts route advertisements of the train being in its one-hop range. Router 1 is intimated of the new association (between the train and router 2) through the out-of-band control channel that connects routers 1 and 2. Data not transmitted by router 1 to the train, is then sent to router 2. The flow of data between router 1 and router 2 is due to optical multicasting and this is shown in Fig. 8.

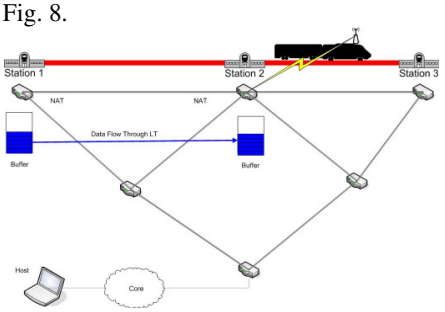


Fig. 8. Populating buffers across interconnected light-trails.

IV. PRINCIPLES OF THE INTEGRATED PROTOCOL DESIGN

In this section we define the key principles of a protocol that integrates the optical and wireless layers. Consider a train on a track with our proposed (optical + wireless) protocol.

In particular, the protocol implements the following key tasks:

1. Management of hand-off leading to the concept of *fast* hand-off by using the dual property of optical multicasting and dynamic provisioning.
2. Signaling between gateways to enable seamless and efficient connection provisioning.
3. Setup and tear-down of optical connections between IP routers (at convener node/end-node) and the gateway.
4. Traffic engineering of connections amongst GWs to provision real-time and data-centric services facilitating seamless communication to and from the moving train.

IV. A. WiFi Protocol Functioning In The Train

The first source of delay for a user that sends a packet is within the WiFi network. In our model, the access point in the train is assumed to use the Point Coordination Function (PCF) to communicate with the users inside the train. PCF enables good performance at high loads, as desired in a train with a large number of active users. The PCF is used in an infrastructure-mode whereby an AP (access point) is able to centrally control user access into the shared wireless medium. To do so, the AP polls users in a time-duration denoted as – *contention-free-period* (CFP). Users are then allocated slots by the AP on the basis of the polling scheme. Time is assumed to be in form of successive superframes, and each superframe is further composed of the CFP and a *contention period* (CP). Transmission in the CFP is the result of the polling mechanism initiated by the AP while transmission in the CP is assumed to be a Distributed Coordination Function (DCF).

IV. B. Frequency allocation in WiMax channel and hand-off using optical multicasting

In this section, we will discuss how the WiMax protocol functions in the light-train system. On account of the fast hand-off and associated optical layer dynamic provisioning/optical multicasting features, the WiMax protocol is closely tied to the optical/physical infrastructure, in the sense that, the protocol is adapted to suit the light-train mode of operation. In particular, two aspects need consideration: (1) WiMax frequency allocation and (2) WiMax hand-off. The first aspect will now be considered.

We assume that f_{WiMax} be the set of OFDM channels available for use in the WiMax spectrum for providing communication between the GW (i.e. WiMax BS) and the train AP (WiMax SS). In typical wireless deployments, we would assume a cellular pattern with Base Stations and antennas launching wireless signal into cells, supported by a frequency allocation pattern that allows efficient reuse of frequencies by maintaining geographic diversity. However, in the case of light-trains, the frequency allocation differs significantly from regular cellular/WiMax networks. The difference in allocation of frequencies is due to the following reasons:

1. Conjecture: On a single track, two moving trains are separated by a *minimum distance* d_{train} that is necessary for avoiding collision at some minimum cruising speed. Hence, if the cell width C_W, is greater than the train length (which is assumed to be true in our case), then it implies, to prevent collision between back-to-back trains, the relation $d_{train}>2C_W$ should likely be valid, and hence two trains will not be in adjacent cells (as shown in Fig. 9).

2. Under the assumption that reason (1) above is valid, this means that the entire frequency band is reused in the adjacent cell as long as another train does not come in this cell's range on an opposite/another track. However two trains moving in opposite directions can be in adjacent cells (just before crossing each other), and this implies a different allocation strategy considered next.

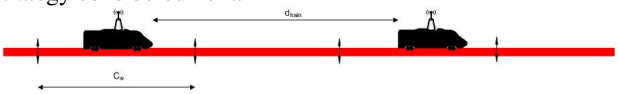


Fig. 9. Frequency allocation and its relation to cell-width (inter-GW distance), minimum distance to avoid collision and train size.

3. For the case when two trains cross each other in opposite tracks, we assume that through the network management system we are able to isolate the cell that would be *live* and contain both the trains. This live cell(s) with two trains in its range is called a *hyperlive cell*. When a cell is *hyperlive*, the OFDM channels are divided based on FDM and all odd channels are allocated to the train moving in one direction while all even channels are allocated to the train moving in the opposite direction

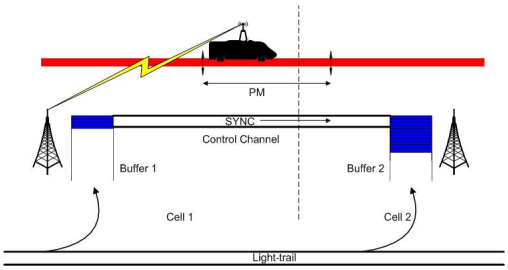


Fig. 10a: Sync.

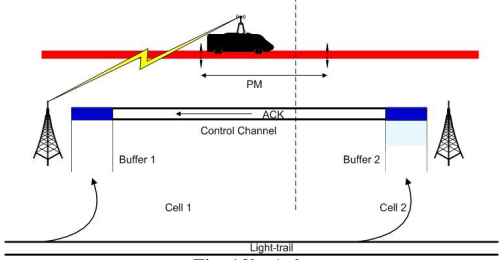


Fig. 10b: Ack.

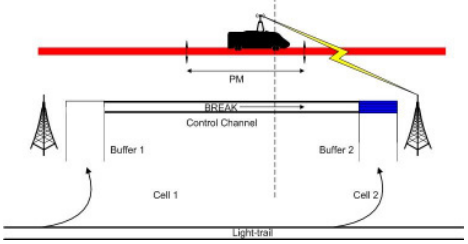


Fig. 10c: Break

4. For the case when multiple (Q) trains are within the range of a single cell, then, the allocation of OFDM channels is done as per (3) except that the set of OFDM channels are now divided into Q disjoint subsets, one for each train. This situation is likely to happen at train stations.

Hand-off in WiMax is based on the principle of *break before make* and is assisted by light-trail properties of optical multicasting and dynamic provisioning. WiMax hand-off involves the following steps:

(1) When a train enters the *provisioning margin* of a cell through which it has traversed, it indicates so to the GW through the *UL Map* of the WiMax uplink frame. The train gets this indication through a combination of distance traveled (since GWs are pre-allocated and static, thereby enabling the GW to tell the train of its geographic range) as well as a power threshold. Upon entering the provisioning margin the train begins invoking hand-off procedures.

(2) This indication (through the UL map) triggers a series of *sync* steps at the WiMax GW. The *live* WiMax GW now has to inform to the next GW (which is dormant) about the arrival of the train in its range and then has to sync with the buffer at the next dormant GW so that there is no duplication of transmitted data (into the wireless channel). This step is pictorially represented in Fig. 10a.

(3) Sync between two adjacent GWs involves the live GW informing the adjacent dormant GW of the packet numbers that are being sent. This is done through the OOB control channel. In light-trails we have assumed that layer 2 frames are encoded with VLAN tags [9], these tags are useful in determining packet numbers. By stacking VLAN tags, we can have sequence numbering. The dormant GW receives the same packets as the live GW through optical multicasting property of the light-trial – though it does not transmit the received packets into the wireless medium. The dormant GW upon receiving the packet number information from the live GW sends an ACK-HAND-OFF frame. This ACK-HAND-OFF frame tells the live GW that the dormant GW has discarded all packets prior to the last packet number sent by the live GW. The live GW then sends (after a duration of time that corresponds to the end of a WiMax cycle) a final BREAK packet to the dormant GW and *stops* transmitting data into the wireless domain (as shown in Fig. 10b). By waiting for the duration of WiMax cycle to end, the live GW ensures that the train does not transmit any packets in the downlink during hand-off. The BREAK packet contains the packet number of the last packet that was transmitted by the live GW before it became *dead*. Once the dormant GW receives the BREAK packet it notes the last transmitted packet number and begins transmitting the packets in its buffer by setting its pointer to the *last packet sent* value (which it received from its adjacent/previous GW). The dormant GW has hence become a live GW, and hand-off has occurred (as shown in Fig. 10c).

IV.C. Light-trail Protocol Design

This protocol works as follows: for every light-trail, the control channel is assumed to be synchronized with respect to each node. This assumption is valid since the control channel is optically dropped and electronically processed at every node in the light-trail; the OEO function

facilitates synchronization amongst nodes in the light-trail. At the data layer, i.e. within the light-trail itself, we assume large duration (1-5 ms) time-slots (data-time-slots) for connection provisioning. Two adjacent time-slots are separated by a guard-band of about 10 *us*.

One of the light-trail nodes acts as an arbiter of bandwidth within the light-trail (and called light-trail controller). This node would receive from every other node (within the light-trail) a bandwidth request in the *t-1*st time-slot and then would arbitrate so as to give the highest deserving node the t^{th} time-slot for transmission. The process of requesting bandwidth by nodes and the subsequent allocation (by the arbiter) is done through the out-of-band control channel. Also this process is done ahead in time, i.e. nodes request for bandwidth in the present time-slot and their request (if granted) is fulfilled in the next time-slot. This protocol is a derivative of the two-stage auction algorithm proposed in [10] and uses *bids* as requests for bandwidth allocation. The protocol uses bids within control packets to communicate between nodes and the arbiter.

We define the following control packet types for our proposed protocol:

IV. D. Light-trail protocol working

The dynamic provisioning protocol works as follows: Each node in a light-trail sends a bid packet indicating the buffer status in the trailponder to the light-trail controller. The light-trail controller then grants the node with the highest critical buffer status rights to form a connection in the next time-slot. The node could be an intermediate node (thus forming a FC connection) or could be the IP/MPLS router (thus forming an RC connection). We first state the conventions used and then describe the protocol in operation.

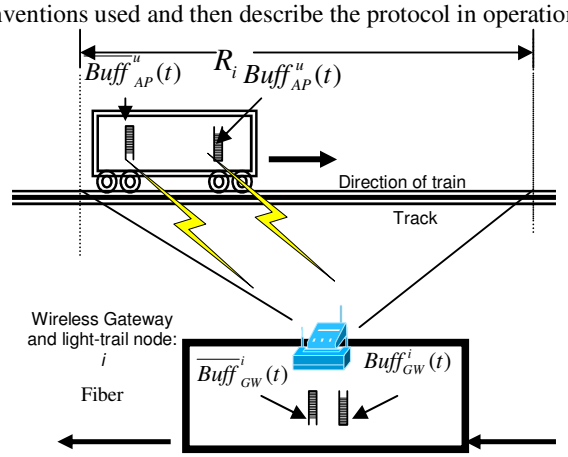


Fig. 11. Buffer representations at gateway and in train along with range of i^{th} gateway.

Convention

a: number of trains in the system.
u: represents a particular train.
m: represents a particular user.
i: represent a particular Gateway (GW)
$R_i(t)$: range of the i^{th} gateway at time *t*.

Let each user *m* generate data at a rate λ^m packets per second, where μ^m is the packet service rate.

Let us assume a set of *h* services be denoted by the set $S = \{S_1, S_2, \ldots, S_q, \ldots, S_h\}$

Then, let us define the maximum allowable latency for each of these *h* services to be represented by a set: $\Delta = \{\Delta_1, \Delta_2, \ldots, \Delta_q, \ldots, \Delta_h\}$.

The train *u* has an Access-Point (AP), and packets arrive at the AP from the users at a combined average arrival rate λ^u_{AP}.

Similarly packets arrive at the AP in train *u* from the underlying core network (through the WiMax channel) at an average rate of $\overline{\lambda}^u_{AP}$.

The buffer at the access point in train *u* that collects packets in FC direction (from the users) is denoted by $Buff^u_{AP}$ and whose instantaneous value is $Buff^u_{AP}(t)$.

In the FC direction, packets are sent by the AP that also functions as a WiMax SS, transmitting them to a WiMax BS that resides in conjunction with the GW, as shown in Fig. 11.

Let us assume GW *i* to be the communicating gateway for train *u* i.e. the train is passing through the geographic range of *i*. Then, the buffer for FC communication at GW *i* is $Buff^i_{GW}$, and it collects packets sent by the AP in train *u*. The instantaneous occupancy value of this buffer is denoted by $Buff^i_{GW}(t)$.

Similarly, at GW *i* we have in the RC direction buffer $\overline{Buff}^i_{GW}$ (with instantaneous value $\overline{Buff}^i_{GW}(t)$) that collects packets from the optical layer and transmits these packets into the WiMax channel (to the passing train).

Buffers $Buff^i_{GW}$ and $\overline{Buff}^i_{GW}$ have service differentiation capability, i.e. different counters exists for different services enabling service differentiation amongst the packets queued in the buffer. These counters are able to measure the time elapsed since packets for a particular service entered the buffer. Service differentiation is done by use of VLAN tagging on layer-2 frames in light-trails [9].

$B_{\max}$: size of the buffer at a node.

T_{WiMax} *be* the expected duration of a WiMax cycle. T_{WiMax} constitutes the following periods – uplink frame, downlink frame and associated guard-bands.

$t^{i,u}_{DL}$ be the expected downlink time duration allocated to GW *i* on the WiMax channel for transmission to train *u*.

$t^{i,u}_{UL}$ be the expected uplink time duration allocated for transmission in the WiMax channel at the AP in train *u* for transmission to GW *i* in a cycle of duration T_{WiMax}.

Then, $t^{i,u}_{DL} + t^{i,u}_{UL} \approx T_{WiMax}$ is the WiMax cycle time, neglecting the guard-band between uplink and downlink frames. The expected values of $t^{i,u}_{DL}$ and $t^{i,u}_{UL}$ are computed as described in the next section.

Bandwidth allocation policy: We will now show how bandwidth (connection) provisioning happens in the light-trail. For bandwidth allocation consider Fig. 11.

Let $y_{ikq}(t)$ be the time elapsed since the first packet of service type q entered the buffer $Buff^{i}_{GW}$ at GW i destined for communication in the light-trail k in the FC direction at time t.

Let $\overline{y}_{ikq}(t)$ be the time elapsed since the first packet of service type q entered the buffer $\overline{Buff}^{i}_{GW}$ at GW i destined for communication in the light-trail k in the RC direction at time t

Packet value computation at a GW i for transmission on light-trail k at time t in FC direction is denoted by:

$$packet\_value_{ik}(t) = \max\left[\frac{Buff^{i}_{GW}(t)}{B_{\max}}, \frac{\max(y_{ikq}(t))}{\max(y_{ikq}(t)) + \gamma_{ik}(t)}\right] \quad (1)$$

Where, $\gamma_{ik}(t) = \min_{q=1,\ldots h}(\Delta_q - y_{ikq})$: service statistics at GW i in FC direction; where service statistics refers to the minimum delay tolerance limit for the buffer $Buff^{i}_{GW}$ at GW i for communication in the light-trail k in the FC direction at time t. Subsequently $q'_i = \arg\left[\min_{q=1,..h}(\Delta_q - y_{ikq})\right]$ denotes the service that corresponds to minimum tolerance value in the buffer at node i in FC direction.

Let $\overline{q'_i} = \arg\left[\min_{q=1,..h}(\Delta - \overline{y_{ikq}})\right]$ be the service that contributes to the minimum delay tolerance in the buffer at node i in RC.

Let $\overline{\gamma_{ik}}(t) = \min_{q=1,\ldots h}(\Delta - \overline{y_{ikq}})$ be the allowable delay at node i (how long can a node i wait without being serviced) in RC direction for the buffer $\overline{Buff^{i}_{GW}(t)}$; i.e. $\overline{\gamma_{ik}}(t)$

Buffer status computation at GW i (for packets received on light-trail k) for transmission on the WiMax channel at time t in the RC direction is denoted by:

$$\overline{packet\_value_{ik}}(t) = \max\left[\frac{\overline{Buff^{i}_{GW}(t)}}{B_{\max}}, \frac{\max(\overline{y_{ikq}}(t))}{\max(\overline{y_{ikq}}(t)) + \overline{\gamma_{ik}}(t)}\right] \quad (2)$$

In [10] we have observed that using a maximization of buffer utilization and service statistics as shown in (1), (2), leads to a proportionally fair [11] allocation of bandwidth resources in a shared medium. The explanation is that $Buff^{i}_{GW}(t)$ value is mapped to a combination of concave and sigmoidal function that ultimately leads to proportional fairness.

Dynamic provisioning protocol in a light-trail

Stage 1:
At every node in the light-trail except IP router :
  compute *packet_value for each node*
  send *packet_value to controller*
At IP router connected to convener of the light-trail k:
  for *every buffer*
    compute *packet_value*
    send *packet_value to controller (through control channel)*
At controller :
    receive *all packet_value*
    compute *max(packet_value)*
    compute $suc_k(t)$ = *argmax(packet_value)*
    send $suc_k(t)$ *information to all nodes*

Stage 2:
At all nodes in light-trail k and at IP router at convener node :
  if $suc_k(t)$=*node index*
    wait *till end_of_current_time-slot*
    transmit *for one time-slot duration or till* $buff_{ik}(t)$=0
  else
    idle
  endif

Algorithm 1: Communication between GW and IP Router (for both FC and RC)

IV. D. Special Case: Reverse Communication

The IP router sends data to the users in the train in reverse communication direction. The data sent depends on the user IP-destination address and whether the router supports a light-trail such that a GW on the light-trail contains the train in its geographic range. To send the data from the IP router to the GW, a connection is set up on the light-trail, established between the IP router and the GW that contains the train. This connection is also multicast to all the GWs downstream (in the direction of the moving train). Hence the GW that contains the train (*live*) and the GWs at which the train is expected to arrive (*dormant*) are privy to the information transmitted on the connection.

The gateway or router in whose range the train resides, receives the data (on the connection) and attempts to send this to the AP (in the train) through the WiMax channel. Data that cannot be sent through the wireless channel because the train has left the range of a live GW is sent by the next GW. Data received by the AP is then transmitted to the end-users over the PCF based WiFi connection.

Shown in algorithm 1 is the pseudo code for bandwidth assignment within the light-trail at both the GWs and at the IP router in both directions of communication, i.e. FC and RC.

V. DELAY ANALYSIS IN THE LIGHT-TRAIN SYSTEM

The integrated optical-wireless solution that gives bandwidth to fast-moving users in trains, consists of wireless access, wireless interconnection and optical infrastructure thereby requiring a unified approach to bandwidth provisioning. In terms of implementation, the variation in provisioning methodologies seen between: the end-users and the AP (WiFi); the AP and the GW (WiMax); the GW and

router (nodes of the light-trail) act as the three principle sources of delay, in the FC and RC directions.

We now compute each of these three delays i. e. at WiFi, within WiMax and accessing the light-trail and then present a unified delay model as a function of network load. We further extend this model to the RC case as well. The final delay is represented by the following set of equations with the associated conventions described subsequently:

The delay experienced by a packet in the FC direction is given by:

$$\delta_{final-FC} = \delta_{WiFi-FC} + \delta_{WiMax-FC} + \delta_{LT-FC} \quad (3)$$

The delay experienced by a packet in the RC direction is given by:

$$\delta_{final-RC} = \delta_{WiFi-RC} + \delta_{WiMax-RC} + \delta_{LT-RC} \quad (4)$$

Finally, the average delay that a packet experiences through the network is:

$$\delta_{final} = \left[\delta_{final-FC} + \delta_{final-RC} \right] / 2 \quad (5)$$

Where, $\delta_{WiFi-FC}$ is the delay in FC through WiFi; $\delta_{WiFi-RC}$ is the delay in RC through WiFi; $\delta_{WiMax-FC}$ is the delay in the FC through WiMax, while $\delta_{WiMax-RC}$ is the corresponding delay in the RC through WiMax and finally, δ_{LT-FC} is the delay in FC through the light-trail, while δ_{LT-RC} is the delay in the RC through the light-trail. We now compute each of the above six delay functions.

V. A. WiFi Delay

To compute the delay we assume that there are *W(u)* users in train *u* that use the PCF mode for communication. The packet inter-arrival times at each user are assumed to be exponentially distributed with rate λ. Let the duration of a superframe (used in the PCF mode) be denoted by T_S^{WiFi} and let *L* be the expected length of a packet at each user. The utilization of each node is denoted by ρ_{WiFi}. We adapt from the model presented in [6] to compute the average queuing delay experienced by an arbitrary packet arriving at the m^{th} user in the train *u*.

As shown in [6], since each polled user transmits exactly once in every superframe, the service rate of each user is $\mu_{WiFi} = 1/T_S^{WiFi}$, and the net-utilization is given by $\rho_{WiFi} = \lambda / \mu_{WiFi} = \lambda T_S^{WiFi}$. From the above, the delay experienced by a packet that passes through the WiFi network sent by end-user *m* in the train *u* is given by:

$$D_{WiFi}^{m} = \frac{1}{1-\lambda T_S^{WiFi}} \left[\frac{T_S^{WiFi}}{2} + \left(\frac{\rho_{WiFi} L^2 (2m-1)(1-\rho_{WiFi})}{T_S^{WiFi}} + L \right) (1-\rho_{WiFi}) \right] \quad (6)$$

Let $\lambda_m^{'u}$ be the departure rate at the end-user *m* in train *u*. This implies that at user *m*, packets arrive (from client applications at a rate λ_m^u and are sent out at an *effective* rate $\lambda_m^{'u}$, where the difference between $\lambda_m^{'u}$ and λ_m^u is the result of the PCF bandwidth allocation strategy (due to control over-head). The net arrival rate at the AP (λ_{ap}^u) is given by:

$$\lambda_{ap}^u = \sum_{m \in W^m(u)} \lambda_m^{'u} \quad (7)$$

Where $W^m(u)$ represents the set of users in train *u*. The departure rate at the end-users, in steady state, can be computed in terms of the arrival rate (λ_m^u) and the average per-packet delay (D_m^u), that packets from user *m* experience due to PCF allocation strategy. In steady state, the number of packets (N_m^u) that are buffered at the user *m* is given by Little's formula: $N_m^u = \lambda_m^u D_m^u$. Hence we have:

$$\lambda_m^u D_m^u = \lambda_m^u T_S^{WFi} - \lambda_m^{'u} T_S^{WFi} = N_m^u \quad (8)$$

$$\therefore \ \lambda_m^{'u} = \lambda_m^u \left(1 - \frac{D_m^u}{T_S^{WFi}} \right) \quad (9)$$

V. B. WiMax Delay

We now compute the average delay experienced by a packet over the WiMAX channel. The MAC frame structure consists of an uplink and a downlink sub-frame, whose durations are dependent on the provisioned services and are controlled by the BS. A SS requests for transmission opportunities on the uplink channel. The BS collects these requests and grants permissions to the SSs based on their service agreements. Allocation in WiMax is done through Downlink Map (DL Map) and Uplink Map (UL Map) that carries information of future allocations i.e. sub-frame assignment, and duration.

Our model is similar to the model proposed in [7] with the primary exception that it is designed for communication with high-speed voluminous moving users. This implies the model has a single BS located at the GW and a single SS located in the train. The BS polls to the SS and based on this polling allocates bandwidth in the UL-map (indicating size of uplink transmission). The BS hence allocates bandwidth to the SS by sharing the entire bandwidth between itself (downlink) and the SS (uplink). To compute the delay experienced by a packet in FC, through the WiMax channel, we consider scenarios that are based on the time of arrival of a packet and state of the buffer at the AP.

Let us denote $t_{UL}^{i,u}$ to be the expected uplink time that train *u* receives through the WiMax channel when moving through the GW *i*. Let μ be the service rate at the AP and $N_{flush} = t_{UL}^{i,u} / \mu$ be the maximum permissible buffer occupancy (in terms of number of packets) that can be scheduled over the WiMax channel in the uplink –FC).

At the GW *i* (in RC)

if train *u* is in *Range*
  In each *slot* of length T_{WiMax}
    compute $\lambda_{GW}^{i,u}$
    send *REQ* to get $\lambda_{AP}^{i,u}$ from the *AP*
    compute $t_{UL}^{i,u} = \frac{\lambda_{AP}^{i,u}}{\lambda_{AP}^{i,u} + \lambda_{GW}^{i,u}}$ and $t_{DL}^{i,u} = T_{WiMax} - t_{UL}^{i,u}$
    send $t_{UL}^{i,u}$ and $t_{DL}^{i,u}$ to *AP (in UL, DL map)*
    receive data in $t_{UL}^{i,u}$
    on completion of $t_{UL}^{i,u}$, **transmit** for duration of $t_{DL}^{i,u}$
else
    sleep till another train *registers*
endif

At the AP in train *u* (in FC)

If GW i is in *Range*
    In each *slot* of length T_{WiMax}
    compute $\lambda_{AP}^{i,u}$
    send $\lambda_{AP}^{i,u}$ to *GW as response to REQ*
    get $t_{UL}^{i,u}$ and $t_{DL}^{i,u}$ from GW *i*
    transmit for the time duration of $t_{UL}^{i,u}$
    receive for the time duration of $t_{DL}^{i,u}$
else
    search for a *GW*
endif

Algorithm 2: Communication between AP and GW (both FC and RC) using WiMax communication

The communication through the WiMax channel consists of the two periods denoted by $t_{UL}^{i,u}$ and $t_{DL}^{i,u}$ for uplink (FC) and downlink (RC) communication respectively, between GW *i* and train *u*. We also assume that the time a train spends communicating to a GW is denoted by $T_{WiMax}^{v(t),u,i} \approx T_{WiMax}^{u,i} \approx T_{WiMax}$ whereby we relax the constraint of velocity dependence (and hence Doppler effect), assuming measurements are made at a *cruise* velocity of in the region of 100~300 kmph. Then, the time the train spends in communicating with the gateway is given by an integral multiple of: $T_{WiMax} = \left(t_{UL}^{i,u} + t_{DL}^{i,u}\right)$. We also assume that $\lambda_{AP}^{i,u}$ is the arrival rate at the AP (from the users in the train) and $\lambda_{GW}^{i,u}$ is the arrival rate at the gateway (from the light-trail).

For bandwidth allocation, the transmission time is divided into small slots with the AP and the GW both allotted sub-slots to transmit within each slot, the size of which is proportional to their respective arrival rates.

We now compute the expected values of $t_{UL}^{i,u}$ and $t_{DL}^{i,u}$ that are critical in computation of the delay through the WiMax channel. The *pdf*s of $t_{UL}^{i,u}$ and $t_{DL}^{i,u}$ are computed as follows:

Assuming that the arrival processes $\lambda_{GW}^{i,u}$ and $\lambda_{AP}^{i,u}$ are independent at the train (u) and the gateway (i), the cycle time, $T_{WiMax} \approx T_{WiMax}^{i,u}$ is decomposed into the following two sub-slots:

$$t_{UL}^{i,u} = \frac{\lambda_{AP}^{i,u}}{\lambda_{AP}^{i,u} + \lambda_{GW}^{i,u}} T_{WiMax} \text{ and } t_{DL}^{i,u} = \frac{\lambda_{GW}^{i,u}}{\lambda_{AP}^{i,u} + \lambda_{GW}^{i,u}} T_{WiMax}^{i,u} \quad (10)$$

In (10) we divide the cycle time T_{WiMax} proportional to the steady-state arrival rates $\lambda_{GW}^{i,u}$ and $\lambda_{AP}^{i,u}$. Their *pdf*s are computed as follows:

$t_{UL}^{i,u} = \left(\frac{1}{1+\lambda_{GW}^{i,u}/\lambda_{AP}^{i,u}}\right) T_{WiMax}$ and denote, $\left(\frac{1}{1+\lambda_{GW}^{i,u}/\lambda_{AP}^{i,u}}\right) = X$, then

$$\frac{1-X}{X} = \lambda_{GW}^{i,u} / \lambda_{AP}^{i,u}. \quad (11)$$

Let us now define $\overline{\lambda}^{i,u} = \frac{\lambda_{GW}^{i,u}}{\lambda_{AP}^{i,u}}$ as the quotient of two independent random variables.

Let $g(\lambda_{GW}^{i,u})$ denote the *pdf* of $\lambda_{GW}^{i,u}$ and $h(\lambda_{AP}^{i,u})$ denote the *pdf* of $\lambda_{AP}^{i,u}$. Then the *pdf* of $\overline{\lambda}^{i,u}$ is $f(\overline{\lambda}^{i,u})$ and given by:

$$f(\overline{\lambda}^{i,u}) = \int_{v=-\infty}^{\infty} g(v\overline{\lambda}^{i,u}) h(v) |v| \, dv. \quad (12)$$

From which we compute the expectation of $t_{UL}^{i,u}$ as:

$E\left[t_{UL}^{i,u}\right] = \int_0^{T_{WiMax}} t.y\left(t_{UL}^{i,u} = t\right) dt$, where $y\left(t_{UL}^{i,u} = t\right)$ is the *pdf* of $t_{UL}^{i,u}$ and can be computed from $f(\overline{\lambda}^{i,u})$.

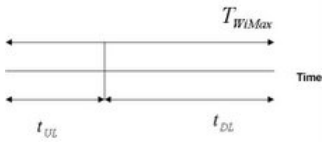
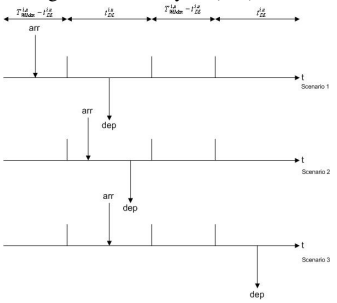
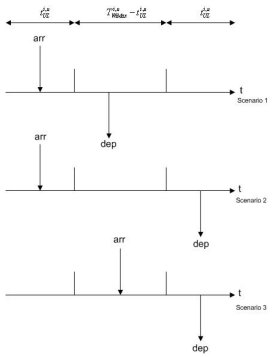


Fig. 12. WiMax cycle (slot) time.



(a)



(b)

Fig. 13. WiMax communication: (a) shows the three scenarios for FCd (b) shows the three scenarios for RC.

Our desire is to compute the delay experienced by packets that move in the FC as well as the delay experienced by packets moving in the RC directions assuming that the WiMax system has an effective service rate of μ_{WiMax} packets per seconds. The composition of the communication cycle in WiMax is as shown in Fig. 12. To compute the average delay that a packet experiences in FC/RC direction through the WiMax channel we abstract three scenarios that a packet can experience and are described next. The assumption is that a packet arrives at the AP (from the end-user) at time $t_{arr} \in [0, T_{WiMax}]$ and gets transmitted at $t_{dep} \in [0, T_{WiMax} + t_{UL}]$ which avoids an infinite queue system.

The scenarios below describe delay computation in the FC direction and the RC is an extension of the FC delay shown subsequently.

Scenario 1): In this scenario the packet arrives while the AP is being serviced (through $t_{UL}^{i,u}$) and departs in the same interval (uplink frame) as shown in Fig. 13a *scenario 1*.

The condition is denoted by: $t_{arr} \in \left[0, t_{UL}^{i,u}\right]$ and $t_{dep} \in \left[0, t_{UL}^{i,u}\right]$.

The average delay experienced by a packet in this case is: $t_{UL}^{i,u}/2$. The probability of occurrence of this scenario is given by the joint probability that the arrival occurs in the duration of the uplink frame and the departure also occurs within the same duration. This is given by:

$$p_1 = p(t_{arr} \in \left[0, t_{UL}^{i,u}\right]).p(t_{dep} \in \left[0, t_{UL}^{i,u}\right]) \tag{13}$$

The two probabilities are computed as:

$$p(\ t_{arr} \in \left[0, t_{UL}^{i,u}\right]) = \frac{t_{UL}^{i,u}}{T_{WiMax}} \tag{14}$$

$$p(t_{dep} \in [0, t_{UL}]) = p(Buff_{AP}^{i,u} < N_{flush}) \tag{15}$$

Where, $Buff_{AP}^{i,u}$ as noted before denotes the occupancy of the buffer at the AP of train *u* while it passes through gateway *i* and N_{flush} denotes the maximum number of packets possible to be transmitted through the WiMax channel, with service rate μ_{WiMax}.

While the first probability ($p(\ t_{arr} \in \left[0, t_{UL}^{i,u}\right])$) can be computed by examining the expected value of $t_{UL}^{i,u}$, to compute the second probability, we assume that at the AP, the buffer forms an M/M/1/c queue. Let us further assume that $N_{flush}<c$ and hence we compute the probability of the occupancy as:

$$p(Buff_{AP}^{i,u} < N_{flush}) = \sum_{r=0}^{N_{flush}-1} \left(\frac{\lambda_{AP}^{i,u}}{\mu_{WiMax}}\right)^r \left(1 - \frac{\lambda_{AP}^{i,u}}{\mu_{WiMax}}\right) \tag{16}$$

$$= \sum_{r=0}^{N_{flush}-1} \left(\rho_{AP}^{i,u}\right)^r \left(1 - \rho_{AP}^{i,u}\right)$$

Hence, $p(Buff_{AP}^{i,u} < N_{flush}) = 1 - \left(\rho_{AP}^{i,u}\right)^{N_{flush}}$ (17)

Therefore, $p_1 = \left(1 - \left(\rho_{AP}^{i,u}\right)^{N_{flush}}\right).\frac{t_{UL}^{i,u}}{T_{WiMax}}$ (18)

Scenario 2): The second case denotes arrival of a packet when an uplink frame is being transmitted and the corresponding departure occurs during the *next* uplink frame, and this is shown in Fig. 13a *scenario 1*. In this case the packet has to wait for the duration of (1) the current uplink frame (to complete transmission) and (2) the next downlink frame. The packet is then transmitted during the subsequent uplink frame.

This condition is stated as: $t_{arr} \in \left[0, t_{UL}^{i,u}\right]$ and $t_{dep} \in \left[T_{WiMax}, T_{WiMax} + t_{UL}^{i,u}\right]$.

The average delay experienced by a packet in this second scenario is T_{WiMax}. The probability of occurrence of this scenario is:

$$p_2 = p(\ t_{arr} \in \left[0, t_{UL}^{i,u}\right]).p(t_{dep} \in \left[T_{WiMax}, T_{WiMax} + t_{UL}^{i,u}\right]), \tag{19}$$

The two independent probabilities are computed as:

$$p(\ t_{arr} \in \left[0, t_{UL}^{i,u}\right]) = \frac{t_{UL}^{i,u}}{T_{WiMax}} \tag{20}$$

$$p(t_{dep} \in \left[T_{WiMax}, T_{WiMax} + t_{UL}^{i,u}\right]) = p(Buff_{AP}^{i,u} \geq N_{flush}) \tag{21}$$

Simplifying and using result (17) we get:

$$p_2 = \frac{t_{UL}^{i,u}}{T_{WiMax}}.p(Buff_{AP}^{i,u} \geq N_{flush}) = \frac{t_{UL}^{i,u}}{T_{WiMax}}\left(\rho_{AP}^{i,u}\right)^{N_{flush}} \tag{22}$$

Scenario 3): The last scenario represents when a packet arrives during the downlink frame and is transmitted in the next uplink frame as shown in Fig. 13a *scenario 3*. This scenario is stated as:

$$t_{arr} \in \left[t_{UL}^{i,u}, T_{WiMax}\right] \text{ and } t_{dep} \in \left[T_{WiMax}, T_{WiMax} + t_{UL}^{i,u}\right]$$

The average delay experienced by a packet in this scenario is given by: $\left(\left(T_{WiMax} - t_{UL}^{i,u}\right) + t_{UL}^{i,u}\right)/2 = T_{WiMax}/2$. (23)

The probability of occurrence of this scenario is given by:

$$p_3 = p(\ t_{arr} \in \left[t_{UL}^{i,u}, T_{WiMax}\right]).p(t_{dep} \in \left[T_{WiMax}, T_{WiMax} + t_{UL}^{i,u}\right]) \tag{24}$$

The two independent probabilities are computed as:

$$p(\ t_{arr} \in \left[t_{UL}^{i,u}, T_{WiMax}\right]) = 1 - \frac{t_{UL}^{i,u}}{T_{WiMax}}, \text{and}$$

$$p(t_{dep} \in \left[T_{WiMax}, T_{WiMax} + t_{UL}^{i,u}\right]) = p(Buff_{AP}^{i,u} < N_{flush}) \tag{25}$$

Hence, $p_3 = \left(1 - \frac{t_{UL}^{i,u}}{T_{WiMax}}\right).\left(1 - \left(\rho_{AP}^{i,u}\right)^{N_{flush}}\right)$ (26)

Therefore by combining (13)–(26), the expected delay a packet experiences through the WiMax channel in the FC direction is given by:

$$\delta_{WiMax}^{FC}(i,u) = p_1.\frac{t_{UL}^{i,u}}{2} + p_2.\left(T_{WiMax} - \frac{t_{UL}^{i,u}}{2}\right) + p_3.\frac{T_{WiMax}}{2} \quad (27)$$

Substituting and solving we get:

$$\delta_{WiMax}^{FC}(i,u) = \left(\frac{\left(t_{UL}^{i,u}\right)^2 + \left(T_{WiMax}\right)^2 - 3.t_{UL}^{i,u}.T_{WiMax}}{2T_{WiMax}}\right)\left(1 - \left(\rho^{i,u}\right)^{N_{flush}}\right) + t_{UL}^{i,u} \quad (28)$$

Since the scenarios for RC are identical (as can be seen from Fig. 13b), the expected delay a packet experiences through the WiMax channel in the RC direction is given by:

$$\delta_{WiMax}^{RC}(i,u) = \left(\frac{\left(t_{DL}^{i,u}\right)^2 + \left(T_{WiMax}\right)^2 - 3t_{DL}^{i,u}.T_{WiMax}}{2T_{WiMax}}\right)\left(1 - \left(\rho^{i,u}\right)^{N_{flush}}\right) + t_{DL}^{i,u} \quad (29)$$

V. C. Light-trail Delay

The final aspect of delay that a packet experiences through the light-train system is at the optical layer – i.e. through the light-trail.

Let us assume a light-trail *k* that has *n(k)* nodes and is supported by two IP-routers at each end (as shown in Fig. 1). The light-trail is time-shared by its *n(k)* constituent nodes for both FC and RC. Time-sharing is done through time-slotting the light-trail [10]. Since multiple nodes time-share the optical wavelength bus, we require a method for arbitration which was described in Section IV. Arbitration of bandwidth (time-slots) between nodes of a light-trail is done as follows:

(1) An arbitrary node is selected in the light-trail as the controller node.

(2) In every time-slot, all the nodes send their *packet_value* to the controller node.

(3) The controller node responds back to the proposing nodes with an ACK or a NACK depending on the proposing node's *packet_value* with respect to that of the other nodes. The ACKing and NACKing is done within the duration of the same time-slot (in which the *packet_value* was sent).

(4) The node that receives an ACK, transmits its data through the duration of the next time-slot.

We now discuss how to compute delay experienced by a packet waiting to be transmitted over this light-trail.

Multiple flows may be provisioned at a node (each with different destinations). The *packet_value* is computed over all the flows provisioned at a single node.

Bandwidth assignment: The largest *packet_value* that the controller receives is denoted by:

$$mpacket\_value_k(t) = \max_{\forall i:N_i \in k}(packet\_value_{ik}(t)) \quad (30)$$

While the node that gets rights to form the next connection is denoted by:

$$suc_k(t) = \arg\max_{i:N_i \in k}(packet\_value_{ik}(t)) \quad (31)$$

The winning node ($suc_k(t)$) would transmit (establish a connection) in the next ($t+1^{st}$) time-slot.

Note: It may happen that a flow at a GW may not be serviced within its permissible delay limit or a buffer may reach an overflow condition. Such a flow or such a buffer implies an event called recourse that requires the algorithm to re-evaluate the flow and provision it through either some other light-trail (wavelength) or create a new light-trail. Recourse occurs when:

$$t_{HP} < \gamma_{ik}(t) < \left(t_{HP} + T_S\right) \quad \text{or}$$

$$\left[1 - \frac{Buff_{GW}^{i}(t)}{B_{\max}}(t_{HP})\right] > B_{ik}(t) > \left[1 - \frac{Buff_{GW}^{i}(t)}{B_{\max}}(t_{HP} + T_S)\right] \quad (32)$$

This condition requires new light-trails to be set up.

To compute the delay we are interested in computation of the probability that a particular node gets transmission rights in a time-slot at steady state. This probability of success of a node *i* in light-trail *k* is now computed. Let T_s be the time-slot duration, and let T_g be the guard-band between two successive time-slots. We assume T_g to have a very small value and hence neglect the effect of T_g on the overall light-trail delay.

Let $P_{succ}^{ik,FC}(t)$ be the probability of success of node *i* in light-trail *k* in the FC direction. Let $P_{succ}^{k-ing,RC}(t)$ be the probability of success for the router at the convener of light-trail *k* in sending data into the light-trail (i.e. in the RC direction). The superscript *ing* represents the ingress port for the light-trail *k* which is at the IP router. The delay in the FC direction by any arbitrary node *i* in light-trail *k* is given by:

$$\delta_{LT}^{FC} = \frac{T_S}{P_{succ}^{ik,FC}(t)} \quad (33)$$

In an *n(k)*-node light-trail a node *i* has to contend with the other *n(k)-1* nodes that also includes the router (at the convener node) for arbitration. Hence the probability that node *i* will successfully send data into the light-trail is the probability that the *packet_value* sent by node *i* is greater than the *packet_value* sent by *any* other nodes *and* the IP router interface at the convener node. This condition is denoted by:

$$P_{succ}^{ik,FC}(t) = p(packet\_value_{ik}^{FC}(t) > packet\_value_{k-ing}^{RC}(t)).$$
$$\prod_{\substack{j=1 \\ j \neq i}}^{n(k)} p(packet\_value_{ik}^{FC}(t) > packet\_value_{jk}^{FC}(t)) \quad (34)$$

The *packet_value* computation is denoted in (1) and (2) from which we compute the probability that the *packet_value* sent by node *i* is the largest as:

$$P_{succ}^{ik,FC}(t) = \left(1 - p\left[\frac{\gamma_{ik}(t)}{\max\limits_{\forall q}(y_{ikq}(t))} \leq \frac{\overline{\gamma_{k-ing}}(t)}{\max\limits_{\forall q}\left(\overline{y_{k-ingq}}(t)\right)}\right]\right).$$
$$\prod_{\substack{j=1 \\ j \neq i}}^{n(k)}\left(1 - p\left[\frac{\gamma_{ik}(t)}{\max(y_{ikq}(t))} \leq \frac{\gamma_{jk}(t)}{\max\limits_{\forall q}(y_{jkq}(t))}\right]\right) \quad (35)$$

The above equation is simplified in [10] and we state the result duly modified for our case taking into consideration

duplex communication over the simplex light-trails (time shared FC and RC) as:

$$P_{succ}^{ik,FC}(t)=\left(1-\sum_{\phi=0}^{1}\delta(\bar{\phi})\right).\prod_{\substack{j=1\\ j\neq i}}^{n(k)}\left(1-\sum_{\phi=0}^{1}\delta(\phi)\right) \quad (36)$$

where, $\bar{\phi}=\dfrac{\overline{\gamma_{k-ing}}(t)/\max_{\forall q}\left(\overline{y_{k-ingq}}(t)\right)}{\gamma_{ik}(t)/\max_{\forall q}(y_{ikq}(t))}$ and $\phi=\dfrac{\gamma_{jk}(t)/\max_{\forall q}(y_{jkq}(t))}{\gamma_{ik}(t)/\max_{\forall q}(y_{ikq}(t))}$

$$P_{succ}^{ik,FC}(t)=\left(1-\sum_{\phi=0}^{1}\delta(\bar{\phi})\right).\prod_{\substack{j=1\\ j\neq i}}^{n(k)}\left(1-\sum_{\phi=0}^{1}\delta(\phi)\right)=\left(1-\int_{0}^{1}\delta(\phi')d\phi'\right).\prod_{\substack{j=1\\ j\neq i}}^{n(k)}\left(1-\int_{0}^{1}\delta(\phi)d\phi\right) \quad (37)$$

and

$$\delta(Q)=\sum_{v=t_{HP}}^{\max_{q\in Buff_{GW}^{q}(t)}(\Delta_q)} p\left(\frac{\gamma_{ik}(t)}{\max_{\forall q}(y_{ikq}(t))}=vQ\right).p\left(\frac{\gamma_{jk}(t)}{\max_{\forall q}(y_{jkq}(t))}=v\right)v \quad (38)$$

Leading to:

$$\delta_{LT}^{FC}=\frac{T_S}{P_{succ}^{k-ing,FC}(t)} \quad (39)$$

The average delay experienced by a packet in the RC direction is computed in a similar fashion:

$$P_{succ}^{k-ing,RC}(t)=\prod_{j=1}^{n(k)}p(packet\_value_{k-ing}^{RC}(t)>packet\_value_{jk}^{FC}(t)) \quad (40)$$

$$P_{succ}^{k-ing,RC}(t)=\prod_{j=1}^{n(k)}\left(1-p\left[\frac{\overline{\gamma_{k-ing}}(t)}{\max_{\forall q}\left(\overline{y_{k-ingq}}(t)\right)}\leq\frac{\gamma_{jk}(t)}{\max_{\forall q}(y_{jkq}(t))}\right]\right) \quad (41)$$

$$P_{succ}^{k-ing,RC}(t)=\prod_{j=1}^{n(k)}\left(1-\sum_{\phi=0}^{1}\delta(\phi')\right)=\prod_{j=1}^{n(k)}\left(1-\int_{0}^{1}\delta(\phi')d\phi'\right) \quad (42)$$

Where, $\phi'=\dfrac{\gamma_{jk}(t)/\max_{\forall q}(y_{jkq}(t))}{\overline{\gamma_{k-ing}}(t)/\max_{\forall q}(\overline{y_{k-ingq}}(t))}$

The delay in the RC direction is given by:

$$\delta_{LT}^{RC}=\frac{T_S}{P_{succ}^{k-ing,RC}(t)} \quad (43)$$

The delays computed in (28), (29) for WiMax, (33), (43) for light-trail are plugged in (3)-(5) to obtain final end-to-end delay, neglecting propagation delay.

VI. RELATED WORK

The concept of providing broadband connectivity to moving users especially fast moving users has been studied in [8], [12], [13], of which the [12] is our own preliminary work.

The approach in [12] proposed the basic architecture – using the light-trail node to serve as an element to provide dynamic provisioning and optical multicasting. This work however did not outline the technology used at the wireless layer nor did it analyze the system for delay and utilization. No protocol was presented for delivery of bandwidth to the end-user through the unified optical-wireless network.

The work in [8, 13] mentions two approaches – the mobile cell concept in [13] and the moving tunnel concept in [8]. While this fundamental work provides a prelude to the problem as well as outlines two solutions, our work differs from the aforementioned two approaches in terms of defining a technology solution. We present a complete end-to-end optical + wireless solution that spans across the network, data and physical layers using contemporary or available technology. In [8, 13] a radio-over-fiber solution is presented – which does not delve into the specifics in terms of architecture, protocol or working. In contrast, our solution defines the node architecture, protocol as well as how to provision bandwidth from user to the network. The solution in [8, 13] uses switching which is detrimental (due to switching speed/uncertainty) in providing bandwidth to fast moving users. The requirement of optical switching in [8, 13] means either that hand-off would suffer (due to infancy of optical technology) or that we require futuristic high-speed optical switches. The light-train scheme relies on existing available, mature technologies, and is glued together by an out-of-band protocol designed to provision (without optical switching) as well as facilitate hand-off at the wireless layer. Our method of integration of optical and wireless technologies, enabling a symbiotic relationship is the key differentiator between the light-train solution and other existing solutions.

VI.A. Qualitative argument for Light-trails as opposed to conventional solutions:

In this subsection we outline the differences and advantages of light-trails as opposed to conventional point-to point circuit technology.

(1) The 2-layer (data + control) light-trail enables dynamic sub-wavelength provisioning of bandwidth by fostering a unique node architecture that supports an optical bus, while using the standard ITU defined OOB control channel to dynamically arbitrate bandwidth within the bus. The dual layered approach allows fast configuration (provisioning) of bandwidth, while also supporting optical multicasting.

(2) Dynamic provisioning is an essential requirement for fast-hand offs – typically those envisioned for such applications like fast moving trains. The fast hand-off concept stems from two critical requirements: (a) reduction of time allocated for hand-off to maintain seamless and efficient communication under a large ratio of train size to cell-width, and (b) due to the high velocity of the train itself – requiring dynamic provisioning of bandwidth to efficiently utilize the wireless channel (if the time required for hand-off is large, then time lost in provisioning the connection when a train moves from one cell to its adjacent neighbor is also large and hence the efficiency of the system degrades). A fast hand-off concept is investigated by us in [14].

(3) The light-trail provides optical multicasting support. In conjunction with the dynamic provisioning property, this feature acts as a further enabler for seamless communication in the overlaid wireless channel. With optical multicasting support, when a train moves from the range of an ingress cell site to the adjacent egress cell site (GW), the data is readily

available at the egress GW. This means that there is no requirement for further signaling to set up a new connection between the network core router and the new GW, and this eliminates the need for pre-fetching. In the conventional case, i.e. using an Ethernet network like the one proposed in [8, 13] or using a backbone SONET/SDH network, we would require layer 3 termination [13] and further provisioning of a moving tunnel type concept. The time required would result in loss of bandwidth and hence degradation of system efficiency.

Our model based on three factors, namely, (a) absence of pre-fetching due to optical multicasting (b) dynamic provisioning of bandwidth within the light-trail bus and (c) the presence of an OOB control that allows an egress GW to accurately estimate the starting and ending times of the train within its range (and hence schedule data accordingly), are key to a low-latency high-efficiency and evolutionary approach to integrated optical-wireless design for support of fast moving and bandwidth intensive users.

VII. SIMULATION AND RESULTS

To demonstrate the working of our protocol for light-trains, we developed a discrete event simulation (DES) model in C#. The topology simulated is based on the European landscape (Fig. 14) with sixteen major cities and distances not assumed to scale but assumed that the cities are within one huge metro region. The train contained end-users that supported applications on their mobile devices. These applications included voice, data and video (multimedia) services, each of which assumed Ethernet frame generation at layer-2. Ethernet frame arrival at a user followed a Poisson arrival process.

The basic event-duration for the DES model was 1 millisecond and is the time required by the system to transmit 6 Ethernet frames (MTU=1500 bytes) at 70 Mbps over the WiMax channel. Light-trail line-rate was assumed at 1 Gbps. A typical light-trail had between 20 and 25 GWs (with reduced attenuation by deploying Wavelength Selectable Switch support). Train velocity was pegged 200 km per hour. The DES was based on a C# class library.

The average distance between two gateways was assumed to be 1.5 km. Each Ethernet frame object belonged to a service class (voice, video and data) that also specified the maximum allowable delay (through VLAN tags). In the simulation, users boarded the train at different stations along the train routes. Simulation was performed under different values of cumulative network load.

Load is computed as the ratio of total number of bits per-second that entered the network to the maximum number of bits that the network could support.

In Fig. 15 we show utilization versus load. Two cases are considered in which the average file size is 100 kb and 200 kb respectively. The load for both the cases is varied from 0.1 to 0.9. We observe that in case 1 as the load varies from 0.1 to 0.9 the effective utilization varies from 10% to 65% while for case 2 the utilization varies from 10% to 60%. The curves in both the cases increase with load, linearly for most parts but are logarithmic at high loads. This implies that light-trains are not only able to cope well at low and medium loads but also achieve good utilization at high loads. Utilization is computed as the average time the system is busy.

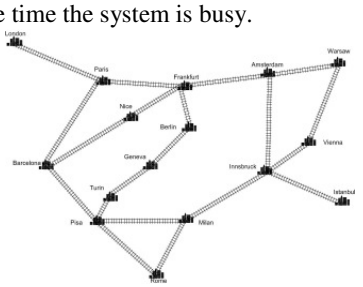


Fig. 14. European simulation topology.

Fig. 16 shows the delay experienced in FC and Fig. 17 shows the delay experienced in RC as the load varies from 0.1 to 0.9. In both directions of communication the average per packet delay is of the order of 100 *us*. We observed that the FC delay is lower than the RC delay (Fig. 17). This is explained by the fact that the end-users in the train instantaneously send generated data into the core thus occupying a larger percentage of the WiMax transmission slot as compared to the IP-router initiated communication (RC). This result is quite counter-intuitive as compared to typical TCP traffic and is explained by the manner in which bandwidth allocation is done in the WiMax channel, taking AP buffer values into consideration in conjunction with BS (GW) values). Fig. 18 shows the average response time per file requested by the end-user; the average response time increases almost linearly as the load increases and is the order of 2-3 ms. Average response time is defined as the time required for the system to fetch the first packet in a file since a request has been made.

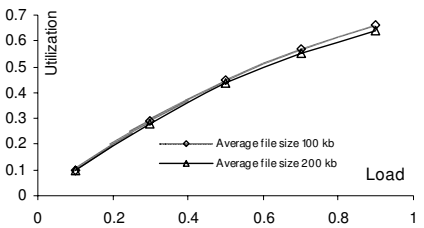


Fig. 15. Utilization versus load

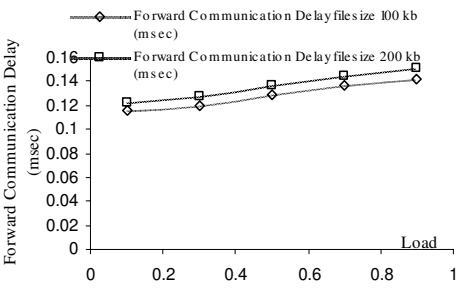


Fig. 16: Forward Communication delay versus load

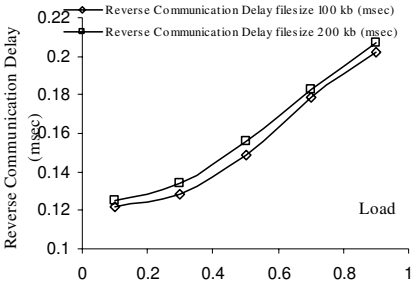


Fig. 17. Reverse Communication delay versus load

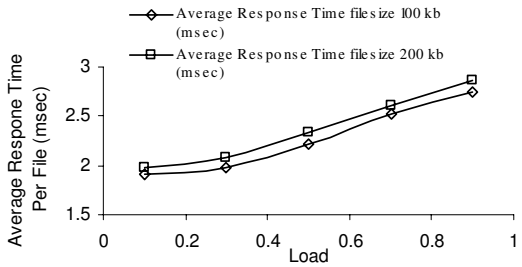


Fig. 18. Average response time versus load

VIII. CONCLUSION

In this paper we have proposed an efficient framework to provide bandwidth-on-demand to moving metro trains in a hybrid (integrated) wireless-optical environment called *light-trains*. In particular, we introduce a new node architecture that provides for cross-medium design between wireless and optical domains. Namely, at the optical layer we use modified light-trail technology that provides bandwidth-on-demand to nodes with each node supporting a wireless gateway. The wireless GW supports a point-to-point service aware WiMax system that facilitates provisioning of bandwidth to a fast-moving train. The light-trail optical layer and WiMax wireless layer are conjoined to support ultra-fast hand-off that is absolutely essential in provisioning of this bandwidth. To do so, we also propose a protocol that enables dynamic bandwidth provisioning across the multiple physical mediums, detailing message types and hand-off functionalities. This protocol is analyzed from a delay perspective. Finally, a sample simulation study is also presented to numerically gauge delay as a function of load for different services and also comparison with legacy SONET/SDH networks is shown.

REFERENCES

[1] A. Gumaste, "*Light-trail and Light-frame Architecture for Optical Networks,"* Ph. D thesis Fall 2003 UT-Dallas.

[2] A. Gumaste and I. Chlamtac, "Light-trails: An IP Centric Solution in the Optical Domain," Proc. of HPSR 2003 Torino Italy.

[3] A. Gumaste and I. Chlamtac, "Light-trails: An Optical Solution for IP Transport, "Invited Paper, *OSA Journal on Optical Networking* May 2004, 864-891.

[4] A. Lodha, A. Gumaste, P. Bafna and N Ghani, "Stochastic Optimization of Light-trails using Benders Decomposition," *IEEE HPSR High Speed Switching and Routing Conf.* June 2007, Brooklyn NY.

[5] A. Gumaste, I. Chlamtac and J. Jue, "Light-frames: A Pragmatic Framework for Conducting Optical Packet Transport," Proc of *IEEE Int'l Conf. on Communications*, (ICC) Paris 2004.

[6] B. Sikdar "Delay Analysis of IEEE 802.11 PCF MAC based Wireless Networks," *Proceedings of IEEE GLOBECOM*, St Louis, MO, November 2005.

[7] R. Iyengar, P. Iyer and B. Sikdar, "Delay Analysis of 802.16 based Last Mile Wireless Networks," *Proceedings of IEEE GLOBECOM*, St. Louis, MO, November 2005.

[8] F. De Greve, B. Lannoo, L. Peters, T. Van Leeuwen, F. Van Quickenborne, D. Colle, F. De Turck, I. Moerman, M. Pickavet, B. Dhoedt, P. Demeester, "FAMOUS: A network architecture for delivering multimedia services to fast moving users", *Wireless Personal Communications* , ISSN 0929-6212, published by Springer, Vol. 33, Nr. 3-4, June 2005, pp.281-304.

[9] A. Gumaste, N. Ghani, P. Bafna, A. Lodha, S. Srivastava, T. Das and S. Zheng, "Achieving Multi-Rate Dynamic Sub-Wavelength Service Provisioning in Strongly connected Light-trails (SLiTs)," *PDP (Post Deadline Paper) IEEE/OSA OFC Optic Fiber Conference*, Anaheim, CA March 2007

[10] A. Gumaste, T. Das, P. Bafna and N. Ghani, "Two Stage Auctioning Protocol for Light-trail Design and Topology Growth in WDM Ring Networks," to IEEE Journal of Light-wave Technology 2007

[11] F. Kelly, A. Maulloo and D. Tan, "Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability," *Journal of Operations Research* 1998, pp. 237-252.

[12] A. Gumaste, N. Ghani and S. Q. Zheng, "Light-trains: A Cross Layer Delivery Mechanism for High Bandwidth Applications in Moving Metro Trains," 41st IEEE Int'l Conf. on Commun. (ICC) 2006 Istanbul Turkey.

[13] B. Lannoo, D. Colle, M. Pickavet and P. Demeester, "Optical Switching Architecture to Implement Moveable Cells in a Multimedia Train Environment", in Proceedings of the 30th European Conference on Optical Communication (ECOC 2004), Stockholm, Sweden, September 2004.

[14] A. Lodha et. al., "Generalized ROADM/Light-trail Assisted Hand-Off for Wireless Metro Network" 10th Wireless Personal Multimedia Communications WPMC 2007, Jaipur India December 2007.

Dr. Ashwin Gumaste is currently the James R. Isaac Chair and faculty member in the Department of Computer Science and Engineering at the Indian Institute of Technology (IIT) Bombay. He is currently also a consultant to Nokia Siemens Networks, Munich where he works on optical access standardization efforts. He was a Visiting Scientist with the Massachusetts Institute of Technology (MIT), Cambridge, USA in the Research Laboratory for Electronics from 2008 to 2010. He was previously with Fujitsu Laboratories (USA) Inc in the Photonics Networking Laboratory (2001-05). He has also worked in Fujitsu Network Communications R&D (in Richardson TX) and prior to that with Cisco Systems in the Optical Networking Group (ONG). His work on light-trails has been widely refered, deployed and recognized by both industry and academia. His recent work on Omnipresent Ethernet has been adopted by tier-1 service providers. Ashwin has 20 granted US patents and over 30 pending patent applications. Ashwin has published about 120 papers in refered conferences and journals. He has also authored three books in broadband networks called DWDM Network Designs and Engineering Solutions (a networking bestseller), First-Mile Access Networks and Enabling Technologies and Broadband Services: User Needs, Business Models and Technologies for John Wiley. Owing to his many research achievements and contributions, Ashwin was awarded the Government of India's DAE-SRC Outstanding Research Investigator Award in 2010 as well as the Indian National Academy of Engineering's (INAE) Young Engineer Award (2010). He has served Program Chair, Co-chair, Publicity chair and workshop chair for IEEE conferences and as Program Committee member for IEEE ICC, Globecom, OFC, ICCCN, Gridnets etc. Ashwin is also a guest editor for IEEE Communications Magazine, IEEE Network and the founding Editor of the IEEE ComSoc ONTC's newsletter Prism. He is the Chair of the IEEE Communication Society's Technical Committee on High Speed Networks (TCHSN) 2011-2013. He has been with IIT Bombay since 2005 where he convenes the Gigabit Networking Laboratory (GNL):

www.cse.iitb.ac.in/gnl. The Gigabit Networking Laboratory has secured over 8 million USD in funding since its inception and has been involved in 4 major technology transfers to the industry. Ashwin can be reached through www.ashwin.name.

Akhil Lodha is currently working as Quantative Analyst on the Electronic Options Trading desk at Citigroup, New York. He completed MS in Computational Finance from Carnegie Mellon University and Dual degree from IIT, Bombay.

Saurabh Mehta is currently pursuing his PhD from Gigabit Networking Laboratory in the Department of Computer Science and Engineering, IIT Bombay. Earlier, he worked with Center for Development of Advanced Computing (C-DAC), Pune for the development of PARAM series of Supercomputers. His research interests include access and metro networks, high-performance computing and real-time systems.

Dr. Jianping Wang is currently an assistant professor in the Department of Computer Science at City University of Hong Kong. She received her BSc and MSc degrees from Nankai University in 1996 and 1999 respectively, and her Ph.D. degree from University of Texas at Dallas in 2003. Prior to joining CityU, she worked at Georgia Southern University and University of Mississippi as an assistant professor.

Dr. Nasir Ghani is an Associate Professor in the Electrical and Computer Engineering Department at the University of New Mexico and Director of the UNM Advanced Cyberinfrastructure Lab. He received the Bachelors degree in Computer Engineering from the University of Waterloo, Canada, the Masters degree in Electrical Engineering from McMaster University, Canada, and the Ph.D. degree in Computer Engineering from the University of Waterloo, Canada.

Determination of Low Pass Filter Coefficients for Receiver with Zero Intermediate Frequency by Differential Evolution Algorithm

Martin Vestenický, Peter Vestenický, Vladimír Hottmar

***Abstract* — In the paper the method of a low pass filter design for a receiver with zero intermediate frequency is presented. Definition of design criteria is performed on the system level instead of specific filter features determination. The design process uses differential evolution algorithm. Main part of the paper describes a creation of cost function and the method of its application in cooperation with the used algorithm. Conclusion of the paper presents a found solution and its comparison with commonly used filters.**

***Index Terms* — differential evolution, low pass filter, transfer function, stochastic algorithm**

I. INTRODUCTION

THE continuously increased demands on the electronic circuits being used in telecommunications impose raising requirements on their optimal design taking technological, economical, construction and other limitations into account. Therefore research must be focused on new methods for optimized design of electronic circuits [2]. One of new approaches to this challenge are the stochastic algorithms. These algorithms are being used to numerical solving of complex optimization problems. The main advantage of stochastic algorithms is the easy applicability in many areas of research and design. Next major feature is the ability of effective solving of multi-criterion optimization problems with a large number of independent variables. One of application fields for using the stochastic algorithms is the optimized design of functional blocks for communication systems.

This paper deals with determination of coefficients for transfer function of a low pass filter which is applied in a zero intermediate frequency receiver whose block diagram is shown in Fig. 1. Such receiver structure has been published in [4]. In the design process the well known algorithm of differential evolution (DE) described in detail in [1] has been used.

Martin Vestenický, Faculty of Electrical Engineering, Department of Telecommunications and Multimedia, University of Žilina. Univerzitná 8215/1, 010 26 Žilina, Slovakia (e-mail: martin.vestenicky@fel.uniza.sk).

Peter Vestenický, Faculty of Electrical Engineering, Department of Control and Information Systems, University of Žilina. Univerzitná 8215/1, 010 26 Žilina, Slovakia (e-mail: peter.vestenicky@fel.uniza.sk).

Vladimír Hottmar, Faculty of Electrical Engineering, Department of Telecommunications and Multimedia, University of Žilina. Univerzitná 8215/1, 010 26 Žilina, Slovakia (e-mail: vladimir.hottmar@fel.uniza.sk).

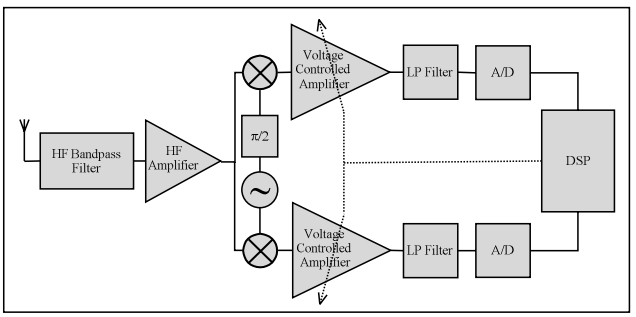


Fig. 1. Block diagram of receiver with zero intermediate frequency

II. APPLICATION OF DE FOR DETERMINATION OF FILTER COEFFICIENTS

The application objective of DE algorithm is to determine the coefficients of a low pass filter transfer function which for the given type of modulation and for the given signal-to-noise ratio minimizes the symbol error ratio of data transfer between transmitter and receiver taking some limitation into account: fixed order, causality and stability of filter. This type of task is in [4] referred to as highly topical.

For this application the sixth order switched capacitor or switched current filter is assumed. The transfer function of this filter is given by formula (1). For using DE algorithm the association of particular variables (filter coefficients) to chromosomes of individual have to be defined (Fig. 2). Therefore, the individual is represented by fourteen element vector where every element represents one coefficient of a searched transfer function in the form of real number without other limitations.

$$H(z)=\frac{b_6\cdot z^6+b_5\cdot z^5+b_4\cdot z^4+b_3\cdot z^3+b_2\cdot z^2+b_1\cdot z+b_0}{a_6\cdot z^6+a_5\cdot z^5+a_4\cdot z^4+a_3\cdot z^3+a_2\cdot z^2+a_1\cdot z+a_0} \quad (1)$$

The only modification of DE algorithm compared to the version described in [1] is the method of a new individual creation which is given by formula (2). This modification which is in detail described in [6] enabled a particular increase of convergence speed against the original implementation of DE algorithm.

The formula (2) describes the creation of *j*-th chromosome of a new individual as linear combination of *j*-th chromosomes of three randomly selected individuals from current population and *j*-th chromosome of the best created individual till. The algorithm input constants are summarized in Table I and their meaning is explained in detail in [1], [5], [6] and [7]. The values of constants have been selected taking conclusions from [5] into account. The criterion for algorithm termination is the maximum number of generations G_{max}.

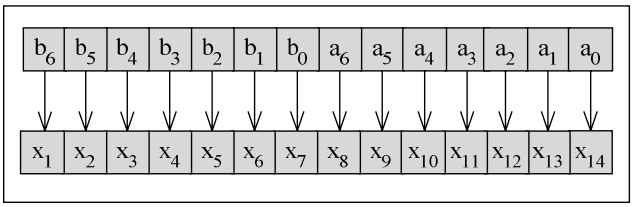


Fig. 2. Association of variables to chromosomes of individual

$$temp_j = \begin{cases} POP_{r_3,j,G} + F_1 \cdot \left(POP_{r_1,j,G} - POP_{r_2,j,G}\right) + \\ \quad + F_2 \cdot \left(best_j - POP_{r_3,j,G}\right) & \text{if } rand_j[0,1] < CR \vee j = j_{rand}\ j = 1,\ldots,D \\ POP_{i,j,G} & \text{else} \end{cases} \quad (2)$$

III. DEFINITION OF COST FUNCTION

The most complex problem in solving such a type of design task is to define the cost function. In this case the cost function has been built by a non-standard method, namely by using the simulation model of a whole transfer chain which is shown in Fig. 3. The model serves for searching close-optimal coefficients of a transfer function given by the general formula (1). This model consists of three basic parts:

- The transmitter of QAM signal which generates random QAM signal with a defined number of states,
- Data transmission channel which simulates signal transmission in real environment, in this case the modified AWGN (Additive White Gaussian Noise) channel was used,
- The receiver whose part is the designed filter. Other functional blocks are parts of the receiver, too.

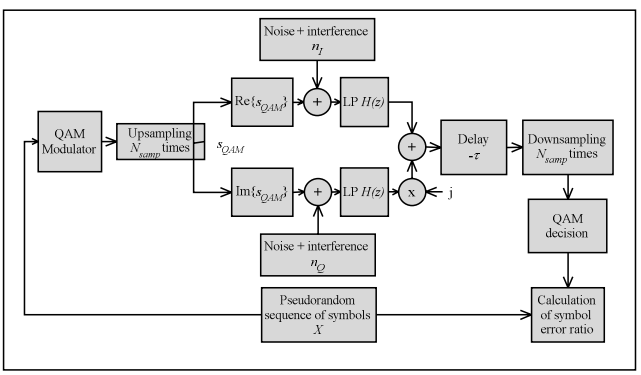


Fig. 3. System model of transfer chain

The model uses signal processing in base band for simplicity. It does not cause significant deviation of the results compared with reality [3]. The model is universal with respect to the number of QAM states *M*. It works with pseudorandom sequence of symbols *X* which can be assumed as a sequence of pseudorandom integer numbers from interval <0, *M*-1> where *M* is the number of QAM modulation states. This sequence is used as an input parameter for the mathematical model of QAM modulator which associates the corresponding points in signal space to particular members of the sequence according to a predefined signal point constellation. By this procedure the QAM modulated signal is being created in base band which is mathematically represented as a complex number sequence. The length of this sequence is the same as the length of the sequence *X*. It is necessary for every symbol from the sequence *X* to assign N_{samp} of samples. In that way the sequence of signal samples s_{QAM} is created. The signals assigned to I (In phase) and Q (Quadrature) planes are obtained by splitting of every s_{QAM} sequence member on real and imaginary components.

The transfer via the radio channel is mathematically modeled by adding of signals representing noise and interference which are given as the sequence of samples n_I and n_Q for I and Q planes respectively. In that way the testing signals have been created and the ones will be used for testing of solution represented by actual transfer function *H(z)*. After processing of both signals from I and Q planes by low pass filters and after mathematical elimination of filter delay τ the signal samples are added. Before adding the samples from the Q plane they are multiplied by imaginary unit *j*. The calculation of delay τ is given by formula (3), where *N* is the number of samples of the received signal T_{sig} before filtration, *Pad* is the number of zero samples which must be added to the original sequences for correct working of the simulation of filtering process, and $T_{sigfilt}$ is the received signal after filtration.

$$\tau = N + Pad - \max\left(corr\left(\vec{Tsig}, \vec{Tsigfilt}\right)\right) \quad (3)$$

The sequence of complex number has been created by the described procedure. This sequence is in the next step downsampled N_{samp} times. A new sequence of complex numbers has its length identical with the sequence *X* and represents the sequence of points in signal space at receiving side. It is an input argument for the mathematical model of QAM decision. The QAM decision assigns an integer number from interval <0, *M*-1> to every member of the input sequence and creates a sequence of received symbols. By comparing both transmitted and received sequences the symbol error ratio *SER* is calculated by formula (4). The symbol error ratio is the value of a cost function for actual solution represented by coefficients of the transfer function *H(z)*.

$$SER = \frac{N_{err}}{N_{symb}} \quad (4)$$

In the course of DE algorithm the cost function has been calculated for every new originated individual by the previously described process. At every cost cycle approximately N_{symb}=10^4 of symbols were transferred. N_{err} is the number of incorrectly received symbols. This value is a compromise between calculation time and accuracy of cost function.

TABLE I
VALUES OF CONTROL CONSTANTS

Parameters of model			Control constants of DE algorithm		
Transfer rate	*DR*	1 Mbps	Number of unknown quantities	*D*	14
Number of symbols	N_{symb}	10000	Maximum number of generations	G_{max}	100
Number of samples per symbol	N_{samp}	8	Number of population individuals	*NP*	280
Number of modulation states	*M*	16	Crossover ratio	*CR*	0.9
Signal - to - noise ratio	*SNR*	18 dB	Weighting coefficient of differential mutation 1	*F1*	0.5
Output signal power	P_{out}	1 W	Weighting coefficient of differential mutation 2	*F2*	0.5
Frequency boundary of interference	*f*	275 kHz			

IV. RESULTS AND DISCUSSION

Applying the DE algorithm, the transfer function (5) which does not correspond to any standard approximation was found. Moreover, no requirements for the filter must be predefined except for its order.

$$H(z)=\frac{1.0722\cdot z^6+1.7945\cdot z^5+2.5056\cdot z^4+2.6168\cdot z^3+2.7587\cdot z^2+1.7411\cdot z+1.5783}{43.170\cdot z^6-33.247\cdot z^5+1.2873\cdot z^4-4.9585\cdot z^3+4.8114\cdot z^2+1.8533\cdot z+0.66119} \quad (5)$$

The requirements for filter stability and causality are tested during the optimizing process. These requirements are necessary for its correct operation and ability to be realized by circuits based on switched capacitor or switched currents technology. The check analysis indicates that the found transfer function (5) is stable and causal. Frequency response of the found filter is shown in Fig. 4 together with frequency response of the matched filter which is calculated for rectangular pulses and modulation rate $DR/\log_2 M$=250 kBd (see Table I).

In Figs. 5a and 5b the constellations of signal points of the testing signal and signal after filtering by a filter with transfer function (5) are presented. These constellations were created for 10^4 symbols and for signal - to - noise ratio 18 dB.

By using of the found filter the bit error ratio is significantly decreased (about one or two orders) in comparison with filters with standard approximations. This comparison is shown in Fig. 6a and Fig. 6b. In this case the simulation was performed for a communication channel with power spectral density of noise which is shown in Fig. 7, i. e. it is not a standard AWGN channel. Theoretical estimation of the bit error ratio for assumed type of modulation was performed according to the formula (6), where E_S is the energy of one symbol and N_0 is the power spectral density of added white noise. The bit error ratio for all simulations was calculated from the symbol error ratio by the formula (7), where *M* is the number of modulation states. Selection of filters for comparison was inspired by experiments presented in [3], [4]. Parameters of compared filters are summarized in Table II.

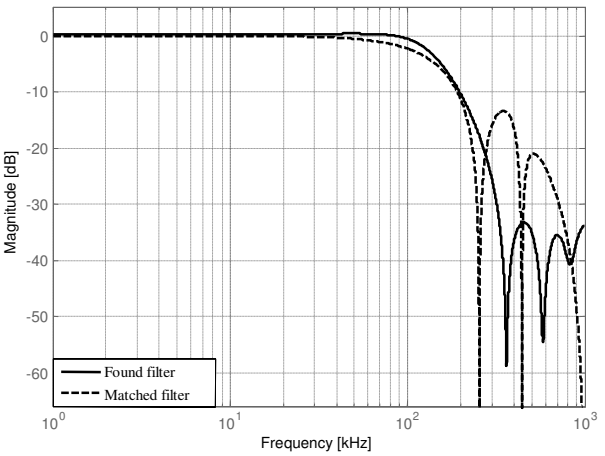


Fig. 4. Frequency response of found filter and matched filters

$$BER_{Theor}=\frac{2\cdot\left(1-\frac{1}{\sqrt{M}}\right)\cdot erfc\left(\sqrt{\frac{3}{2\cdot(M-1)}\cdot\frac{E_s}{N_0}}\right)-\left(1-\frac{1}{\sqrt{M}}\right)^2\cdot erfc^2\left(\sqrt{\frac{3}{2\cdot(M-1)}\cdot\frac{E_s}{N_0}}\right)}{\log_2 M} \quad (6)$$

$$BER=\frac{SER}{\log_2 M} \quad (7)$$

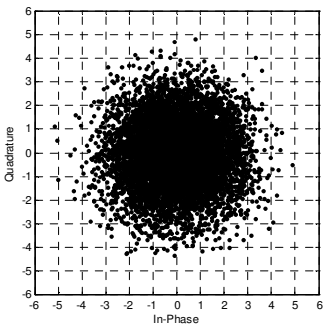


Fig. 5a. Constellation diagram before filtration

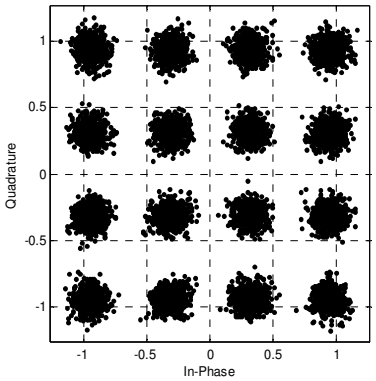


Fig. 5b. Constellation diagram after filtration

TABLE II
PARAMETERS OF COMPARED FILTERS

Type of filter	Filter order	Cut-off frequency [kHz]	Passband ripple [dB]	Stopband ripple [dB]	Roll-off factor
Inverse Chebyshev	6	250	-	40	-
Cauer	6	250	0.1	44	-
Raised cosine	6	-	-	-	0.5
Matched filter	6	250	-	-	-
Theoretical guess	-	250	-	-	0
Found filter	6	-	-	-	-

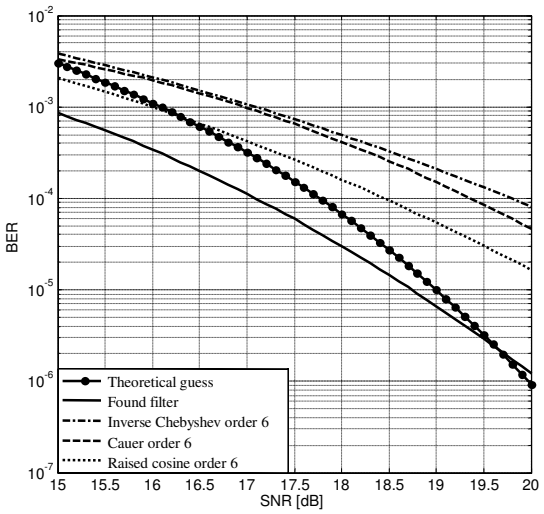


Fig. 6a. Comparison of BER for various filters and non-standard noise channel

For comparison, simulation of BER was performed for standard AWGN channel, too. The results are shown in Fig. 8 for the found filter and matched filter.

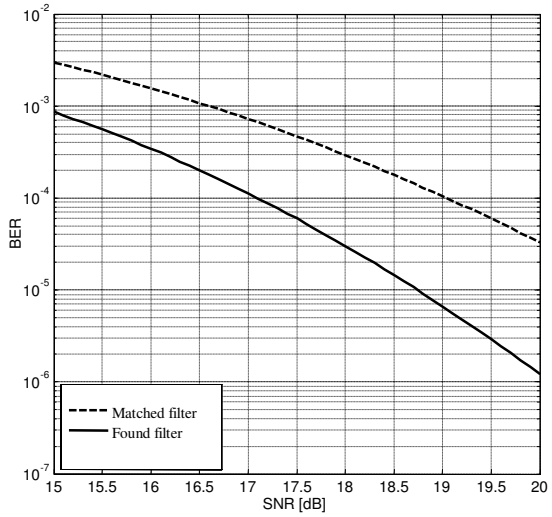


Fig. 6b. Comparison of BER for matched filter, found filter and non-standard noise channel

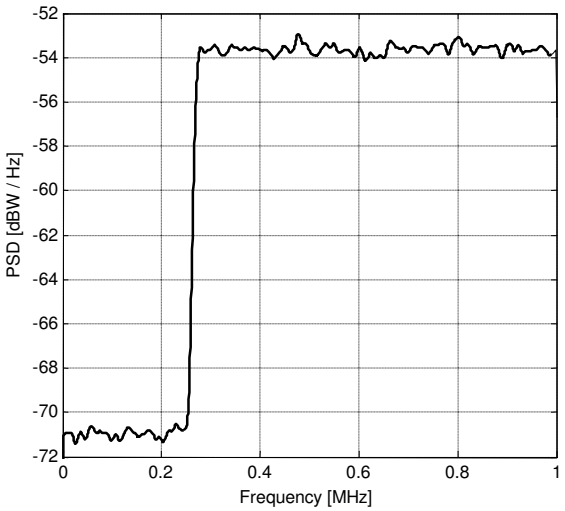


Fig. 7. Power spectral density for non-standard noise channel

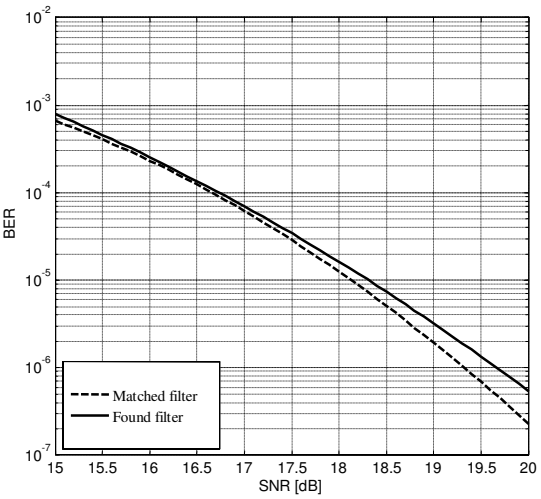


Fig. 8. Comparison of BER for standard AWGN channel

V. CONCLUSION

Application of stochastic algorithms enabled to put into the filter design various and often contradictory requirements which are hardly implementable simultaneously by standard design procedures. Based on the attained results it can be assumed that stochastic algorithms are applicable to the other functional blocks of the system, too. Essentially the whole system can be designed based only on predefined requirements on the functionality of whole system without definition of requirements for its particular functional blocks. Slight disadvantage of stochastic algorithms is a considerable time consumption of the design process. This disadvantage can be eliminated by parallel processing of stochastic algorithm described in [7].

REFERENCES

[1] Corne, D., Dorigo, M., Glover, F., "New Ideas in Optimization," McGraw-Hill, London 1999, ISBN 001-709506-5

[2] Craninckx J., Donnay S., "4G Terminals: How Are We Going to Design Them?" Proceedings of the 40th Design Automation Conference 2003, pp. 79 - 84, June 2003, ISBN 1-58113-688-9

[3] Emira A. A., Valdes-Garcia A., Xia B., Mohieldin A. N., Valero-Lopez A. Y., Moon S. T., Xin C., Sánchez-Sinencio, E., "Chameleon: A Dual Mode 802.11b/Bluetooth Receiver System Design." *IEEE Transaction on Circuits and Systems - I: Regular Papers*, Vol. 53, No. 5, pp. 977-984, May 2006, ISSN 1057-7122

[4] Korotkov A. S., Morozov D. V., Tutyshkin A. A., Hauer H., "Channel Filters for Microelectronic Receivers of Wireless Systems." IEEE 7th CAS Symposium on Emerging Technologies: Circuits and Systems for 4G Mobile Wireless Communications, pp. 24 - 31, St. Petersburg State Polytechnic University, Russia 2005, ISBN 5-7422-0895-2

[5] Price, K., Storn, R., M., Lampinen, J., A., "Differential Evolution: A Practical Approach to Global Optimization." Springer, 2005. ISBN 978-3-540-20950-8.

[6] Vestenický, M., "The Use of Evolutionary Algorithms in Selective Systems Synthesis for Communication Systems." Dissertation thesis, University of Žilina, Faculty of Electrical Engineering, 2007 (in Slovak)

[7] Vestenický M., Vestenický P., "The Use of MDCE Environment for Parallel Processing of Stochastic Algorithms." *Annals of Faculty Engineering Hunedoara – International Journal of Engineering*, vol. 8, nr. 2/2010, pp. 55 - 60. ISSN 1584-2665

Martin Vestenický was born in Martin, Slovakia in 1980. He received his Ing. (M.Sc.) degree in 2003 and his Ph.D. degree in 2008 both at the Department of Telecommunications and Multimedia, University of Žilina. He joined the Department of Telecommunications and Multimedia, University of Žilina in 2006. His recent research interests are stochastic algorithms based optimization, intelligent transportation systems, and sensors networks.

Peter Vestenický was born in 1967. He received Ing. (M.Sc.) degree in 1990 at the Department of Telecommunications and Multimedia, University of Žilina and the Dr. (Ph.D.) degree in 1994 at the same department. His Ph.D. thesis was focused on the Asynchronous Transfer Mode (ATM) and mathematical modelling of policing mechanisms for ATM networks. Since 1994 he has been working at the Department of Control and Information Systems, University of Žilina. His main research areas are radiofrequency identification, intelligent transportation systems, industrial and computer communication networks.

Vladimír Hottmar graduated from the University of Žilina in 1975 with the Master's degree in Transport and Communications. At the beginning of his career he was employed at the Research Institute of Computers Techniques in Žilina, where he worked as a researcher at the Department of Microcomputers and Development Systems for 22 years, later being responsible for running the Research Department. In 1999 he accomplished his Ph.D. studies at the University of Žilina. Currently he is employed at the University of Žilina at the Department of Telecommunications and Multimedia in the position of associate professor and is also involved in some research work. He is a VEGA project leader.

Distributed and anonymous way of the malware detection

Peter Kenyeres, Gabor Feher

Abstract — **In this paper we introduce a multi-domain architecture and novel algorithms for malicious (potential botnet) activity recognition based on NetFlow network traffic statistics. Scalability and robustness were the main principles during the design of this architecture. We demonstrate a new method is able to recognize botnet participant computers (zombies), while the algorithms provide utmost anonymity to network operators. Furthermore, we also provide an aggregation scheme to significantly reduce the number of NetFlow records. This is important to handle the current high-speed networks efficiently.**

Index Terms — **anonymous, botnet, distributed malware detection, netflow**

I. INTRODUCTION

In recent years, the Internet and the services are built upon it turned into part of our everyday life. Thus, we have to face the drawbacks of the technology more and more frequently. The criminals quickly recognized the exceptional possibilities lie in the new medium. In some cases, such as spam or phishing a whole business model was built around these activities. Nowadays, numerous and different abuses may threaten the confiding and/or careless users. And the number of users concerned by these attacks is continuously growing [1]. Generally speaking, the botnets give the technical background of the largest attacks.

Actually, botnet is vast network of compromised hosts under the control of single master who possesses the ability to launch crippling denial of service attacks (DoS), send enormous quantities of unsolicited e-mail messages (spam) and infect thousands of vulnerable systems with privacy-violating spyware or serving phishing sites, performing click fraud, etc. Besides, they also have aggressive exploit activity as they rope in new vulnerable systems to increase size of the network. The detection of above mentioned attacks are a relatively easy task, there are numerous solutions in the literature, e. g. [2] [3].

Despite these solutions the elimination or paralysis of attacks' sources raises more serious challenges. Researchers have proposed many different approaches to detect botnet behaviour in the monitored network. Gu, G. et al. show correlation based approach of the botnet detection process [9]. Livadas, C. et al. [10] integrate the recent results of the machine learning technique to detect botnet activities and [11], [12] use compression methods before the classification of the network traffic. However, one of the most common problem of the currently existing solutions, almost all of them are designed to use data from one single network only.

In this paper we introduce a novel security architecture which is reliable, efficiently scalable and can be anonymous. The architecture relies on a structured peer-to-peer (P2P) network to satisfy the scalability and the global availability requirements. Considering the huge amount of traffic data NetFlow [4] is applied to reduce the storage space required for traffic logs. Furthermore, since joined peers do not have any intention of revealing their traffic properties, so data anony-mization is a key issue in the system. However, network ad-ministrators will be able to recognize new threats and they can react to the infections more efficiently by contribution of our work.

In order to measuring the risk of the botnet threat together with gathering inputs to evaluate our algorithms: during six months long test period a HoneyPot [5] was run over an unused IP domain. Table 1 summarizes the results. Roughly speaking, one suspicious attempt occurred in every five seconds averagely. According to this, we may relate that the botnets still mean serious threat to the world computer networks. And we may deduct the inference that more than 80 per cent of the captured botnet clients still use IRC protocol as a C&C channel. However, the considerable part of the attacks embittering our everyday life (spam and particularly DDoS) are successful, only if they are executed with many computers in near identical time from many distinct places. Therefore, extended and distributed protections are desirable, which can be reached by collecting data from different local networks.

Thus, the whole malware network's recon, disablement and elimination become quicker and easier. But it brings up the following problems: the sample recognition can be quite difficult in the networks because of the different structure and their unique traffic patterns. The users' contrariety may mean additional difficulty because the network operators do not want to reveal their managed networks' structure or communication processes.

Manuscript received January 9, 2011.

Peter Kenyeres is with the *Budapest University of Technology and Economics, Department of Telecommunication and Media Informatics, High-Speed Networks Laboratory* (e-mail: kcnycrcs@tmit.bmc.hu).

Gabor Feher is with the *Budapest University of Technology and Economics, Department of Telecommunication and Media Informatics, High-Speed Networks Laboratory* (e-mail: feher@tmit.bme.hu).

TABLE I
RESULTS OF OUR HONEYPOT

Measured value	Quantity
Attack record	3 303 194
Mean attack frequency (per day)	18 352
Logged submission	516 838
Captured infected file	10 689
Captured unique malware	108
Captured botnet clients	56
Different IP address	907
The highest attempt from single IP	31 295
Origin countries	64

Currently, this area of the botnet issue is quite open. Exactly, this is where we can fill the vacuum and can prove the necessity of a distributed architecture which provides efficiency, robustness and utmost anonymity. We put steps to organizing the defense based on the separately collected network traffic data. Furthermore, the anonymity guaranteed by our algorithms helps to win the users' confidence.

The remainder of the paper is organized as follows. In Section 2 the system model is introduced including the system architecture and the different type of nodes participating in it. In Section 3 the phases of our system and realization of the design priorities are presented. In Section 4 efficiency of algorithms are evaluated. Finally, the results are summarized in Section 5.

II. SYSTEM ARCHITECTURE

The architecture consists of four elements: agents, honeypots, data processor and distributors. This layout is depicted in Figure 1. The first three are connected via a structured P2P network which implements Distributed Hash Table (DHT). This property helps to the participants join into system easily, hence to reach a globally available and distributed malware detection system. Further, it also improves scalability and robustness of the structure.

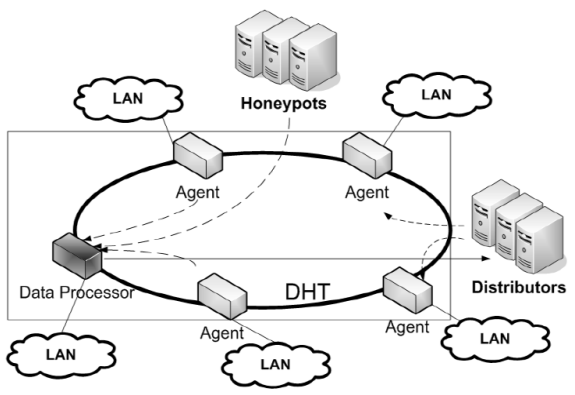


Fig. 1 The main components of the structure

The roles of the components are the following:

• Agents: These nodes are located between the border of the local area network and the Internet. They interactively dump and analyze the local network traffic to detect possible attacks, e. g. DDoS, spam, scan, etc.

• HoneyPots: These entities mark the suspicious traffic. When a new threat was detected, honeypot creates traffic traces of the malware, mark their command and control (C&C) channel and send the marked and anonymized trace to the current data processor.

• Data Processor: It is a specific agent node. Only one data processor exists in the network at same time, but this role is passed on at certain times. It collects the reports of malicious activities and the compressed and anonymized flows from agents as well as the anonymized and marked flows from honeypots. Its task is to create clusters from the data, evaluate the results. Furthermore, if malicious activities are detected it will have to send the network traces to a distributors.

• Distributors: They are responsible for sharing the collected anomalous traces with the agents. Distributors are independent from the P2P network. They accept requests from the data processors and serve the available sample updates to the agents.

III. METHODOLOGY

In this chapter, we describe the details of the aforementioned task, such as, flow aggregation and sample creation.

A. Flow Aggregation

NetFlow [4] logs are the inputs of this method. The most important fields of these are the source and destination IP addresses, source and destination port numbers and the transport protocol, since these define a session. In that case, if the same IP addresses are communicating with the same transport protocol, and at least one of them on the same port. Formally, if it is true for two flows A and B that

$$IP_{src\_}A=IP_{dest\_}B,\ IP_{dest\_}A=IP_{src\_}B,$$
$$Port_{src\_}A=Port_{dest\_}B\ OR\ Port_{dest\_}A=Port_{src\_}B$$

These flows can be put into the same group, because they represent the different direction of the same connection. Hence, we can assume that the two records belong to the same session so it is unnecessary to treat them separately. This idea gives the key for the compression.

In the first step flow regrouping can be done by putting flow records with the similar connection parameters to the same group to represent each group by a single flow. Then, outlier filtering is applied for both directions to separately handle the salient data. This filtering due to all dimensions respectively is done by computing the m mean and the σ variance of the values. If the variance is relatively high (σ/m is greater than a fixed ε_0), a new group will be created for the most outlying flow. This step is iterated until there will be no more outliers. The last step is the aggregation of the remaining flows in each group to obtain a representant. The values for packets, octets and active time will be added up, the earliest start time and the latest end time will be selected, and 5 more values will be computed: number of flows aggregated, mean packet size, mean active time, duration (the time elapsed between the

earliest start time and the latest end time), up/down+down/up (up/down stands for the sum of octets in the flows with direction up/down, but these directions can be chosen arbitrary). This 5-tuple will represent a group. The IP addresses, port numbers and the transport protocol are omitted to get a kind of anonymity.

B. Flow Processing and Sample Generation

The incoming aggregated NetFlow logs have to be classified to obtain flow samples belong to the botnet traffic we want to detect. Logs are sent by agents which are detected an attack. If this agent is a honeypot, the traffic logs will contain botnet traffic related flows (C&C channel communication and attack) without any legal background traffic. These flows are trusted in the sense that these are originated from a trusted entity and can be used as a sample of the botnet traffic. Therefore, these flows are referred as marked flows. The flows captured by honeypots that do not belong to the C&C channel can be marked differently or simply omitted. The clustering is applied to partition the data set that consists of marked and unmarked flows. Several previous works [7] [8] demonstrated that clustering of Internet traffic using flow statistics has the ability to group together flows according to the same traffic. The unmarked flows are used to improve the precision of the classifier. In this paper we applied the X-Means algorithm [6]. After clustering supervised learning and maximum likelihood estimation is applied to identify botnet traffic related cluster(s). It is not our purpose to identify all of the clusters, our aim is just to select those belong to botnet communication.

C. C&C Channel Recognition

After agents have downloaded the samples they can start the C&C channel recognition procedure. First of all, all agents have to aggregate their flows to present a similar data structure like the aggregated sample. It not just decreases the size of the data set, but offers relatively fast search and preserves anonymity as well. To select all botnet related flows from the agent's flow set the clustering method discussed in Section 3.2 can be applied. Let x be a five dimensional vector from the agents aggregated flow set (described in Section 3.1). Then the following steps are required:

1. Calculate the distances of the feature vector x from the cluster center(s) in the samples:
$$d_1 = d(x, C_1), \ldots, d_r = d(x, C_r)$$
2. Select an index i, if there exists such that $d_i < \varepsilon_i$ (Note that if such an index exists, then it will be unique)

We can assume that, if those vectors are closer to the botnet releated cluster than certain di they belong to the botnet communication. Because if this vector is added to the training data set, then after the next iteration of X-Means the vector will be an element of this cluster.

IV. EXPERIMENTAL RESULTS

We implemented our algorithms in native C and we have created a testbed network in the laboratory of the university to collect Netflow logs which contain certain malware traffic.

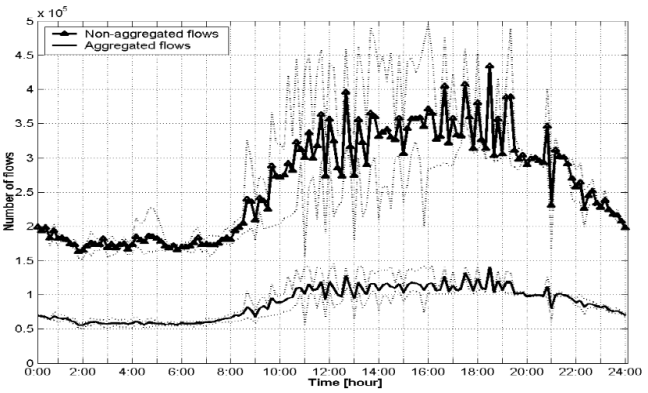


Fig. 2 Size of the original and the aggregated data sets over one day

At first, the efficiency of the aggregation scheme was analyzed separately by each campus log. Note that these traffic data came from a live network environment. And for all the 432 of 10-minute logs the compression ratio of the algorithm was between 0.3 and 0.35. Which means it reduced the size of the data set by 2/3. The average single-threaded preprocessing running time for one 10-minute log was less than 10 seconds. Figure 2 shows the size of the original and the aggregated data sets in the time period of one day.

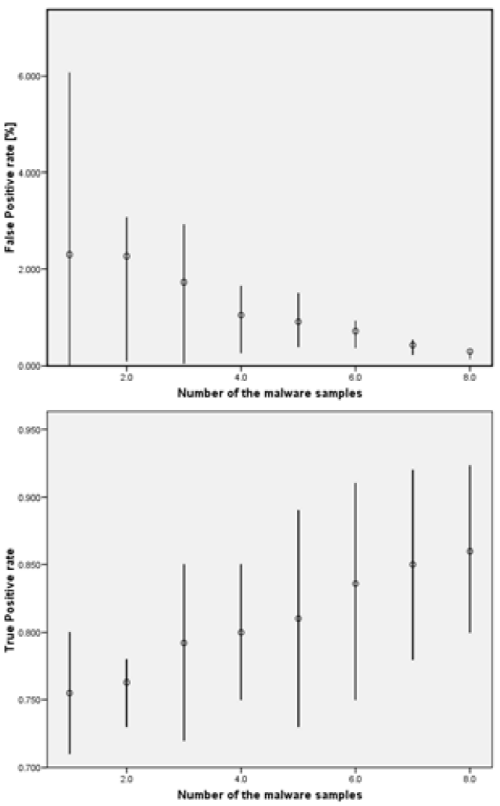


Fig. 3 and true positive rate depending on number of the samples

Further, we tested the C&C channel recognition algorithm. The data set came from campus live network and the laboratory testbed. We simulated three virtual LAN networks and in-

fected them with different botnet clients. In addition we had two more subnets: one for the victim and one for the botnet controller. Besides the legal traffic generated by the computers of the subnets, such as FTP, HTTP and e-mail, we simulated an attack against the victim directed by the botnet controller. This event triggered the sample generation process discussed in Section 3.2. Figure 3 depicts the recognition becomes more efficient (number of the false positive elements significantly decreasing while true positive ratio increasing) by increasing number of the malware samples which come from the detector agents.

V. CONCLUSION

In this paper, we have shown architecture for anonymous and distributed malware detection. After the basics of system we presented our solution proposals to provide scalability, robustness and anonymization together with generate and distribute malware sample in multi-domain environment. In addition, we proposed two algorithms: one for detection with the help of samples which was generated in different subnets and another one for the reduction of the huge amount of network statistical data. We demonstrated the strength of the algorithms: i) the detection algorithm was able to find botnet clients using the aggregated samples. ii) the aggregation method reduced the NetFlow entries to one third in practice. We note that these samples provide anonymity in that sense they do not contain any kind of valid IP information. Consequently, each and every user can be sure that their network traffic is not revealed totally. As a result, there is no need to establish mutual and unconditional trust among all participants. This property of the architecture can facilitate to make extensive use of the system.

REFERENCES

[1] Cloudmark International Spam Survey, http://www.cloudmark.com/en/survey-results/2010-08-02.
[2] V. Sekar, N. Duffield, O. Spatscheck, J. Van Der Merwe, H. Zhang, "LADS: Large-scale Automated DDoS detection System" USENIX ATC, pp. 171-184., 2006W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.
[3] A. Garg, N. Reddy, "Mitigation of DoS attacks through QoS regulation" Microprocessors and Microsystems, vol. 28, Issue 10, pp. 521-530, Elsevier, 2004
[4] Cisco Systems NetFlow Services Export Version 9, RFC 3954, http://www.ietf.org/rfc/rfc3954.txt
[5] L. Spitzner, "Honeypots - Tracking hackers" Pearson Education, 2003
[6] D. Pelleg, A. Moore, "X-means: Extending K-means with efficient estimation of the number of clusters" Int. Conf. on Machine Learning, pp. 727-734., Morgan Kaufmann, San Francisco, CA, 2000
[7] J. Erman, M. Arlitt, A. Mahanti, "Traffic Classification using Clustering Algorithms" SIGCOMM'06 MineNet Workshop, Pisa, Italy, 2006
[8] A. McGregor, M. Hall, P. Lorier, J. Brunskill, "Flow Clustering Using Machine Learning Techniques" PAM 2004, Antibes Juan-les-Pins, France, 2004
[9] G. Gu, P. Porras, V. Yegneswaran, M. Fong, W. Lee, "BotHunter: Detecting malware infection through ids-driven dialog correlation" Security'07, 2007
[10] C. Livadas, R. Walsh, D. Lapsley, W. T. Strayer, "Using machine learning techniques to identify botnet traffic" 2nd IEEE LCN WoNS'2006, Tampa, USA, 2006
[11] M. K. Reiter, T. F. Yen, "Traffic aggregation for malware detection" DIMVA'08, Paris, France, 2008
[12] S. Wehner, "Analyzing worms and network traffic using compression" Journal of Computer Security, pp. 303-320, vol. 15., IOS Press, 2007

Peter Kenyeres was born in Budapest, Hungary in 1983. He has been studying at the Budapest University of Technology and Economics since 2003. He obtained his MSc degree with major in Security of Information and Communication Systems in 2008. Currently, he is a PhD student of the Department of Telecommunication and Media Informatics in BME and a member of the High-Speed Networks Laboratory.

Gabor Feher graduated in 1998 as a computer engineer at the Budapest University of Technology and Economics. He received his PhD degree in 2004 in the field of resource control in IP networks. Currently he is an associate professor of the Department of Telecommunication and Media Informatics. He gives lectures in the topics of network and multimedia security. Since 1997 he is a member of the High Speed Network Laboratory of the Department and is involved in several national and international research projects.

Improving TCP-friendliness and Fairness for mHIP

Tatiana Polishchuk and Andrei Gurtov

Abstract—**Multihomed environments are getting increasingly common, especially for mobile users. mHIP was designed to provide secure multipath data transmission for the multihomed hosts and boost throughput of a single TCP connection by effectively distributing data over multiple available paths.**

In this paper we develop a TCP-friendly congestion control scheme for mHIP secure multipath scheduling solution. We enable two-level control over aggressiveness of the multipath flows to prevent stealing bandwidth from the traditional transport connections in the shared bottleneck. We demonstrate how to achieve a desired level of friendliness at the expense of inessential performance degradation. A series of simulations verifies that mHIP meets the criteria of TCP-compatibility, TCP-equivalence and TCP-equal share, preserving friendliness to UDP and another mHIP traffic. Additionally we show that the proposed congestion control scheme improves TCP-fairness of mHIP.

Keywords: Internet, HIP, multipath routing, TCP-friendliness, goodput

I. INTRODUCTION

Multipath data transfer is a promising technique for enhancing reliability of Internet connections. New mobile devices and laptops are equipped with several network interfaces (e.g., WLAN, GPRS, 3G) and have multiple links to the Internet, which results in availability of multiple paths between a source and destination end host.

TCP [24] comprises a major share of the total Internet traffic. Among its other management tasks, TCP controls segment size, the rate at which data is exchanged, and network traffic congestion [25]. However, traditional TCP flow is constrained to use one path only per one connection between two communicating hosts. There are efforts within the networking community to overcome this limitation. Most of these efforts rely on the mechanisms which aggressively compete for network resources. Naive designs and implementations risk substantial unfairness to well-behaved TCP flows. Proper per-flow congestion control is required to limit aggressiveness of the proposed multipath solutions.

Other multipath communication methods, proposed to efficiently utilize multiple access links, unable to take advantage of all available multipath bandwidth because they do not properly consider end-to-end delay of packet transmission. Out-of-order data arrivals at a receiver cause unpredictable underutilization of spare network capacity. Packet reordering and non-congestion packet loss can significantly degrade TCP performance.

Manuscript received January 26, 2011

T. Polishchuk is with Helsinki Institute for Information Technology HIIT, Aalto University, Finland email:firstname.lastname@hiit.fi

A. Gurtov is with Helsinki Institute for Information Technology HIIT, Aalto University, and the Centre for Wireless Communications, University of Oulu, Finland email:lastname@hiit.fi

TCP-friendliness has emerged as a measure of correctness in Internet congestion control. The notion of TCP-friendliness was introduced to restrict non-TCP flows from exceeding the bandwidth of a conforming TCP running under comparable conditions. Protocols commonly meet this requirement by using some form of AIMD (Additive Increase Multiplicative Decrease) congestion window management, or by computing a transmission rate based on equations derived from AIMD model.

In the prior work [8] we proposed a multipath solution on HIP layer [7]. Multipath HIP (mHIP) combines the advantages of HIP advanced security with the benefits of multipath routing such as better resource utilization, increased throughput and fault tolerance. HIP multihoming extension [26] supports multiaddressing in a functional layer between IP and transport and provides HIP hosts with the ability to use multiple access networks simultaneously. Simultaneous Multiaccess (SIMA) [21] utilizes multihoming for assigning separate transport connections independently to different paths. We take alternative approach by using multiple parallel paths simultaneously inside one transport connection. mHIP multipath scheduler effectively distributes incoming data over available paths on the per-packet basis, taking into account their rapidly changing parameters. mHIP is a generic multipath solution. It was designed to schedule not only the most common TCP traffic, but also data from different transport protocols, which are not necessarily TCP-friendly, e.g. UDP, SCTP, DCCP. Simple congestion control measures were suggested to provide reliable multipath data delivery.

In this paper we study TCP-friendliness of multipath HIP design with respect to coexisting connections. The contributions of this work include the development of a two-level congestion control concept for a reliable multipath data transmission and methods of tuning aggressiveness of individual flows from the multipath bundle in order to provide a desirable level of TCP-friendliness while avoiding significant performance degradation. The proposed congestion control scheme also improves TCP-fairness for mHIP, which allows to relax the original assumption that the chosen paths should necessarily be bottleneck-disjoint.

The rest of the paper is organized as follows. Section II summarizes the related work. Preliminaries are presented in Section III and contain the review of multipath HIP simple congestion control and definitions of TCP-friendliness. Section IV presents the step-by-step work which was done to enable TCP-friendly congestion control for mHIP. We verify the correctness of the proposed congestion control scheme in Section V. Conclusions and future work are given in Section VI.

II. RELATED WORK

Despite the fact that multiple multipath solutions for multihomed hosts has recently emerged, multipath routing is not yet widely deployed in practice. Researchers study advantages of its implementation on different layers of the TCP/IP stack.

Transport layer solutions, such as SCTP [15], MPTCP [6], TCP-MH [18], can naturally obtain the most recent information on the quality of different paths and detect congestion situations in timely manner. For example, SCTP can perform measurements across several paths simultaneously, and then map flows on one or another path. However implementing multiaddressing and multipath functionality on transport layer involves significant re-factoring of the code, in particular separating connection and path-specific components so that functions such as congestion control could be implemented per each path. Transport layer multipath solutions are not easy to deploy also because they involve inevitable changes to the corresponding applications.

Network layer approaches [2], [5] are generally easy to deploy and involve only minimal changes, contrary to the application and transport layer solutions. They are transparent to the applications, but do not support proper per-flow congestion control, which is needed to provide the required level or TCP-friendliness to the external connections.

Wedge-layer multipath solutions implemented in mHIP [8], LIN6 [11], MIP6 [16] have an advantage of being able to maintain multiaddressing information across transport associations. The transport activity between two endpoints may well be able to use multiaddressing immediately and with no further administrative overhead. Edge-based locator exchange protocols can be incorporated without necessitating modification to any host's IP or transport modules, which makes them the best choice to provide generic multipath functionality for legacy Internet applications and transport protocols.

mHIP naturally solves the tasks which are challenging for any multipath design. These include providing end-to-end security for each individual multipath flows, facilitating the ability to traverse NATs and middleboxes, as well as mobility support, which are inherited from the standard HIP protocol implementation. mHIP identifies individual multipath flows with SPI (Security Parameters Index) specific for each path, which guarantees the proper packet to path assignment, contributes to the flow congestion control and helps to prevent output data sequence from reordering. As a wedge-layer solution mHIP design does not require modifications to any transport modules or applications. Legacy IPv4 and IPv6 applications unaware of multiple paths can benefit from it transparently.

There is an effort in the community to create new methods which effectively and TCP-friendly utilize a spare network capacity. In [9] authors created a parallel multipath TCP solution, which controls data transmission over coordinated multiple TCP connections. They stressed the importance of TCP-friendliness for multipath schemes and suggested a way to find a balance between effectiveness and fairness. Their work provided a motivation to design a TCP-friendly congestion control over multipath flows inside one TCP connection.

When data packets are sent over several paths inside one connection they can experience different end-to-end delays and arrive out of order. In case of TCP traffic, packet reordering causes significant performance degradation. The authors of [19] surveyed and analyzed relevant techniques on coping with multipath TCP packet reordering. They conclude that there exists no one-fits-all solution to solve the problem of packet reordering for multipath TCP. Basing on the methods [3], [4], [20], [29] we suggest the improvement for multipath HIP which reduced the level of reordering on the receiver and significantly improved TCP-friendliness of our scheme.

According to the resource pooling principle [28] when several subflows of one connection share a bottleneck, their resource consumption adds up. Multipath connections with a large number of TCP-friendly subflows can compete unfairly against a smaller number of regular TCP connections. Each subflow is as aggressive as a single TCP, and a bundle of n TCP-friendly subflows will hence use an approximately n times greater share of the bottleneck resource than they should. *TCP-fair* multipath connection should displace no more TCP traffic than a traditional TCP stream would displace. A number of methods [9], [22], [10] were proposed to study and solve the TCP-fairness problem. The congestion control solution for mHIP, which we present further in this paper, is also designed to meet the TCP-fairness criterion.

III. PRELIMINARIES

A. TCP-friendliness Definitions

TCP-friendliness is a generic term describing a scheme that aims to use no more bandwidth than TCP uses. In this paper we study mHIP congestion control in view of the criteria proposed in [27]:

A *TCP-compatible* flow, in the steady state, should use no more bandwidth than a TCP flow under comparable conditions, such as packet-loss rate and round-trip time (RTT). However, a TCP-compatible congestion control scheme is not preferred if it always offers far lower throughput than a TCP flow.

A *TCP-equivalent* scheme merely ensures the same throughput as TCP when they experience identical network conditions. Although a TCP-equivalent scheme consumes TCP-equivalent bandwidth when working by itself, it may not coexist well with TCP in the Internet.

TCP-equal share is a more realistic but more challenging criterion than TCP-equivalence and states that a flow should have the same throughput as TCP if competing with TCP for the same bottleneck. A TCP-equivalent flow may not be TCP-equal share, but the opposite is always true.

To be able to meet all three criteria a TCP-friendly scheme should use the same bandwidth as TCP in a steady-state region, while being aggressive enough to capture the available bandwidth and being responsive enough to protect itself from congestion, as the packet-loss condition changes in the paths in the transient state. *Aggressiveness* of a scheme describes how the scheme increases the throughput of a flow before encountering the next packet loss, while *responsiveness* describes how the scheme decreases the throughput of a flow when the packet-loss condition becomes severe.

In what follows we will examine the ability of our multipath solution to adhere to the proposed definitions of TCP-friendliness. To evaluate its performance we intruduce the factor of friendliness metric:

$FF(flow) = \frac{T(flow)}{T(TCP)}$

Here $T(\cdot)$ denotes the average flow throughput in Mbps. $FF = 1$ indicates the solution satisfies the strongest TCP-equal share criterion, while solution resulting in $FF > 1$ is more aggressive than a typical TCP and the one with $FF < 1$ may be not TCP-compatible.

B. Review of Multipath HIP with Simple Congestion Control

In the prior research [8] HIP multipath scheduling showed a potential to aggregate about 99% of the sum of individual paths bandwidth. Simple congestion detection and avoidance are able to prevent the sending rate of the multipath traffic from significant degradation caused by congestion in the paths. Before we start evaluating mHIP congestion control scheme in the view of TCP-friendliness criteria, we recall how it operates.

1) *Connection establishment*
During the base exchange HIP obtains information about the number of available interfaces on both communicating hosts and the number of available paths with the initial parameters such as available bandwidth and propagation delay.

2) *Updating parameters of the paths*
mHIP uses HIP signaling packets for path probing. The frequency of heartbeats can vary depending on the particular setup.

3) *Sending data*
HIP multipath scheduler optimally splits data among the paths according to their capacities. The details of scheduling algorithm are provided in [8].
mHIP stores packet-to-path assignments at the sender and also in the ESP packet headers, which are used according the HIP standard [13]. SPI number, specified in the packet header corresponds to the path which is assigned to deliver this particular packet.

4) *Congestion control*
Marking and multipath congestion avoidance techniques provide a simple congestion control for mHIP. One packet per round-trip time is marked on the departure to each path. The expected delivery time of the marked packet is stored at the sender and then compared to its actual arrival time value on the receipt of the corresponding ACK. If the estimated delivery time and the actual arrival time of the marked packet are noticeably different, the scheduler considers the path to be congested.
Multipath congestion avoidance technique specifies two indicators of the path congestion:
   - Case 1: standard TCP *dupack action*, when the sender is retransmitting the packet after the receipt of three duplicate acknowledgments from the receiver;
   - Case 2: observed delivery time of the marked packet exceeds its corresponding expected delivery time by more than some preset value.

   If any of the two indicators suggest congestion, the path is temporarily closed and the packets are redirected to the other available paths. mHIP sends regular probes to the congested path to detect when the path becomes again free for reliable data transmission.

5) *Assumptions and limitations*
Our approach corresponds to the class of disjoint multipath routing [23]. The paths are restricted to have independent bottlenecks. The scheduler resides at the sender side, no information from the receiver is available other than TCP acknowledgments (ACKs) received by the sender. At least one available path should not be congested at any given point of time.

IV. Improving mHIP Step by Step

Next we examine mHIP congestion control in the view of TCP-friendliness criteria. We analyze the reasons why multipath flows not always fairly share available bandwidth with TCP and propose the methods to improve TCP-friendliness of our multipath solution.

A. Experimental Evaluation of mHIP with Simple Congestion Control

All simulations presented in this work were run using ns-2 network simulator [1]. A new protocol agent was implemented on the basis of TCP New Reno to deal with the multipath flow controlled by HIP. Existing TCP and UDP modules were also used to simulate external cross-traffic competing with HIP multipath flows for bottleneck bandwidth.

Consider a simulation model shown in Figure 1. A TCP traffic flow, controlled by multipath HIP, is sent from $n0$ to $n1$ over two available paths: $Path1 = n0 - n2 - n1$ and $Path2 = n0 - n3 - n1$ with the bandwidth of 8Mbps and 4Mbps respectively. Since multipath scheduler is distributing the traffic according to bandwidth-delay product of the paths, for simplicity the propagation delay is fixed to be the same for all the links and equals 30 ms. mHIP is calculating the end-to-end propagation delays in the paths, they can consist of any number of connected links and intermediate nodes. Node $n4$ is used for the path $n2 - n1 - n4$ construction, which accommodates a standard TCP New Reno flow, competing against one flow from the mHIP bundle for the bottleneck

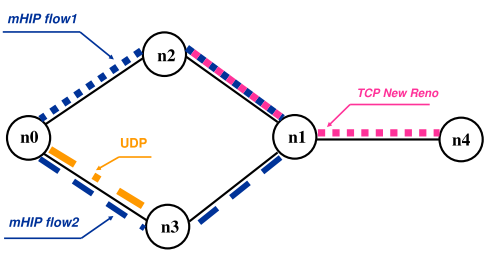


Fig. 1. 2-path simulation model.

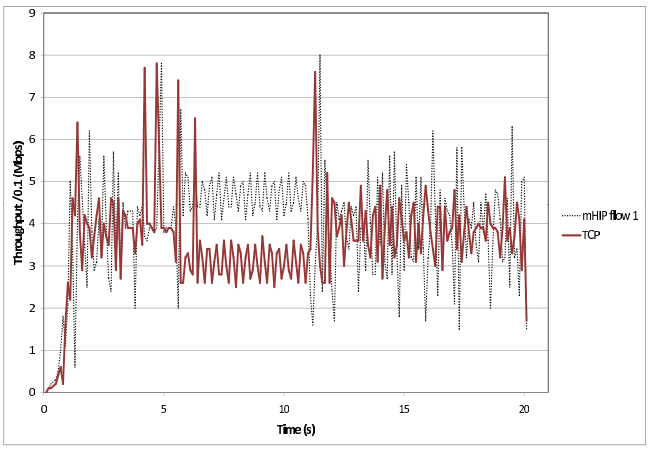


Fig. 2. mHIP flow competes with TCP New Reno flow in $Path1$.

link $n2 - n1$. Drop-Tail scenario was used to manage the bottleneck link, its size is 1.5 times the bandwidth-delay product of the link. The packet size in each flow is 1250 bytes. The simulation runs for 20 seconds, which we believe is sufficient to reflect the difference between the proposed congestion control solutions. Appropriate $Rwin$ values were used at the receivers to allow maximum throughput of the flows.

We begin our first experiment with an empty network and then allocate multipath HIP subflows to the two end-to-end paths. At the same time we start sending a TCP traffic from $n2$ to $n4$, which will compete with mHIP flow in the bottleneck link $n2-n1$. To simulate variable network conditions we also introduce cross-traffic to $Path2$. A 4Mbps UDP flow was scheduled between 5 and 11 seconds of the simulation run, triggering a congestion situation in $Path2$.

Figure 2 shows mHIP and TCP New Reno flow throughputs, averaged over 0.1 sec. As one can clearly conclude from the chart the flows do not share the bottleneck bandwidth fairly. mHIP (dotted curve) occupies more bandwidth, with the average of $T(mHIP1) = 3.98$Mbps and TCP takes just $T(TCP) = 3.56$Mbps resulting in the friendliness factor $FF = \frac{T(mHIP1)}{T(TCP)} = 1.11$.

Lets try to understand the reason why mHIP starts starving the TCP flow during the particular time period. In the beginning of the simulation run mHIP and TCP flows share the bandwidth mostly fair. At some point after 5 seconds the marking technique reports a congestion situation, resulting from the competition with UDP cross-traffic in $Path2$. Let w be the number of packets at the sender, which corresponds to the $cwnd$ value of the global TCP flow controlled by mHIP. The multipath scheduler sends $w1$ packets to $Path1$ and $w2$ packets to $Path2$ in the share correspondent to path characteristics with the total $w1 + w2 = w$. According to the congestion avoidance scheme $Path2$ is closed and all the traffic from the congested $Path2$ is rerouted to $Path1$, meaning that at this same time $Path1$ receives not only its own share $w1$ but also extra $w2$ packets. In this region mHIP is dominating and stealing bandwidth from the competing TCP transport transmission in the bottleneck link $n2 - n1$.

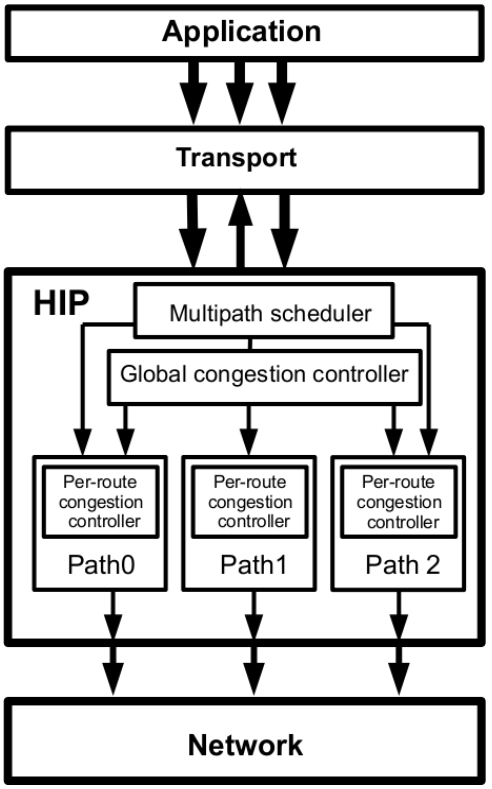


Fig. 3. Two-level multipath congestion control.

The proposed congestion control method is definitely more aggressive than AIMD policy of a typical TCP.

B. Designing TCP-friendly Congestion Control for mHIP

We want our mHIP connections to coexist with other traffic providing opportunities for all to progress satisfactory. To limit aggressiveness of the flow growth we propose the following two-level congestion control scheme - per-path AIMD plus TCP global stream congestion control on top of it, and introduce a sender-side buffer to provide better control on the packet sequence in congestion situations.

The proposed twofold congestion control scheme is illustrated in Figure 3. Global congestion controller coordinates the work of the individual per-path controllers and balances traffic load between the paths according to their available capacity. If $cwnd$ capacity of the quickest path is exceeded, the path with the next minimum estimated arrival time is chosen.

An important property of the proposed scheme is that per-path controllers are connected so that the aggregated congestion window is a simple sum of per-flow congestion windows. Same rule applies to the threshold values. Connecting per-path congestion control parameters in such a way we guarantee the resulting multipath bundle behaves as a single TCP if all are sent to the same path.

Below we summarize the proposed updates to the mHIP multipath scheduling design presented in subsection III-B. Parts 1,2 and 5 (connection establishment, path parameters updates and assumptions) remain unchanged, while there are some additions to the rest:

3) *Sending data*
After per-path congestion control limitations were introduced the scheduler takes in consideration the current sizes of per-path congestion windows. If $cwnd$ capacity of the best path is exceeded, the path with the next minimum estimated arrival time is chosen. If there is no available capacity in any of the paths, the packet is placed to the sender-side buffer until new ACK arrives.

4) *Congestion control*
Marking is now removed from the congestion control scheme. Multipath congestion avoidance retains only one congestion indication, the standard TCP *dupack* event. Upon receipt of a preset number of $dupacks$ (3 for standard TCP) the scheduler determines from which path the packet is missing and halves $cwnd$ and $ssthresh$ values of the corresponding path. This action reduces data intake in the congested path and automatically redirects traffic to the other paths which have available capacity. If there is no capacity in the paths, extra data goes to the sender-side buffer. Maximum capacity of the buffer is set to TCP receiver window size $Rwin$, making it capable to occupy the maximum flight-size number of packets in case of severe congestion situations.

It should be noted that the congestion control parameters of the global TCP flow differs from the standard TCP New Reno only in the way how the congestion control window grows and decreases (AIMD parameters): the increase of the global $cwnd$ is now dictated by the cumulative increase of the per-flow congestion windows and the reaction on losses (*dupack action*) has changed so that the global $cwnd$ is not divided by half, but only the window corresponding to the path from which the packed was lost is decreasing.

C. Experimental Evaluation of mHIP with the Updated Congestion Control

To validate correctness of the proposed congestion control scheme we repeat the experiments with the simulation scenario described in Section IV-A. Again, one of the multipath HIP flows sent to $Path1$ meets with the external TCP flow in the bottleneck link $n2 - n1$, while the other flow sent to $Path2$ is interrupted by UDP cross-traffic in the link $n0 - n3$.

The resulting throughputs of the two flows competing in $Path1$ are shown in Figure 4. mHIP average flow throughput is $T(mHIP) = 3.56$Mbps and TCP takes about $T(TCP) = 3.98$Mbps resulting in the fairness factor $FF = \frac{T(mHIP)}{T(TCP)} = 0.89$.

Here we observe the opposite extreme: mHIP flow behaves too leniently and is not able to occupy available bandwidth effectively. In the following section we analyze the problem and propose a method to solve it.

D. Balancing between Aggressiveness and Responsiveness

Competition with the external traffic naturally influences effectiveness of multipath scheduling. Mistakes in the expected delivery time estimations result in the output sequence reordering at the receiver. TCP sender receives multiple $dupacks$ in response to reordering, which mHIP scheduler treats as an indication of congestion. In response to the congestion mHIP scheduler halves congestion window of the corresponding path, reducing aggressiveness of the traffic flow. This precaution could be too strict in case when the missing sequence numbers are not lost but just slightly delayed in competition with the external flows.

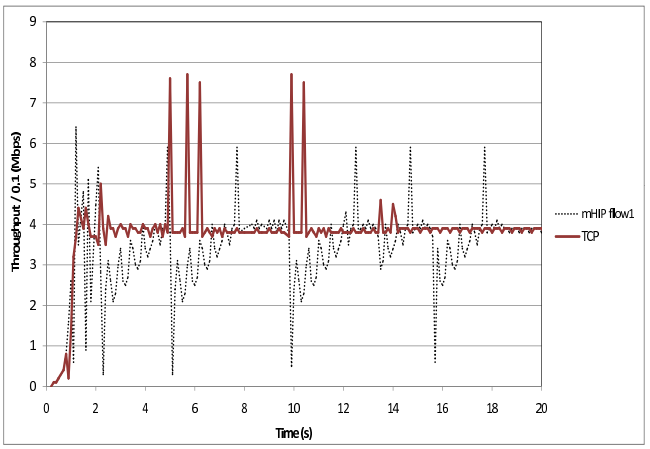


Fig. 4. mHIP flow controlled by the proposed twofold multipath congestion control is suppressed by TCP.

To differentiate between the reordering signals and actual losses we propose the following modifications to mHIP congestion control scheme. First, we increase $dupthresh$ value defining the number or $dupacks$ which serve as an indication of congestion. This method is proposed in the related work [4], [29] as a cure from the mild packet reordering. Compared with the default $dupthresh$ of three, the proposed techniques improves connection throughput by reducing the number of unnecessary retransmissions. But one should adjust $dupthresh$ value carefully since making it too large slows down the reaction of the system to the actual losses and can significantly degrade the overall performance in the networks with high loss rates.

Additionally we introduce a new time variable ADDR (allowable delay due to reordering), which keeps how much time has elapsed since the congestion situation in some path was reported. If the missing sequence number has arrived successfully during this allowable time period and the corresponding ACK arrives to the sender, $cwnd$ and $ssthresh$ of the path should be returned to the values prior to the congestion notification. ADDR is chosen to be less than the shortest RTT among the paths used to deliver multipath flow. It will assure accurate differentiation between the packets delayed due to reordering and their duplicates retransmitted after the loss was reported.

E. Controlling Friendliness with a Receiving Buffer

Another way to control aggressiveness of the multipath flow is to locate a sufficiently large buffer at the receiver and use SACK [12] together with SMART option [17]. Standard TCP with cumulative acknowledgement scheme often does not provide the sender with sufficient information to recover

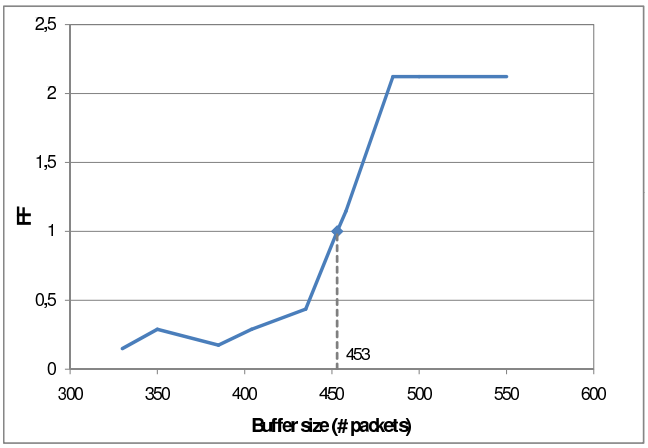


Fig. 5. Factor or friendliness depends on the size of receiving buffer.

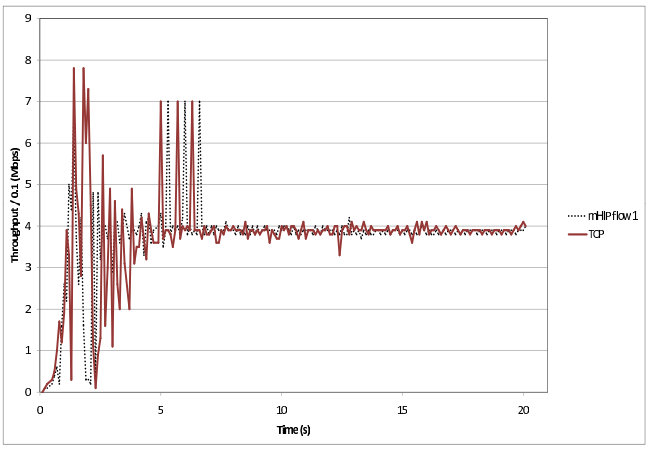


Fig. 6. mHIP flow 1 friendly coexists with TCP New Reno flow.

quickly from packet losses. When TCP with SACK option is used, each acknowledgement contains the information about up to three non-contiguous blocks of data that have been received successfully. SMART variation of the SACK provides each acknowledgement with the sequence number of the last received packet that caused the receiver to generate the acknowledgement.

We include SACK with SMART option into our multipath congestion control scheme as follows. When out-of-order packet arrives at the receiver, it is stored in the buffer and the receiver sends a cumulative acknowledgement with SACK containing the information on the sequence number of that packet. On the receipt of a duplicate acknowledgement the sender determines from which particular path the next expected packet is missing and counts the number of such events per path. After receipt of the three consecutive acknowledgements about the missing packet from the same path, the *dupack action* is invoked and the missing packet is retransmitted in the standard New Reno way. The difference from the previous multipath HIP proposal is that the sender now stores the information about the packets, retrieved from the SACK, and does not retransmit the reordered packets, which are buffered at the receiver.

The buffer stores all the packets received out of order until the holes in the sequence are filled. It will hide multipath reordering from the global TCP and improve performance of the multipath connection. When the buffer capacity is limited, the packets arriving after the buffer is full, are dropped and considered lost. It will cause regular TCP retransmissions and reductions of the global congestion window. As a result, aggressiveness of the multipath flow is limited by the choice of the buffer size.

Figure 5 illustrates how the choice of the receiver buffer size influences the resulting factor of friendliness for some fixed network settings. The developer, seeking to achieve the optimal factor of friendliness $FF = 1$, should set the buffer size to about 453 packets in this particular setup.

The dependency between the buffer size and the friendliness factor could be non-monotonic in some small intervals. The reason is that the factor of friendliness is influenced by the number of reordered packets in the multipath HIP flows. If the capacity of the receiving buffer is limited and some packets are dropped, the recovery time can vary depending on the current congestion conditions. But in the global scope by increasing the size of the buffer, one can provide multipath flow with better conditions in comparison to the regular TCP against which it competes, making it more aggressive. The buffer of maximum capacity will accommodate all the packets reordered in competition against TCP. Hence, further increase of the buffer size will not influence the factor or friendliness any more.

V. FINAL VALIDATION

Below we provide the final experimental validation of the effectiveness of our proposed modifications to mHIP congestion control. Again, we repeat the experiment described in Section IV-A with the last version of mHIP with two-level congestion control scheme and all the proposed modifications applied.

A. TCP-friendliness

Figure 6 illustrates significant improvement in TCP-friendliness of the mHIP flow when it competes against TCP for the bottleneck link bandwidth. Finally both mHIP and TCP flows are able to achieve comparable average throughputs of $T(mHIP1) = 3.80$Mbps and $T(TCP) = 3.71$Mbps with the friendliness factor $FF = \frac{T(mHIP1)}{T(TCP)} = 1.02$. The competition demonstrated high variation about the average during a short stabilization phase. This unfairness is rather moderate and can be tolerated as far as the flows quickly achieve stability and later coexist friendly.

B. UDP-friendliness

An interesting observation is that the second mHIP flow in $Path2$ behaves also about friendly competing against the UDP cross-traffic which we used to simulate variable network conditions between 5 and 11 seconds. On this interval mHIP achieves the throughput of $T(mHIP2) = 4.20$Mbps. The solid curve in Figure 7 corresponds to the UDP cross-traffic

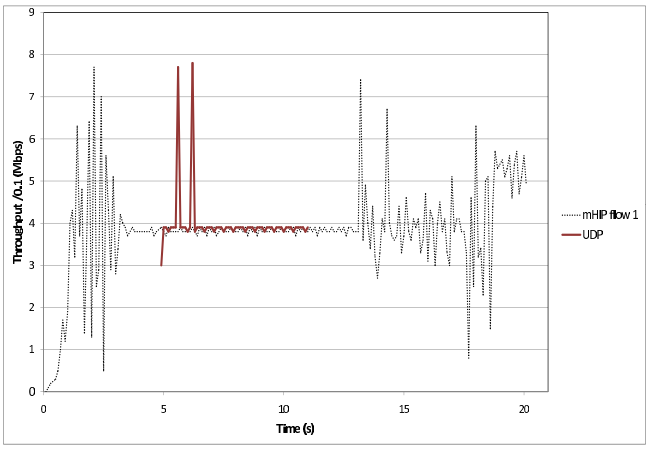


Fig. 7. mHIP flow 2 competes almost friendly with UDP cross-traffic.

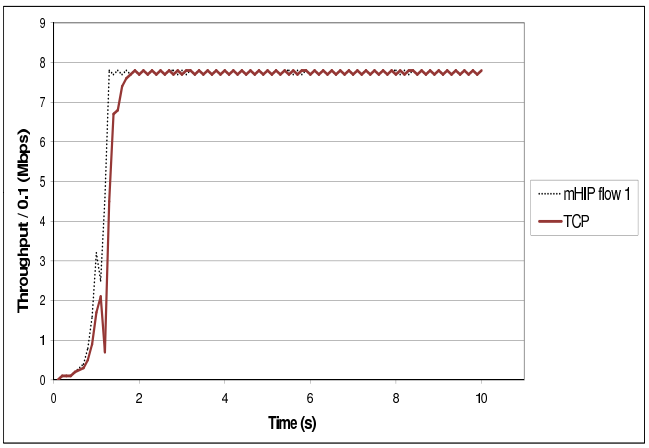


Fig. 8. Testing TCP-compatibility and equivalence of mHIP.

flow with the average flow throughput $T(UDP) = 3.98$Mbps. The flows fight during negligible time period and then find stability to share the bottleneck about fairly with a moderate unfairness of $FF = \frac{T(mHIP2)}{T(UDP)} = 1.05$.

C. TCP-compatibility and TCP-equivalence

According to the definitions TCP-compatible flow, in the steady state, should use no more bandwidth than a TCP flow under comparable conditions, while TCP-equivalent scheme ensures the same throughput as TCP when they experience identical network conditions. We send mHIP to the empty 2-path network with no cross-traffic to determine how effectively the protocol is able to use a spare network capacity in the steady state.

Figure 8 shows mHIP flow occupies no more available bandwidth than a TCP flow sent to the same path making it TCP-compatible. Moreover, mHIP achieves the same average flow throughput of 7.8Mbps as TCP in the steady state and thus meets the criteria of TCP-equivalence.

D. TCP-fairness in the Shared Bottlenecks

A flow is *TCP-fair* if its arrival rate does not exceed the rate of a conformant TCP connection in the same circumstances.

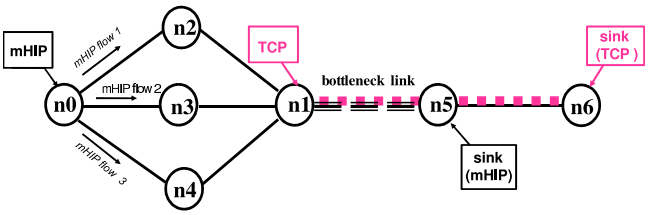


Fig. 9. Simulation model for testing TCP-fairness of mHIP.

Put another way, a TCP-fair flow sharing a bottleneck link with N other flows should receive less than or equal to $1/(N+1)$ of bandwidth available.

Previously mHIP multipath scheduling assumed the paths are bottleneck-disjoint. This automatically liberated us from the necessity to prove TCP-fairness of our solution since multiple flows of a single multipath HIP connection never shared the same bottleneck link. Now we would like to relax the original assumption about the bottleneck-disjointness of the paths. The two-level congestion control scheme introduced in this paper provides TCP-fairness of mHIP. We support the last statement with the following experimental results.

We modified the simulation model by adding one more parallel path between the nodes $n0$ and $n1$ to accommodate an additional mHIP flow and added link $n5 - n6$ as shown in Fig. 9. TCP traffic flow, controlled by multipath HIP, is now sent from $n0$ to $n5$ over three available paths: $Path1 = n0 - n2 - n1 - n5$, $Path2 = n0 - n3 - n1 - n5$ and $Path3 = n0 - n4 - n1 - n5$. $Path4 = n1 - n5 - n6$ now accommodates a standard TCP New Reno connection, which meets all the mHIP flows in the bottleneck link $n1 - n5$. To provide compatible starting conditions for their competition all four paths are set to have similar end-to-end characteristics (RTT, queue lengths and types).

Multiple experiments with various path characteristics confirmed that mHIP flows inside one TCP connection share available bandwidth mostly fairly and still friendly to the external TCP flow. The observed friendliness factor lies within the interval $[0.95, 1.03]$. A typical example of such a bandwidth distribution is shown in Figure 10. mHIP bundle behaves almost as a standard TCP when all of its flows occasionally meet in one link. This result confirms that after we improved the congestion control scheme and limited the increase of the global TCP congestion window, our mHIP solution also meets the TCP-fairness criterion.

E. The Cost of Friendliness

We achieved the desired level of TCP-friendliness for our multipath HIP solution and would like to evaluate the cost in terms of performance degradation paid for this improvement.

We calculate the total throughput TT of the traffic flow controlled by multipath HIP. In the experiment where mHIP with simple congestion control policy demonstrated an excessive unfriendliness competing against TCP NewReno, $TT(mHIP) = 6.45$Mbps. After we applied a series of modifications to mHIP congestion control, similar experiment with

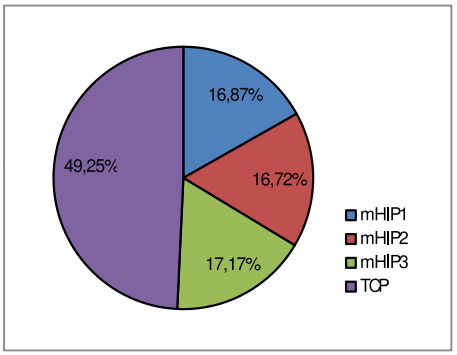


Fig. 10. Three mHIP flows from one connection compete against one TCP NewReno for the bottleneck bandwidth.

the TCP-friendly mHIP resulted in $TT(mHIP) = 5.30$Mbps, which corresponds to $\sim$18% performance reduction. A number of experiments with different network conditions confirmed the desired TCP-friendliness can be achieved at the cost of about 15-20% performance degradation.

VI. Conclusions and Future Work

We showed a way how to tune aggressiveness of the multipath data transmission controlled by mHIP without loosing its responsiveness in competition with cross-traffic. We designed a twofold congestion control scheme, and adjusted it to meet the TCP-friendliness definitions. Simulation results verify the improved congestion control algorithm meets TCP-compatibility, TCP-equivalence and TCP-equal share criteria under the proposed testing conditions, and allows mHIP to coexist friendly with TCP, UDP and mHIP connections.

The proposed congestion control scheme also assures TCP-fairness of mHIP. Having achieved the fairness of mHIP subflows in sharing the common bottleneck links we now can relax the original assumption that the paths should necessarily be bottleneck-disjoint.

The work could be extended to find a method to dynamically adjust mHIP congestion control variables and enable adaptivity to random congestion scenarios including extreme cases. We will continue examining mHIP friendliness in competition against different transport protocols and compare the results against the alternative multipath proposals.

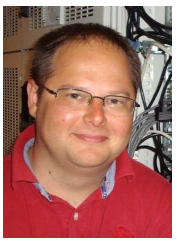
Acknowledgments

This work was supported in part by TEKES as part of the Future Internet program of the ICT cluster of the Finnish Strategic Centers for Science, Technology and Innovation.

References

[1] Network simulator ns-2. http://www.isi.edu/nsnam/ns/, last checked 15/10/2010.
[2] S. Barre and O. Bonaventure. Shim6 implementation report : Linshim6. Internet draft, draft-barre-shim6-impl-03.txt, September 2009.
[3] S. Bhandarkar and A. L. N. Reddy. TCP-DCR: Making TCP robust to non-congestion events. In N. Mitrou, K. P. Kontovasilis, G. N. Rouskas, I. Iliadis, and L. F. Merakos, editors, *NETWORKING*, volume 3042 of *Lecture Notes in Computer Science*, pages 712–724. Springer, 2004.
[4] E. Blanton and M. Allman. On making TCP more robust to packet reordering. *ACM Computer Communication Review*, 32:2002, 2002.
[5] K. Chebrolu, B. Raman, and R. R. Rao. A network layer approach to enable TCP over multiple interfaces. *Wirel. Netw.*, 11(5):637–650, 2005.
[6] A. Ford, C. Raiciu, S. Barre, and J. Iyengar. Architectural guidelines for multipath TCP development. Technical report, June 2010. Internet draft, draft-ietf-mptcp-architecture-01, work in progress.
[7] A. Gurtov. *Host Identity Protocol (HIP): Towards the Secure Mobile Internet*. Wiley and Sons, 2008.
[8] A. Gurtov and T. Polishchuk. Secure multipath transport for legacy Internet applications. In *Proc. of BROADNETS'09*, Madrid, Spain, Sept. 2009.
[9] T. J. Hacker, B. D. Noble, and B. D. Athey. Improving throughput and maintaining fairness using parallel TCP. In *IEEE InfoCom*, 2004.
[10] T. Ishida, K. Ueda, and T. Yakoh. Fairness and utilization in multipath network flow optimization. In *Proc. of 2006 IEEE International Conference on Industrial Informatics*, pages 1096–1101, 2006.
[11] M. Ishiyama, M. Kunishi, and F. Teraoka. An analysis of mobility handling in LIN6. In *Proc. of International Symposium on Wireless Personal Multimedia Communications (WPMC'01)*, Aug. 2001.
[12] V. Jacobson, and R. T. Braden. TCP extensions for long-delay paths. RFC 1072, IETF, Oct. 1988.
[13] P. Jokela, R. Moskowitz, and P. Nikander. Using the Encapsulating Security Payload (ESP) transport format with the Host Identity Protocol (HIP). RFC 5202, IETF, Mar. 2008.
[14] P. Nikander, T. Henderson, C. Vogt, and J. Arkko. End-host mobility and multihoming with the Host Identity Protocol (HIP). RFC 5206, IETF, Apr. 2008.
[15] A. Jungmaier, E. Rescorla, and M. Tuexen. Transport layer security over Stream Control Transmission Protocol. RFC 3436, IETF, Dec. 2002.
[16] J. Kempf, J. Arkko, and P. Nikander. Mobile IPv6 security. *Wirel. Pers. Commun.*, 29(3-4):389–414, 2004.
[17] S. Keshav, and S. P. Morgan. SMART Retransmission: Performance with Overload and Random Losses. In *Proc. of INFOCOM'97, Annual Joint Conference of the IEEE Computer and Communications Societies*, page 1131, 1997.
[18] K.-H. Kim and K. G. Shin. Improving TCP performance over wireless networks with collaborative multi-homed mobile hosts. In *Proc. of the 3rd Int. conf. on Mobile systems, applications, and services (MobiSys'05)*, pages 107–120, June 2005.
[19] K.-C. Leung, V. O. Li, and D. Yang. An overview of packet reordering in Transmission Control Protocol (tcp): Problems, solutions, and challenges. *IEEE Transactions on Parallel and Distributed Systems*, 18:522–535, 2007.
[20] R. Ludwig and R. H. Katz. The Eifel algorithm: making TCP robust against spurious retransmissions. *SIGCOMM Comput. Commun. Rev.*, 30(1):30–36, 2000.
[21] S. Pierrel, P. Jokela, and J. M. Melen. Simultaneous Multi-Access extension to the Host Identity Protocol. IETF Internet-draft: draft-pierrel-hip-sima-00, June 2006.
[22] C. Raiciu. Coupled multipath-aware congestion control, March 2010. Work in progress.
[23] S. Ramasubramanian, H. Krishnamoorthy, and M. Krunz. Disjoint multipath routing using colored trees. *Comput. Netw.*, 51(8):2163–2180, 2007.
[24] W. R. Stevens. *TCP/IP illustrated (vol. 3): TCP for transactions, HTTP, NNTP, and the Unix domain protocols*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1996.
[25] W. R. Stevens. TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms. RFC 2001, IETF, Jan. 1997.
[26] P. Nikander, T. Henderson, C. Vogt, and J. Arkko. End-host mobility and multihoming with the Host Identity Protocol (HIP). RFC 5206, IETF, Apr. 2008.
[27] S.-C. Tsao and N. Chiao. Taxonomy and evaluation of TCP-friendly congestion-control schemes on fairness, aggressiveness, and responsiveness. *Network, IEEE*, 21(6):6–15, Nov. 2007.
[28] D. Wischik, M. Handley, and M. B. Braun. The resource pooling principle. *SIGCOMM Comput. Commun. Rev.*, 38(5):47–52, 2008.
[29] M. Zhang, B. Karp, S. Floyd, and L. Peterson. RR-TCP: A reordering-robust TCP with DSACK. In *Proc. of IEEE ICNP*, pages 95–106, 2003.

Tatiana Polishchuk received her M.Sc in Computer Science from Moscow University of Electronic Technologies (1998) and M.Sc in Applied Mathematics and Statistics from State University of New York at Stony Brook(2007). Tatiana is a Ph.D. candidate in Computer Science at Helsinki University and works as a researcher within the Networking Research Group at the Helsinki Institute for Information Technology since August 2008. Her research area includes developing multipath and multicast solutions for Internet protocols.



Andrei Gurtov received his M.Sc (2000) and Ph.D. (2004) degrees in Computer Science from the University of Helsinki, Finland. He was appointed a Professor at University of Oulu in the area of Wireless Internet in December 2009. He is also a Principal Scientist (on leave currently) leading the Networking Research group at the Helsinki Institute for Information Technology focusing on the Host Identity Protocol and next generation Internet architecture. He is co-chairing the IRTF research group on HIP and teaches as an adjunct professor at the Aalto University and University of Helsinki. Previously, his research focused on the performance of transport protocols in heterogeneous wireless networks. In 2000-2004, he served as a senior researcher at Sonera Finland contributing to performance optimization of GPRS/UMTS networks, intersystem mobility, and IETF standardization. In 2003, he spent six months as a visiting researcher in the International Computer Science Institute at Berkeley working with Dr.Sally Floyd on simulation models of transport protocols in wireless networks. In 2004, he was a consultant at the Ericsson NomadicLab. Prof. Gurtov is a co-author of over 100 publications including a book, research papers, patents, and IETF RFCs. His articles received more than 1000 citations according to Google Scholar. He is a senior member of IEEE.

CALLS FOR PAPERS

Call for Papers

Prospective authors are invited to submit original research papers for publication in the upcoming issues of our Infocommunications Journal.

Topics of interests include the following areas:

- Data and network security
- Digital broadcasting
- Infocommunication services
- Internet technologies and applications
- Media informatics
- Multimedia systems
- Optical communications
- Society-related issues
- Space communications
- Telecommunication software
- Telecom. economy and regulation
- Testbeds and research infrastructures
- Wireless and mobile communications

Theoretical and experimentation research results achieved within the framework of European ICT projects are particularly welcome. From time to time we publish special issues and feature topics so please follow the announcements. Propo-sals for new special issues and feature topics are welcome.

Our journal is currently published quarterly and the editors try to keep the review and decision process as short as possible to ensure a timely publication of the paper, if accepted. As for manuscript preparation and submission, please follow the guidelines published on our website

http://www.infocommunications.hu/for_our_authors.

Authors are requested to send their manuscripts via electronic mail (preferably) or on a CD by regular mail to the Editor-in-Chief:

Csaba A. Szabo
Dept. of Telecommunications,
Budapest University of Technology and Economics
2 Magyar Tudosok krt. Budapest 1117 Hungary
E-mail: szabo@hit.bme.hu

Call for Proposals of Special Issues

Infocommunications Journal welcomes proposals for Special Issues – collections of papers dedicated to a particular topic of interest for the readers of the journal.

A Special Issue can be based on a recent high quality workshop or conference or papers can be collected from open call. Invited papers can be part of the special issue as well.

A Special Issue can fill in a whole issue, in which case the number of papers is expected to be 8 to 10, or it can be a Mini – Special Issue. In the latter case, at least 3, preferably 4 papers are required.

Proposals for special issues should include:

– contact information (name, e-mail, title, affiliation and address);
– resume(s) of the proposer(s), with a representative list of recent publications and related experience (Editorial Board memberships, Guest Editorships, or roles in relevant conferences' program committees);
– the proposed title for the special issue;
– the way the special issue will be compiled (contributions solicited from a technical event, or to be collected from call for this special issue;
– intent to include invited papers should be also indicated, if possible with the names of professionals who are planned to be invited;
– scope and motivation and description of the special issue;
– guest editors (if different from the proposers) with detailed contact information and resumes.

Proposals should be sent to the Editor-in-Chief:

Csaba A. Szabo
Dept. of Telecommunications,
Budapest University of Technology and Economics
2 Magyar Tudosok krt. Budapest 1117 Hungary
E-mail: szabo@hit.bme.hu

Peer-to-Peer VoD: Streaming or Progressive Downloading?

Attila Kőrösi and Balázs Székely

***Abstract*—In this paper we compare the necessary server load of a generic Peer-to-Peer supported Video-on-Demand service in peak hours for two server delivery methods, *streaming* and *elastic progressive downloading*. We focus on the playback fault free scenarios. A simple user model is used for the VCR (Video Cassette Recording) actions behavior and for the behavior as an uploader peer. Based on analytical results we identify those settings in which streaming works better. This gives the opportunity for the operator to choose between the delivery methods.**

***Index Terms*—Peer-to-Peer, Video-on-Demand, VCR, Progressive downloading, Streaming**

I. INTRODUCTION

Recently, wide range of video services have been introduced generating substantional network traffic. A potential killer video service is True Video-on-Demand (TVoD) endowed with VCR (Video Cassette Recording) options.[1] While, VoD services gain foothold one can observe improving performance close to the subscribers such as increasing bandwidth and storage capacity. These improving capacities are exploited by Peer-to-Peer supported VoD services, by existing solutions [1]–[3] and by models investigated in the literature e.g. [4]–[7].

There are two possibilities for delivering TVoD content from the server to the users: streaming and progressive downloading. In P2P-VoD system with streaming solutions the clients download the video content with a speed close to the bitrate of the video. This speed is the sum of the download speed from the server and from other clients. With progressive downloading the clients download the video with the highest speed possible between the server and the receiver. The speed achieved by progressive download is typically determined by one of these bottlenecks: the sending capacity of the server and the last mile link.

In this paper, we are investigating the circumstances that make streaming or progressive downloading more bandwidth efficient for transmitting the videos if Peer-to-peer like uploads support the system. Note that this problem has been investigates without Peer-to-Peer support in [8]. As far as we know, exect this paper and [8] there is no such a comparision available in the literature. If VCR actions are not allowed and there is a significant peak in the daily popularity, then progressive downloading is the better choice because the downloads of the clients that started the playback before the peak finish earlier because of the higher download speed before the peak. However, in this paper we compare the two delivery methods from the bandwidth consumption point of view if VCR actions are allowed. In Section II assumptions of the model are discussed and the analytical formulas are derived helping to compare the delivery methods. In Section III we present the practical consequences of the analysis.

The work has been partially supported by HSNLab, Budapest University of Technology and Economics, http://www.hsnlab.hu. The research of Székely was partially supported by NKTH-OTKA Foundation #77778. This work of Kőrösi is connected to the scientific program of the "Development of quality-oriented and cooperative R+D+I strategy and functional model at BME" project, this project is supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002).

A. Kőrösi is with the Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, e-mail: korosi@tmit.bme.hu.

B. Székely is with the Institute of Mathematics Budapest University of Technology and Economics, e-mail: szbalazs@math.bme.hu

[1]In True VoD services the users experiment very short time between the video request and the starting of the video. Common VCR options are pause, rewind or fast forward.

II. SYSTEM MODEL AND THE REQUIRED SERVER CAPACITIES

A. Assumptions

We suppose that the VoD service is a high quality service, that is, failure in the play-back is not allowed.

Let us consider a video with length L from the video library of the operator. We assume that on a given day N subscribers watch this video according to the daily starting intensity function $I(t)$. We suppose that $I(t)$ is periodic with a period of 24 hours, differentiable, and for any t $\int_t^{t+24} I(t)\,\mathrm{d}t = 1$. Starting intensity means that in the time interval $[a,b]$ $N \cdot \int_a^b I(t)\,\mathrm{d}t$ subscribers start watching the video. We also suppose that the client downloads the video until either the whole video is downloaded or the client stops the connection.

We suppose that the video is CBR coded with bitrate ρ. Note that several papers [9]–[11] have shown that VBR coded videos can also be considered as CBR coded.

We assume that the upload capacities of the clients are utilized maximally. This means that if the aggregated upload capacity of the clients at time t is $U(t)$ then the server is offloaded by $U(t)$. We also assume that the capacity $U(t)$ is shared equally among the current downloaders, that is, the bandwidth share that a downloader obtains from the uploader clients is $u(t) = U(t)/N(t)$, where $N(t)$ denotes the number of active downloaders at time t.

If the video is delivered via streaming, then the aggregated download speed (the download coming from the server and the client) of a video on the downloader downlink is considered to be ρ. Using proper sliding window algorithm [4], $\rho - u(t)$ is enough server speed for the vast majority of the download

time for achieving fault-free playback with high probability [4].

If the video is delivered using elastic progressive downloading (EPD), then *three* factors limit the download speed from the server, (i) the download limit assigned to the service on the first mile d, (ii) the upload share $u(t)$ that the downloaders obtain from other clients at time t. Consequently, the download speed from the server on the downlink can not be larger than $d-u(t)$. Further, (iii) the capacity of the central server S, so the download speed from the server of a downloader cannot be larger than $\frac{S}{N(t)}$ assuming equally distributed server capacity.

B. User model

In this subsection, we describe those user characteristics that are relevant when comparing the bandwidth consumption of the delivery methods.

We use two user characteristics that have already been introduced in [8]. In that paper the authors compare the delivery methods based on the user behavior with respect to the VCR actions if client uploads do not support the system. Two parameters that describe the user behavior during watching videos are the average viewing time (l) and the average ratio of the time spent by playing new parts (b). l is the average time elapsed between the start action and the stop action called. b is the average ratio of the time spent by playing new parts from the video and the entire viewing time. b is called "slowing down" parameter because b is small if the users apply backward jumps or pauses. The connection between l and b is that $b \leq L/l$ or even $b \leq \min\{L/l, 1\}$.

If client uploads support the system, then we have to introduce two additional parameters for describing the behavior of the clients as uploaders. The first parameter is the average upload bandwidth capacity which is denoted by u. The second parameter is the ratio of the non-active clients (currently do not watch any video) who do not turn off their STBs [2] over all non-active clients. This parameter is denoted by α.

We note that the bandwidth share $u(t)$ that a downloader obtains from the uploaders, defined in the previous subsection, can be written as follows

$$u(t) = u\frac{N(t) + \alpha(N - N(t))}{N(t)}. \tag{1}$$

C. The minimal required server capacities

In this subsection, we will determine the minimal required server capacities in the case of both delivery methods. First of all, we summarize the notation:

- ρ and L: the bitrate and the length of the video,
- N: number of subscribers,
- d and u: users' download and upload bandwidth capacities on the last mile assigned to the video service,
- α: the rate of those clients who share its upload capacity but are not active in watching video among the clients who are not active,
- $I(t)$: the daily *starting* intensity function,
- l: average viewing time,
- b: average ratio of the time spent by playing new parts,,
- $N(t)$: the number of active users at time t,
- $u(t)$: the average upload capacity of one client at time t,
- S: the capacity of the central server in the elastic case,
- $a \wedge b$: the minimum of the real numbers a and b.

Concerning the streaming case, let $A_s(t)$ denote the server load at time t. The minimal required server capacity in the streaming case is $S_s = \max_t A_s(t)$. For $A_s(t)$ we have the following simple expression:

$$A_s(t) = (\rho - u(t))b \cdot N \int_{t-l}^{t} I(\tau)\,\mathrm{d}\tau.$$

The integral along with multiplier N expresses the average number of downloads that started in interval $[t-l, t]$ (note that $I(t)$ is starting intensity). In streaming case the server uploads for those clients at time t who initiated the download at most l long time earlier. The first term $(\rho - u(t))b$ expresses the average speed of one video download from the server per user. The multiplier b decreases the required download speed since, if $b < 1$, then there are clients that pause the video or jump back and watch an already downloaded part.

The daily server load, that is, the number of bits downloaded from the server in a day can be written as

$$D_s = \int_t A_s(t)\,\mathrm{d}t.$$

We determine the minimal required server capacity S_{epd} in the elastic progressive download case. We have assumed that the server capacity S for arbitrary S is distributed equally among the clients hence the server speed per user is $S/N(t)$. The following system of equation has to be solved for obtaining $N(t)$:

$$\int_{t-y(t)}^{t} \left(\frac{S}{N(u)} + u(t)\right) \wedge d\,\mathrm{d}u = L\varrho, \tag{2}$$

$$N \cdot \int_{t-y(t)\wedge l}^{t} I(s)\,\mathrm{d}s = N(t). \tag{3}$$

In (2), $y(t)$ denotes the length of time in which the whole L long video can be downloaded by time t. From the point of the downloaded data the connection between the functions $N(t)$ and $y(t)$ is described in (2). Equation (3) expresses that the active clients at time t are those who started their download after $t-y(t)$ and are still watching the video (see the definition of l).

The system of equations (2) (3) can not be solved explicitly for $N(t)$. However, approximate solution with given accuracy can be reached, which converges to the exact solution.

Since the service does not allow play-back fault $S/N(t) \geq \rho - u(t)$ has to hold for each user. Therefore, the required minimal server capacity can be written as follows:

$$S_{epd} = \min\left\{S : \forall t,\, S/N(t) + u(t) \geq \rho\right\}.$$

Thus, the server load at time t with server having the minimally required capacity S_{epd} can be calculated as follows:

$$A_{epd}(t) = \max\left\{0, (d - u(t)) \wedge \frac{S_{epd}}{N(t)}\right\}.$$

[2] The videos shared in the system are stored in clients' equipments that can be a PC or a Set-Top-Box (STB).

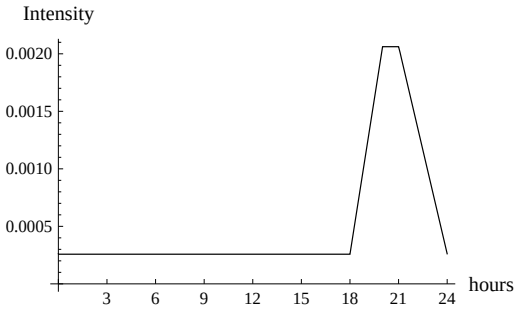


Figure 1. Starting intensity function. In this special case $I(t)$ is constant except in the interval $[18, 24]$ where it is linear and the peak is eight times higher than the constant part.

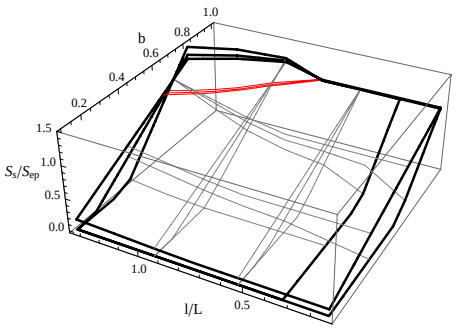


Figure 2. The capacity ratio $S_s/S_{ep}(u/, l/L, b)$ is depicted as a function of the first (l/L) and the second user parameter (b) for fixed upload capacities $u = 0$, $u = 0.5$, $u = 1$

Further, similarly as in the streaming case the number of bits that are downloaded daily from the server D_{epd} can be written as follows

$$D_{epd} = \int_t A_{epd}(t)\,\mathrm{d}t.$$

III. Practical consequences

In this section, we make several observation of practical relevance, that is, we investigate which delivery method performs better for certain parameter set-up.

If it is not mentioned in the investigated scenarios below, then the following parameter set up will be used: $\rho = 3$ Mbps, $d = 2\rho$, $b = 0.9$; $L = 100$ minutes, $l = 0.8L$; $\alpha = 0.2$. We fix a starting intensity function $I(t)$ given in Figure 1 and we will use it to derive the following results. It is a typical intensity function in a local service provider network, similar to the one found in [12].

Observation 1: (On the effect of the viewing time parameter (l/L) and the slowing down parameter (b))
S_s/S_{epd} is a growing function of both the viewing time parameter (l/L) and the slowing down parameter (b), see Figure 2. Without any client upload support the same conclusion was drawn, see [8]. It can be seen in both cases (with and without client support) that streaming is the better choice if b and l/L is close to 1, that is, the majority of the users watch the entire video without any interaction. Practically, this usually holds for popular videos [13].

In Figure 2 it can be observed that the effect of the size of the client upload ($u = 0$, $u = 0.5$, $u = 1$ in the figure) is significant but the region in which streaming is better (the upper corner of the surfaces, above the thick curves) does not change significantly if the upload capacity increases.

Observation 2: (Finer study of S_{epd} as a function of l/L)
If the viewing time parameter (l/L) increases then the required server capacity (S_{epd}) increases until it takes its maximum, then S_{epd} decreases. See Figure 3 for the graphs of this function under several fixed downlink capacities. This phenomenon is reasonable since if the video has not been fully downloaded, then the minimal server capacity in the EPD case (S_{epd}) is a growing function of l/L until some point M_d where the maximum is taken. (We do not have explicit formula for M_d.) On the right hand side of M_d S_{epd} decrerases. This case, $l/L > M_d$ corresponds to that there are several clients having

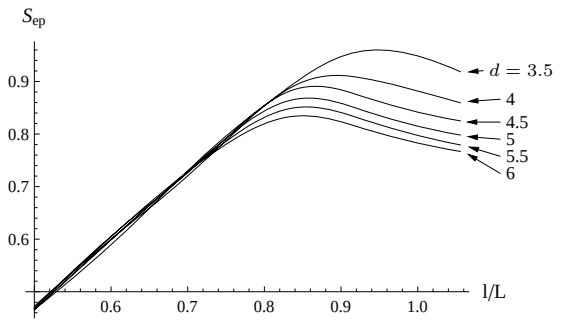


Figure 3. The server capacity S_{epd} is depicted as a function of the viewing time parameter (l/L) for fixed downlink capacity $d = 3.5$, $d = 4$, $d = 4.5$, $d = 5$, $d = 5.5$, $d = 6$

finished their downloads but they still in the system hence they contribute their upload capacities offloading the server. Before M_d the offloading property of the client is not stronger than their download bandwidth demand.

M_d is a decreasing function of d since the clients can download the video earlier with larger d and hence become "server offloader" sooner.

In this way, we have showed that if the operator choose EPD, then by increasing the downlink capacity the server capacity can be decreased.

Observation 3: (On the green-conscious subscribers) If the number of *ON* STBs (α) increases, the minimal required server capacity decreases for both delivery methods. Thus, the more user leave their STBs on the lower capacities are necessary. Indeed, the effect of the change α on $u(t)$ is similar to that of u since in equation (1) $u(t)$ is a linear function of u and α.

Observation 4: (*On the amount of downloaded data*) The consequence of Observation 1 is the same in qualitative, if we investigate the ratio D_s/D_{ep} of the amount of downloaded data from the server, see Figure 4. That is, D_s/D_{ep} is increasing in both b and l/L. The region where EPD is better is not sensitive to the size of the client upload capacity. Further, the region where EPD is better is smaller than in the case when the bandwidths were compared in Observation 1.

Observation 5: (*On almost constant starting intensity*) If the starting intensity function I of the video is constant then $N(t)$ and $y(t)$ are constant in Eq. (2) and (3). Computing

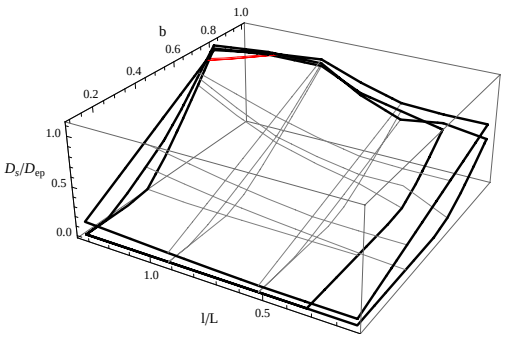


Figure 4. The capacity ratio $D_s/D_{ep}(u/,l/L,b)$ is depicted as a function of the first (l/L) and the second user parameter b for fixed upload $u = 0$, $u = 0.5$, $u = 1$

S_s/S_{epd} in this case, we have

$$S_s/S_{epd} = b{\cdot}\max\{1, l/L\} \leq \min\{L/l, 1\}{\cdot}\max\{1, l/L\} = 1.$$

This means that streaming is always the better choice independently of the other parameters including the upload capacity of the clients. This property has already been observed in [8] without any client upload.

IV. SUMMARY

The main conclusion of the paper is that comparing the systems with and without supporting client upload the results, which delivery method performs better, are the same in qualitative but different in quantitative. This is reason why the main property of a VoD system without supporting client upload remains valid: If the content is not educational and there is a significant peak in the daily popularity, then progressive downloading is a better choice than streaming. Otherwise, streaming performs better. Further, many in-depth observations have been made such as if the upload speeds of the clients increase, the provider should choose streaming instead of EPD, or if the operator choose EPD, then by increasing the downlink capacity the server capacity can be decreased.

REFERENCES

[1] "GridCast, http://www.gridcast.com."
[2] "PPLive, http://www.pplive.com."
[3] "UUSee, http://www.uusee.com."
[4] A. Kőrösi, B. Székely, and C. Lukovszki, "High quality P2P-Video-on-Demand with download bandwidth limitation," in *In Proc. of the 17th IEEE International Workshop on Quality of Service*, 2009.
[5] K. Suh, C. Diot, J. Kurose, L. Massoulie, C. Neumann, D. Towsley, and M. Varvello, "Push-to-Peer Video-on-Demand System: Design and Evaluation," *Selected Areas in Communications, IEEE Journal on*, vol. 25, no. 9, pp. 1706–1716, December 2007.
[6] C. Dana, D. Li, D. Harrison, and C.-N. Chuah, "BASSS: BitTorrent Assisted Streaming System for Video-on-Demand," in *Multimedia Signal Processing, 2005 IEEE 7th Workshop on*, 2005, pp. 1–4. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4014007
[7] C.-S. Lin and Y.-C. Cheng, "P2MCMD: A scalable approach to VoD service over peer-to-peer networks," *J. Parallel Distrib. Comput.*, vol. 67, no. 8, pp. 903–921, 2007.
[8] A. Kőrösi, B. Székely, and A. Császár, "TrueVoD: Streaming or Progressive Downloading?" *Communication Letters*, to appear, 2010.
[9] J. D. Salehi, Z.-L. Zhang, J. F. Kurose, and D. Towsley, "Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing," *IEEE/ACM Trans. Networking*, vol. 6, pp. 397–410, Aug. 1988.
[10] R. Chang, M. Chen, M. Ko, and J. Ho, "Optimizations of stored vbr video transmission on CBR channel," 1996. [Online]. Available: citeseer.ist.psu.edu/348654.html
[11] W. Feng, "Rate-constrained bandwidth smoothing for the delivery of stored video," in *IS&T/SPIE Multimedia Networking and Computing*, Feb. 1997, pp. 316–327.
[12] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, "Understanding user behavior in large-scale video-on-demand systems," *SIGOPS Oper. Syst. Rev.*, vol. 40, no. 4, pp. 333–344, 2006.
[13] B. Cheng, X. Liu, Z. Zhang, H. Jin, L. Stein, and X. Liao, "Evaluation and optimization of a peer-to-peer video-on-demand system," *Journal of Systems Architecture*, p. to appear, 2008.

Attila Kőrösi is an engineer at the Budapest University of Technology and Economics, Hungary working at the High Speed Network Laboratory. He received a M.S. degree in Mathematics from the Budapest University of Technology and Economics in 2007. His research interest is multimedia services and their stochastic models. He has published some papers on priority queuing systems.

Balázs Székely is an assistant professor at the Department of Stochastics at Budapest University of Technology and Economics, Hungary. He received his M.S. degree in Mathematics from Eötvös Loránd University, Hungary in 2000, and Ph.D. degree in Mathematics and Computer Science from Budapest University of Technology and Economics in 2005. His main research interests are queuing system analysis, random fractals, traffic and network modeling.

Mojette Transform Software – Hardware Implementations and its Applications

Jan Turan[1], Peter Szoboszlai[2], and Jozsef Vasarhelyi[3]

[1]Department of Electronics and Multimedia Telecommunications, Technical University of Kosice, Park Komenského 13, 041 20 Košice, Slovak Republic, e-mail: jan.turan@tuke.sk
[2]Magyar Telekom, Hungary 1117 Budapest, Magyar tudosok korutja 9., e-mail: szoboszlai.peter@telekom.hu
[3] Department of Automation, University of Miskolc, Miskolc-Egyetemvaros, Hungary, e-mail: vajo@mazsola.iit.uni-miskolc.hu

Abstract — **In the last ten years there were intensive researches to find new methods for handling internet image processing and distributed databases. One of the many methods is the Mojette transform (MT). The MT is used mainly in image processing applications, but can be used also for distributed databases also. The MT implemented under different operating systems and for different processors presented in helps the performance analyses of the MT and also let to conclude the need for reconfigurable hardware implementation. The Mojette Transformation Tool (MTTool) is an implementation of MT and inverse MT (IMT) in .Net environment. The software development provides us with the endless possibility of different variations of the MT implementation in a shorter time frame and on lower costs. Also the testing with such a tool is much easier and it's also better for demonstration and training purposes. There is also analyzed the hardware implementation of MT and IMT based on Field Programmable Gate Arrays (FPGA) using reconfigurable platform. The paper tries to conclude the necessity for hardware implementation for real time processing. The paper outline the development work in order to create an embedded reconfigur-able hardware based on FPGA board.**

Index Terms — **Mojette Transform, MTTool, image processing, distributed databases, FPGA, embedded systems, MoTIMoT co-processor**

I. INTRODUCTION

The Mojette Transform (MT) is originated from France where J-P. Guédon named it according an old French class of white beans, which were used to teach children computing basics of arithmetic with simple addition and subtraction [1-5]. He named MT after the analogy of beans and bins. The bins contain the sum of pixel values of the respective projection line [1]. There are several different variations of MT applications nowadays which are used in different areas such as tomography [2], internet distributed data bases [3], encoding, multimedia error correction [4,5,7], or The Mojette Transform Tool (MTTool) which was created for testing purposes, by the way it can also be used for demonstrations and training purposes as well.

However the MTTool development isn't finished yet we already gained many experiences with it, and we see how it could become more helpful for further projects both in software and in hardware development in the area of MT applications. So the main purpose to build such an environment was that with the help of it we could try to compare the software implementation of the MT later on with the hardware implementations.

In the first section we introduce the mathematical background of the MT and the IMT briefly and this is followed by the description of the transforms implementation in several versions, concentrated not on the stodgy programming stuff rather on a basic description.

Next we share some of our test results which were focusing on the time consumption of the MT and IMT and after that we also introduce the differences between the first and the other versions from the Graphical User Interface point of view illustrated with some figures as well.

Until today the aim of the researches was software implementation of the MT and IMT [1-5,7]. This kind of implementation can be used only on still images, because they cannot process the data in real-time. Motion pictures, video streams and distributed data bases require run-time processing of the frames. Only specialized hardware or embedded systems with real-time operating systems can ensure this.

This paper presents one implementation method to implement the MT and IMT in Field Programmable Gate Array (FPGA) using reconfigurable platform.

II. MOJETTE AND INVERSE MOJETTE TRANSFORM

A. Mojette Transform

The main idea behind the Mojette transform- MT (similarly to the Radon transform) is to calculate a group of projections on an image block [5]. The MT [1,11,16,17] projects the original digital 2D image:

$$F = \left\{ F(i,j); i = 1,...,N; j = 1,...,M \right\} \tag{1}$$

to a set of K discrete 1D projections with:

$$M = \left\{ M_k(1); k = 1,...,K; 1 = 1,...,1_K \right\} \tag{2}$$

MT is an exact discrete Radon transform defined for a set $S = \{(p_k, q_k), k = 1,...,K\}$ specific projections angles:

$$M_K(l) = proj(p_k, q_k, b_l) = \sum_{(i,j)\in L} F(i,j)\,\delta(b_l - iq_k - jp_k) \tag{3}$$

where *proj* (p_k,q_k,b_l) defines the projection lines p_k, q_k, and $\delta(x)$ is the Dirac delta in the form:

$$\delta(x)=\begin{cases}1, if\ \_x=0\\ 0, if\ \_x=1\end{cases} \tag{4}$$

and

$$L=\{(i,j);b_l-iq_k-jp_k=0\} \tag{5}$$

is a digital bin in the direction θ_k and on the set b_l.

So the projection operator sums up all pixels values which centers are intersected by the discrete projection line *l*. The restriction of angle θ_k leads both to a different sampling and to a different number of bins in each projection (p_k, q_k). For a projection defined by θ_i, the number of bins n_i can be calculated by:

$$n_i=(N-1)|p_i|+(M-1)|q_i|+1 \tag{6}$$

The direct MT is depicted in Fig. 1 for a 4x4 pixel image. The set of three directions *S={(-1,2),(1,1),(0,-1)}* is resulting 20 bins.

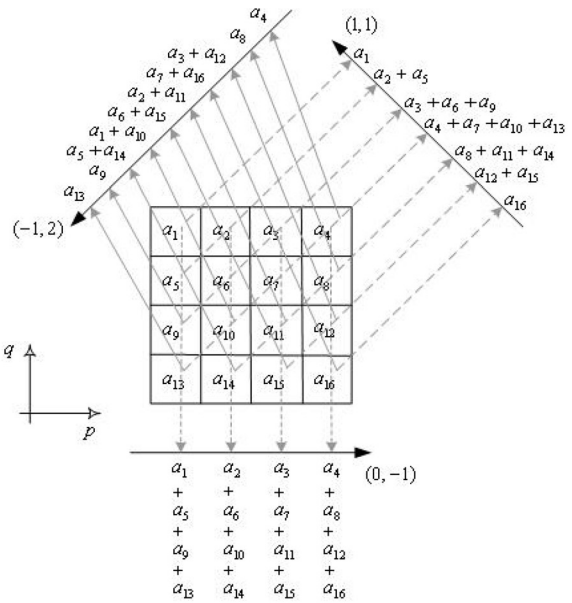


Fig.1. The set of three projections computed from a 4x4 image.

The MT can be performed by directly addition of the image pixel values in grey scale images and for bitmaps we can add the different bitmap color table values.

B. *Inverse Mojette Transform*

Computing Inverse Mojette Transform (IMT), bins are back-projected. A single pixel-*bin* correspondence must be found at each iteration cycle in order to reconstruct a pixel value. When it has been done, this pixel value is substituted in the adequate coordinate of a blank image and subtracted from the corresponding bins in each projection of the original image. To find this single pixel-*bin* correspondence examination of the projections of unary image is necessary. The *bin* values in those projections show the number of pixels which are stored in the given *bin*. Therefore the modification of the unary projections has to be done parallel (Fig.2). The IMT is iterating this process until the image is completely reconstructed. For the reconstruction both projection sets needed (one is the MT of the image and the other is the MT of a unary image where each pixel value is 1). Fig.3 shows the first step of the reconstruction process.

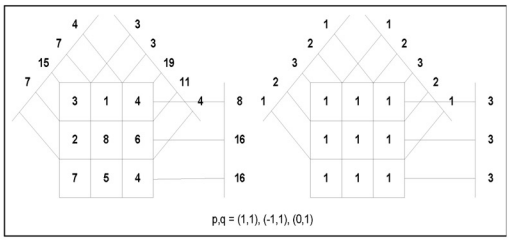


Fig.2. Mojette Transform on integers.

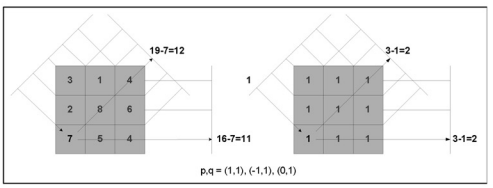


Fig.3. Inverse Mojette Transform – first step.

C. *Condition of Reconstruction*

The key of the transform with regard to the controlled redundancy is that the number of computed projections can be larger than we need for the reconstruction (inverse transform). Thus, we can control a first step of redundancy with the number of projections. The question is then to know how many projections and which projections give an adequate set to reconstruct the image (i.e. compute the Inverse Mojette Transform). The result for rectangular image was given by Katz in [6].

According to the Katz's Lemma [6] image of dimension KxL can be reconstructed with the *Mf* set of directions (p_i, q_i), i=1, 2, …, I if

$$K \le P_i=\sum_{i=1}^{I}|p_i| \quad or \quad L \le Q_i=\sum_{i=1}^{I}|q_i|, \tag{7}$$

where K and L are the dimensions of the image p_i the horizontal projection coordinate (x axis) and q_i the vertical projection coordinate (y axis).

The condition of reconstruction ensures that the number of *bins* (N_b) is greater than the number of pixels (N_p) therefore the number of equations will be larger than the number of unknown variables during the reconstruction process. At the same time the condition of reconstruction answers the question whether is any projection in the projection set which is unnecessary to reconstruct the original image.

MT for binary image of simply binary data is defined with the operations „AND" and „XOR" respectively instead of the multiplication „*" and addition „+" [1,5,11].

III. MOJETTE TRANSFORM IN MTTOOL

In MTTool the implementation of the MT was applied in three different ways [15-17]. This is due to the fact that this application is still in development and the three different ways were constructed not in the same time but in the last years.

TABLE I
MT IMPLEMENTATION AND IT'S MAIN DIFFERENCES

Nr.	Image Format	Projections	MT and IMT
1	PGM	p={1,-1,3,-3}; q={quarter of the image size}	Addition and subtraction
2	BMP	p={2,-2}; q={1} and p={3,-3,2}; q={1}	Addition and subtraction
3	BMP	p={2,-2}; q={1} and p={3,-3,2}; q={1}	Matrix

A. The First Version

In the first release the declaration of some rules, which had to be somehow flexible and at the same time they shouldn't be very complex was one of the hardest to decide. We had to declare the image sizes with which we had to work later and to look for useful relation between the picture size and the vectors we use in the MT and IMT. After calculating with several file sizes, it was clear we had to choose it so, that it should be easy to remember and have something common with all the other images as well. We decided to take the picture size 2^nx2^n where n is equal to 8 and 9, but can be changed later on easily. So the picture size is 256 x 256 and 512 x 512. In the Picture Preview we can open and display any kind of PGM or BMP file it doesn't matter which size the picture got, but some of the images are increased or decreased to fit on the screen.

TABLE II
IMAGE DISPLAY

Original size	Displayed size	Ratio
1600 x 1200	400 x 300	0,25
1599 x 1199	799 x 599	0,5
1024 x 768	512 x 384	0,5
Height < 1024	Height +180	Other

The first step in the MT after checking the restrictions is to make a vector from the pixels of the image. To define the size of this vector is easy, when we are following this simple rule (1, 2^n x 2^n). If n=8 this results the vector (1, 65536), in which every line contains a pixel value from the picture. Because the PGM picture is a 256 grayscale image, a PGM file contains just pixel values from 0 to 255. In case of a BMP image we could process it three times because of the different bitmap color table values. [13,14]

In the second step we are executing the MT. The vector p is predefined for the four projection directions and the q vector has the same value in each four case (quarter size of the 2^n x 2^n image). We generate four files for the four different projections, these are the following:

1) originalfilename.pgm.moj1 (1, q)
2) originalfilename.pgm.moj1 (-1, q)
3) originalfilename.pgm.moj1 (3, q)
4) originalfilename.pgm.moj1 (-3, q)

From the existing MT files (moj1,…, moj4) we get the original PGM picture with the IMT. All of the four MT files are needed in this case to rebuild the original image completely without any error. If any of the MT files is defect or incomplete the IMT doesn't give back the original image. Each of the four files is containing a vector described above. The next step of the IMT is, to read the first and last vectors of the third and forth MT files and put it on their place. So we have in all four corner of the picture filled up with the valid pixel values. Step 1, 2, 3 and 4 on the following figure:

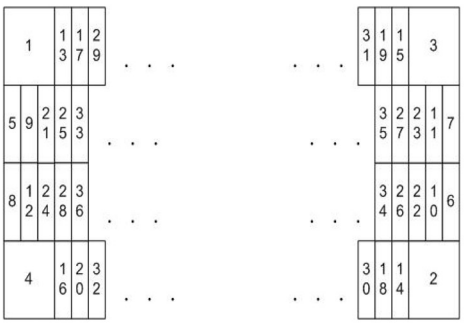


Fig.4. First 30 steps of the IMT.

After recreating the pixel values we only need to add the new header for the file and the restoration of the original image is already performed.

B. The Second and Third Version

These solutions are differing from the previous one in such a way that these are applied on BMP images and in these cases we are performing the MT and IMT on the three different bitmap color table. We use the same algorithm for the three different color maps and collecting the bins into 3 separate files which are differing only in their extensions and of course in their content. On the bitmap images we are using the directions S_1={(2,1),(-2,1)} and S_2={(3,1),(-3,1),(2,1)} for the block sizes 4 and 8. Although the MT is also prepared for the block size 16 and 32 but the implementation of the IMT isn't done yet. In the second version we use simple addition and subtraction different from the one mentioned in the first version, since here we have block sizes 4 and 8 and there we are performing the MT and IMT on the whole image at once and not step by step. In the third version instead of addition and subtraction we use matrices for the MT and IMT on the above mentioned block sizes. The MT with matrices is implemented in the following way, where bi is the bin resulted from the following equation:

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \\ b_9 \\ b_{10} \\ b_{15} \\ b_{16} \\ b_{17} \\ b_{18} \\ b_{19} \\ b_{20} \end{bmatrix} = \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0 \end{bmatrix} * \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \\ a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \\ a_{15} \\ a_{16} \end{bmatrix} = \begin{bmatrix} 10 \\ 123 \\ 37 \\ 137 \\ 254 \\ 319 \\ 433 \\ 68 \\ 6 \\ 234 \\ 125 \\ 267 \\ 312 \\ 8 \\ 45 \\ 178 \end{bmatrix} \quad (8)$$

The inverse matrix for the previous example (for the 4x4 matrix size) is implemented as it's showed on the next equation, where a_i are the original values of the matrix:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \\ a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \\ a_{15} \\ a_{16} \end{bmatrix} = \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&1&0&0&0&0&-1&0&0&0&-1 \\ 0&0&0&1&0&0&0&1&0&0&-1&0&0&0&-1&0 \\ 0&0&0&0&0&0&-1&0&0&0&0&1&0&0&0&1 \\ 0&0&0&0&0&0&0&-1&0&0&1&0&0&0&1&0 \\ 0&0&0&0&1&0&0&0&1&0&0&0&0&-1&0&0 \\ 0&0&0&0&0&1&0&0&0&1&0&0&-1&0&0&0 \\ 0&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&-1&0&0&1&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&-1 \\ 0&0&0&0&0&0&0&1&0&0&0&0&0&-1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0 \end{bmatrix} * \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \\ b_9 \\ b_{10} \\ b_{15} \\ b_{16} \\ b_{17} \\ b_{18} \\ b_{19} \\ b_{20} \end{bmatrix} = \begin{bmatrix} 10 \\ 123 \\ 25 \\ 35 \\ 12 \\ 102 \\ 252 \\ 241 \\ 2 \\ 78 \\ 255 \\ 23 \\ 178 \\ 45 \\ 6 \\ 234 \end{bmatrix} \quad (9)$$

IV. EXPERIMENTS WITH MTTOOL

With the built in ZIP and Huffman coding opportunities (Fig. 5) we can decrease the size of any vectors which are created from the projections of MT [13-17]. The Huffman lossless encoding and decoding algorithm was chosen due to it's binary block encoding attribute and not because of it's compression capability. Good data compression can be achieved with Zip and Unzip which is also implemented. The possibility of time measuring with simple tools, such as labels or easily generated text files which are including the test results can give us a good insight into the MT and IMT. From these results we can estimate and predict the consumed time on hardware implementation and also the cost of it.

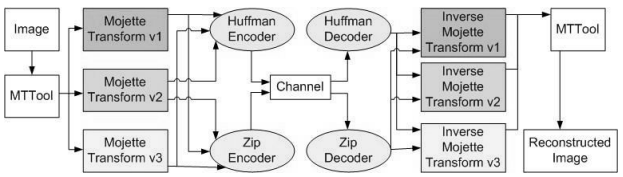


Fig. 5. Logical system architecture of the MTTool.

The time measurement was applied on three different images with three different image sizes and with three different periods. The images were a black and white PGM files with pixel values of 0 and 255 and the LENA.PGM. First test ran only once, after that the second test was running for 6 times in a row, and the last test was running 24 times. Each test was performed on the sizes of 16x16, 32x32 and on 512x512. the results of the two smallest image sizes are nearly identical and the result were nearly always under 20 millisecond for MT and IMT, but on the 512x512 image size we could see the following difference:

TABLE III
TEST RESULT OF THE MT AND IMT WITH THE FIRST VERSION

IMAGE	Black (512x512)		White (512x512)	
	Minute: Second: Millisecond	MT and IMT in Millisecond	Minute: Second: Millisecond	MT and IMT in Millisecond
MT start	57:14:277		0,1621528	
MT end IMT start	57:15:439	1162	0,1623032	1893
IMT end	57:15:910	471	0,1687963	561
MT start	57:22:259		0,1761806	
MT end IMT start	57:23:411	1152	0,1744792	1733
IMT end	57:23:891	480	0,1699653	550

From this we can see that the difference between the black and the white images is more than 50 percent, when it comes to the MT, and only 20 percent when we apply the IMT on the Mojette files.

V. USER INTERFACE OF THE MTTOOL

Basically the MTTool program surface is bade on MDI structure of Microsoft. The parent window contains the main menu and nearly all child windows are opened here. The only exception is the Help/Contents (help.html) which appears outside from the parent window in the default internet browser application.

At the beginning the main goal of the implementation was to represent both the original and the restored image together with the Mojette files which contained the bins of the Mojette transform described in [11,13-17]. We created two image boxes to show the "before and after" images for the user, and we duplicated the information on the two tabbed bar to help the users in comparing the original values, the projections and the inverse values to each other. On the Fig. 6. we can see, the amount of information was too much to check it visually even if the different projections were separated not only by their colors but also were listed after each other. Due to the image size of 512 x 512 we had too many bins to display. That's also one of the many things we decided to make different versions as well.

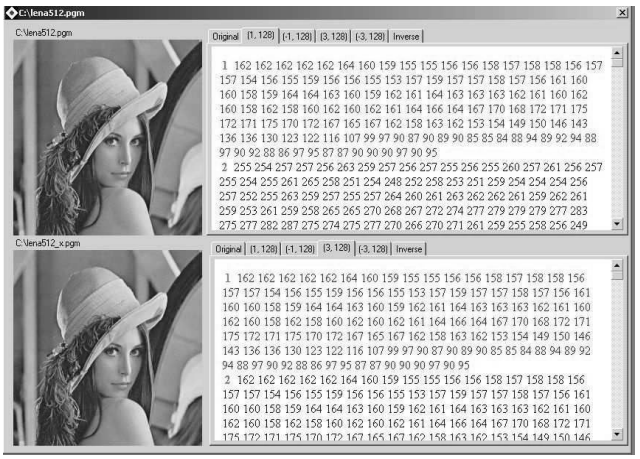


Fig. 6. MT GUI of the first version.

Multiple images in multiple instances can be opened so we are able to compare them not only by the image itself but also by their properties as well we can see the following information from the pictures such as: Original size, Colors used, Pixel size, Size of the DIB-element, Actual size and Last write time.

Later on we developed the Mojette transform in a different way and this gave us the opportunity to create a new interface as well for them (Fig. 7). On an image we can perform the Mojette Transform (Block) with the help of the following child window, where all the steps are visible for a better overview. First we have to choose between two differently based Mojette Transform type such as Algorithm and Matrix. After that we can choose from the predefined Block sizes, which we want to use to perform the MT. Here we will automatically see the Number of blocks, Number of projections, the Projections and the number of bins used for the chosen Block size. If both parameters are set we can click the Next button. As soon as we pushed the Next button we will see the result of the MT with the created Red, Green and Blue bin values grouped together and ordered by the blocks. Here we can check the values of each Block where the MT was performed, and see what are the bin values stored in these Blocks.

Mojette block window (Fig. 7) represents the blocked version of the MT. As it can be seen we can choose between Matrix and Algorithm versions with different blck sizes from 4 up to 32 from this and from the hardcoded Projections the other values are calculated.

Although this information is also stored in a file we are reading it out from the memory. With the help of Show button and the X and Y values we can choose and display the different Blocks and bin values. By hitting the Next button we can create the bin file which contains all the data presented in the step before. Here we have the opportunity to add error to the Blocks, which will result vertical lines in the reconstructed image from the faulty bin file. However we can only see this error on the image if we reconstruct the bin into a BMP file with the IMT which can transform the bin file automatically, so we don't need to set the IMT like we had to do it with the MT.

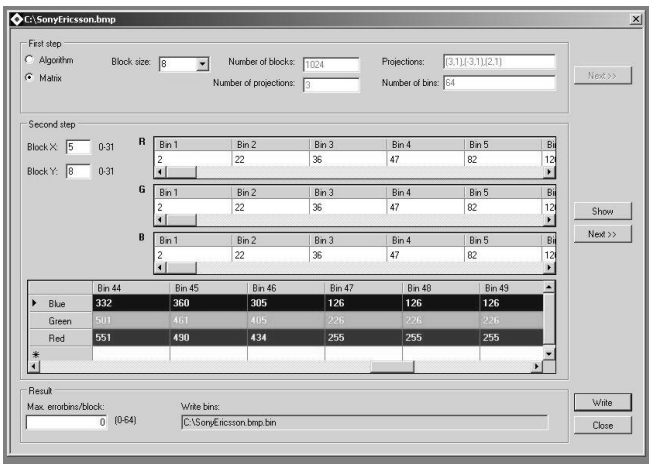


Fig. 7. Mojette transform (Block) window.

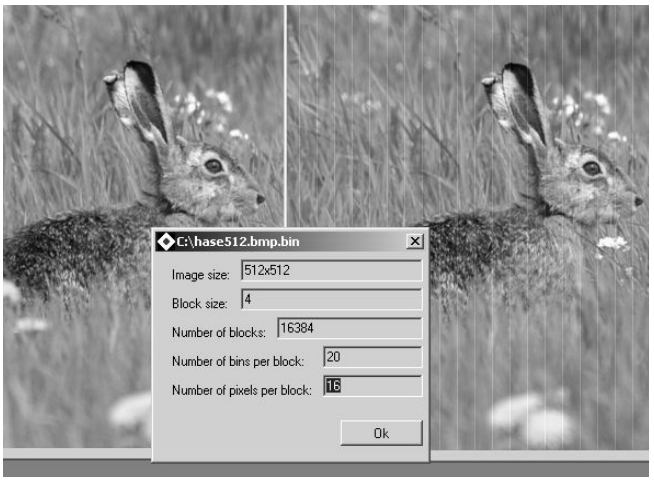


Fig. 8. Information about the restored BIN file in the Inverse MT window.

The Inverse Mojette (Block) window (Fig. 8) shows us some information about the restored image, we can see that the restored image on the right contains error. This was caused in the Mojette Transform phase where we overwrite the value 0 in the 'Max. errorbins/block' field to add some noise to the Mojette Transform. We can also gain information about the MT which was performed on the image.

VI. LOW POWER – LOW COMPLEXITY HW DESIGN METHODOLOGY FOR SELECTED ALGORITHM- MT AND IMT

Power consumption is a very important question of our days. Many energy saving technologies and techniques were developed in different fields. The spread of portable computers and communication devices inspires the developers to design low power consumption devices (CPUs, storage devices, etc.) [8]. In the computer technology there are several options when low power consumption devices are needed (DSPs, μPs). One of these options is using field programmable gate arrays (FPGAs) [9]..

The "MOTIMOT" co-processor denotes the hardware implementation of the direct and inverse Mojette transform as co-processing elements of an embedded processor [12,16,17].

To construct a new hardware for a special task is difficult and expensive both in time and costs [8,10]. Estimating the hardware and software needs to implement a MOTIMOT processing system there is necessary to map the tasks what such a system should implement. These tasks are not limited to the direct and inverse Mojette transformation, but also posses embedded computer functions with or without (depends on the application) real time operating system kernel. Fig.9 shows the functional block scheme of the MoTIMoT co-processor. Both of the Virtex II PRO and Virtex IV FPGAs (FPGAs used for implementation) contain an embedded power PC 405 RISC processor (PPC) [9]. This general purpose processor can manage the MoTIMoT hardware, its driver and the incoming and outgoing data. The MT and IMT blocks are connected to the PPC via the processor local bus (PLB).

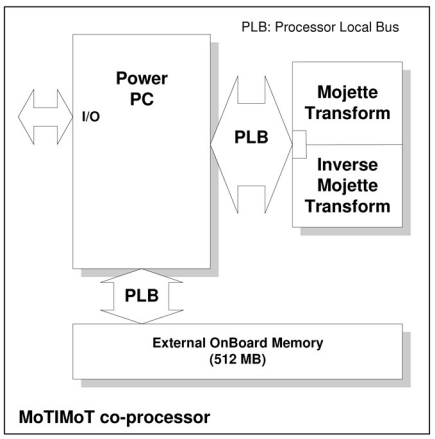


Fig.9. Block schema of MoTIMoT co-processor.

Probably calculation of MT or IMT of an image is not necessary in the same time, therefore only one of the configuration files is loaded into the device. At this point we can use the advantage of partial run-time reconfiguration.

The images are received in real-time thru an Ethernet connection or from a digital camera connected to the platform via an USB port, after processing the image or the frames these are returned to the sender or sent to a client (PC, PDA, etc.). On the motherboard there is an external onboard memory with the size of 512 MB. In our days this resource is available on almost any mobile devices. There are several types of cellular phones which contain place for a micro secure digital card.

A. *Low complexity system design for stepping window method*

Fig. 10 shows the symbolic logic circuitry of MT on a 4x4 image. The input registers ($IR_{x,y}$, x = 1, .., 4; y = 1, .., 4) contain the values of pixels of the image while the output registers ($OR_{i,j}$, i = 1, .., 3; j = 1, .., max_*bin*_number$_i$) contain the values of bins. Three projection lines (1,1; -1,1; 1,2) gives an adequate set of bins and the set of projections meets the reconstruction criteria. The Mojette operator is the XOR logic operator and the different colour (greenscale) of the line means separate the three projection lines and their bins. Fig. 10 represents the MT of a 4x4 pixel size image. This size do not meet with real image sizes of coarse, even so the 4x4 pixel size window is usable in the stepping window method. The register size depends on the width of input data (byte, word, double word) but notice that the larger the data width the more hardware resources are required.

The symbolized logic circuit of the reconstruction (IMT) is depicted in Fig.11. The pixels can calculated in two ways. At first every pixel value calculable from *bin* values only and secondly a pixel value is calculable from *bin* values and the already calculated pixel values. The second version gives more simple (low-complexity) hardware. In Fig. 11 the input registers (IR) contain the *bin*-values, the output registers (OR) contain the original pixel-values, while the IMT operator is the XOR as in the MT was. The number of registers is the same in both cases (MT and IMT).

As a matter of fact the number of input registers can be smaller than the number of bins because of the redundancy.

The single pixel-*bin* correspondences (most of projection lines contain some) give the original pixel value without any calculation this is called zero level. Other bins contain more pixel values. The number of XOR operations need to be performed on them to get the original pixel value is the number of its level. In this case there are five levels from zero to four. In software solution it means a cycle ("for") from zero to five and every cycle contains another cycle ("for") from one to n_{bl} where n_{bl} is the number of bins on the same level. If the window is chosen larger the number of level will increase with it. Using the already calculated pixel values the complexity of the whole system will increase also of course. However increasing of complexity will be slower then when only the *bin* values are used in the calculations.

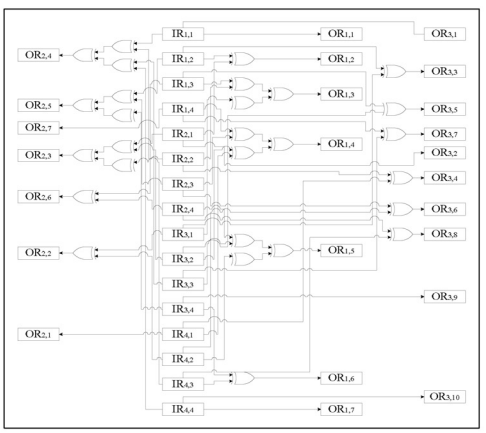


Fig.10. Logical model of the MT on a 4x4 image

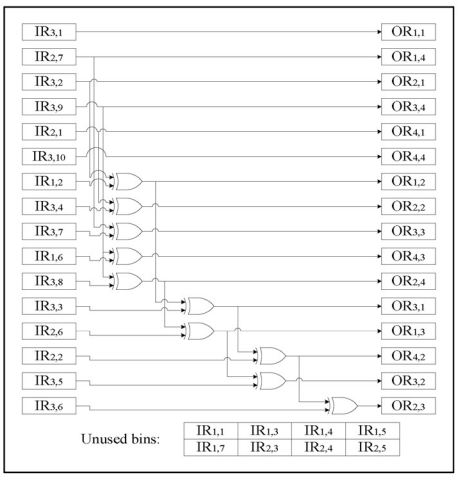


Fig.11. Symbolized logic circuitry of a 4x4 size image reconstruction

Definitely there is a window size limit (WSL) for every FPGA chip. The WSL is the block size what the given FPGA can process in parallel. The WSL depends on the number of logic cells in the given chip and other resources. To enlarge the window size above to the WSL more FPGA chips must be applied. Next figure (Fig.12) shows a MT/IMT computing system. The data source means image or other types of data pre-processing before the MT is not necessary. Post-processing after the MT can be any kind of lossless compression method to decrease the redundancy. Data sink can be a storage device (e.g. PC). In the decoding process the data source is the file of projection lines. Pre-processing is necessary (uncompress the files). Post processing is not necessary here. The data sink in this case is the user's application.

As it is shown in Fig. 12 when larger window size must chosen then one need to multiply the WSL hardware. The sub-picture limited by the broken line shows a multi-chip MT/IMT system. The distribution and merge of data (by the multiplexer and demultiplexer units) also managed by an FPGA chip or if the hardware resources of the FPGA chip allows it, can managed by the on-chip general purpose processor (eg.: the PPC in the Xilinx Virtex II Pro FPGA). A four FPGA solution of the stepping window method for MT is depicted in Fig. 13. The sub-window size is fit to the possibilities of the given FPGA and can not be larger than the WSL.

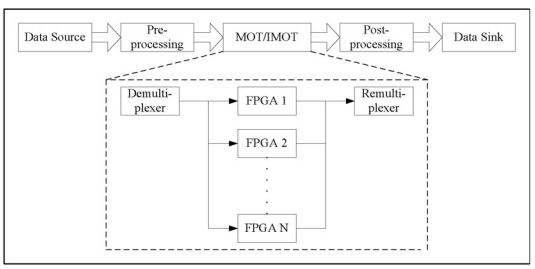


Fig. 12 MT/IMT system block-schema

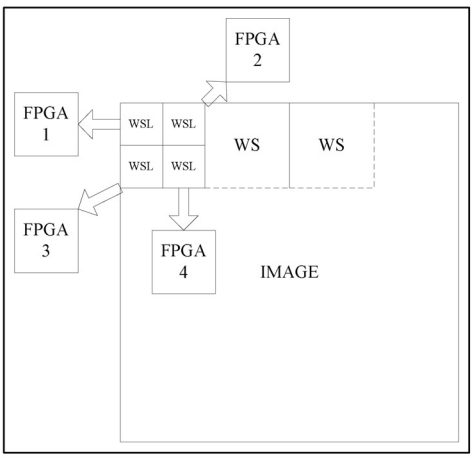


Fig.13. Stepping window method with four FPGAs

With this method the window size (area) equal to the window size limit (area) multiplied by four (if n FPGAs are in use of course the WSL will be multiplied by n). It results shorter computing time but larger power consumption. To find the balance between the power consumption and speed, it is necessary to known where the algorithm of the Mojette transform will be used.

B. Low complexity system design for sliding window method

The sliding window method differs to the stepping window method in its basic. At the stepping window method as it is suggested by its name, there are no common pixels of the windows in two neighbor steps. Contrarily the sliding window method moves the window only with one row or column (depends on the direction of processing) forward. It means most of the pixels are common in two windows, which are neighbors of each other.

Another difference compared to the stepping window method that while at the stepping window method the MT/IMT computing is one single step, the MT/IMT computing has two parts at the sliding window method. First part is to calculate the final value of the given *bins*/pixels and calculate the next temporal value of the other *bins*/pixels in the window and second part is to move the data: write out the final values, move the temporal values into the corresponding registers and read in new data.

Fig. 14 shows the symbolic logic circuitry of the sliding window method where $p_i = \{1, -1, 3, -3\}$ and $q_i = 1$. The virtual image size is defined by P and Q where Q = 4 and P = file size/Q. This means that the size of sliding window is chosen independently the file size. The size of sliding window is given by the hardware resources (number of logic circuits, memory, etc.) of the given FPGA.

Note if q is chosen larger (q=2) the logic circuitry and the size of sliding window (number of registers) will be multiplied by the new q value. When larger window size is required then the WSL, more FPGAs can be used. Four FPGAs using system is depicted in Fig. 15.

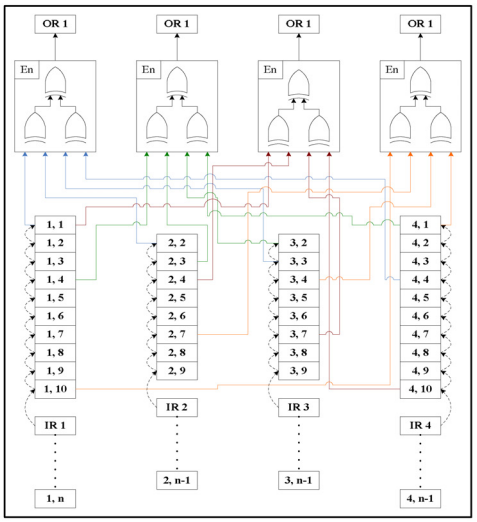


Fig. 14 Symbolic logic circuitry for sliding window method.

Each of the FPGAs contain the same logic circuitry and the size of WSL is the same but the virtual WSL of the whole system is four times larger then the WSL of an individual FPGA. Every FPGA processes a four word width data stream transparently, so the whole system processes a sixteen word width data stream. The projection coordinate q is equal to one in every projection of every FPGA, therefore the corresponding projection lines are merged into one line. Four FPGAs and four projection lines give sixteen projection lines. The merge of the mentioned projections will result four *bin* vectors where the resultant q coordinate is equal to four ($q_r = 4$). This way the permanently processed data width is multiplied by four. The IMT computing system will reconstruct the original data whether it contains only one FPGA (a larger one) or four FPGAs.

Fig. 16 shows the merge process after the MT computing. Every FPGA has four outputs (four *bin* vectors). As it is depicted in the above mentioned figure the corresponding *bin* vectors of the FPGAs are merged into one *bin* vector.

C. IMT

The logic circuitry of the IMT compared with the MT shows that the reconstruction process is more complicated. There are single-pixel *bin* correspondences which result pixel values very simply but the bins of every other projection line need correction. It is necessary, because the *bin* value corrections generate new single-pixel *bin* correspondences and ensures the continuity of the reconstruction process.

The symbolic logic circuitry of the IMT computing SLW co-processor is depicted in Fig. 17. The picture does not show the total system only a part of it. In the image Pi (i=1..8) means the reconstructed pixel values, PLi (i=1..4) are the projection lines, while the rectangles above them represents the bins of the projection lines. In the image "tr" means temporary register which are necessary for the second step, the *bin* value correction. The outputs of Unit 1 are the reconstructed pixel values and it has an enable input. Unit 2 gives the *bin* values after the correction and works with the same enable signal as the Unit 1. In the image (Fig. 17) the bit correction unit is depicted only for one projection line but the other three units are very similar. After the *bin* correction the new values are stored in temporary registers.

In the second phase the correct *bin* values from the temporary registers are written back to the original registers. The third phase is to slide the window. Move the values of the register forward by two places, and read in new data.

The MT and IMT functions are implemented as separate hardware co-processors of the main processor. This is possible in two ways. The main processor can be an off-chip or an on-chip solution. The implemented algorithms are realized as separate co-processors and they work either in parallel or using run-time reconfiguration (This method was not tested yet) using relatively low working frequencies (100-300 MHz). This way can be obtained very high processing speeds.

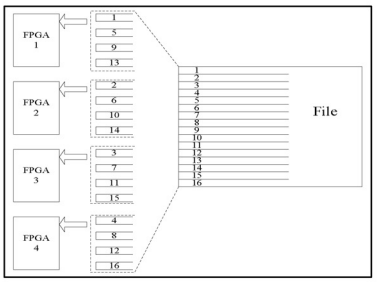


Fig.15. Sliding Window Method with four FPGAs.

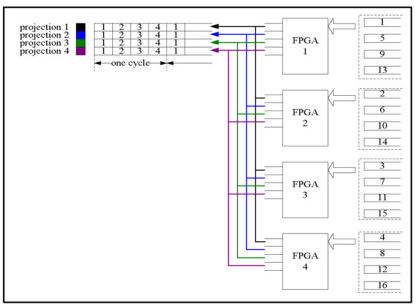


Fig.16. Merge process.

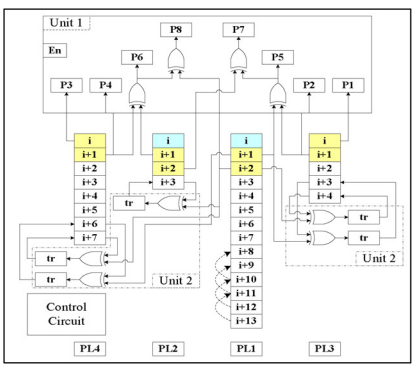


Fig.17. Symbolic logic circuitry for IMT (SLW)

VII. CONCLUSIONS

The paper outlined the different ways, how the MT and IMT is currently implemented in the MTTool. In the MTTool software version more tests should be performed to get more

accurate results, and possible directions should be introduced in near future how the MT should be implemented in a hardware platform. There was presented the hardware implementation of MT and IMT in FPGAs using parallel implementation of the Mojette algorithm. For complex 256x256 size images the MT/IMT can be implemented using multiple FPGA chips as MoTIMoT co-processor.

ACKNOWLEDGMENT

This work was partially supported from the grants VEGA 01/0045/10 and project COST IC0802.

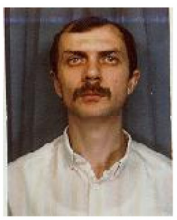
REFERENCES

[1] Guédon J-P., Normand N.: The Mojette Transform: The First Ten Years, Proceedings of DGCI 2005, LNCS 3429, 2005, 79-91.
[2] Guédon J-P., Normand N.: Spline Mojette Transform Application in tomography and communication, In EUSIPCO, sep. 2002.
[3] Guedon, J. P., Parrein, B., Normand, N.: Internet Distributed Image Databases. Int. Comp. Aided Eng., Vol. 8, 2001, 205 – 214.
[4] Parrein B., Normand N., Guédon J-P.: Multimedia Forward Error Correcting Codes For Wireless Lan, Annals of Telecommunications (3-4) 448-463 March-April, 2003.
[5] Normand, N., Guedon, J. P.: La transformee Mojette: une representation recordante pour l'image, Comptes Rendus Academie des Sciences de Paris, Theoretical Comp. SCI. Section, 1998, 124 – 127.
[6] Katz, M.: Questions of uniqueness and resolution in reconstruction from projections. Springer Verlag, Berlin, 1977.
[7] Autrusseau, F., Guedon, J. P.: Image Watermarking for Copyright Protection and Data Hiding via the Mojette Transform, Proceedings of SPIE, VOL. 4675, 2002, 378 – 386.
[8] Turán, J., Ovsenik, L., Benca, M., Turán, J. Jr: Implementation of CT and IHT Processors for invariant Object Recognition System, Radioengineering, Vol. 13, No. 4 2004, 65-71.
[9] Xilinx, *ML403 User Guide*, http:\\xilinx.com/ml403
[10] Wu,J.-Olivier,C.-Chatellier,C.-Poussard,A-M.: Redundant Transformation on the Binary Data and Its Applications to Image Coding and Transmisssion, IEEE APCCAS 2000, Dec 2000, 763-766.
[11] Szoboszlai,P. – Turán,J.: Implementation of the Mojette Transform and HAAr Wavelet in .NET, Carpathian Journal of Electrical Engineering, Vol. 1, No. 1, 2008, 63-68.
[12] Serfőző, P., Vásárhelyi, J., Szoboszlai, P.: Test Result of Mojette Transform Implementation On Embedded PowerPC, Proceedings of 9th International Carpathian Control Conference ICCC 2008, May 25-28, 2008, 24-29.
[13] Serfőző, P., Vásárhelyi, J., Szoboszlai, P., Turán, J.: Performance Requirements of the Mojette Transform for Internet Distributed Databases and Image Processing, OPTIM 2008, Brasov, Romania, May 22-24,2008, 78-82.
[14] Szoboszlai,P.-Turán,J.–Vásárhelyi,J.-Serfozo,P.: Mojette Transform Based Real-Time Video Surveillance System In: Carpathian Journal of Electronic and Computer Engineering, Vol., 3,No.2, 2009, ISSN 1844-9689, 1-6.
[15] Szoboszlai, P., Turán, J., Vásárhelyi, J., Serfőző, P.: The Mojette Transform Tool and Its Feasibility, MACRo 2009, Târgu Mureş, Romania, Acta Universitatis Sapientiae, Electrical and Mechanical Engineering, ISSN 2065-5916, No.1, 2009, 163-174.
[16] Szoboszlai,P.-Vasarhelyi,J.-Turan,J.-Serfőző,P.: Mojette Transform implementation in software and hardware, ESDA'09
[17] Szoboszlai, P., Vásárhelyi, J., Turán, J., Serfőző, P.: MTTool Software Tool and Low Complexity Hardware for Mojette Transformation, Studies in Computational Intelligence, ISBN 978-3-642-15219-1, Springer-Verlag, Berlin, Heidelberg, Vol. 313, 2010, DOI: 10.1007/978-3-642-15220-7, 15-26.

AUTHORS BIOGRAPHY

Ján TURÁN (Prof., Ing., RNDr., DrSc.), was born in Šahy, in 1951. He received an Ing. degree in physical engineering from the Czech Technical University, Prague, in 1974 and a RNDr. degree in experimental physics from the Charles University, Prague, in 1980. He received a CSc. (PhD.) and DrSc. degree in radio electronics from the Technical University, Košice, in 1983 and 1992 respectively. Since March 1979, he has been at the Technical University of Košice, as a professor of electronics and telecommunications technology. His research interests include multimedia signal processing and fiber optics communications and sensing.

Péter SZOBOSZLAI (Ing.) was born in Miskolc,in 1980. He received his Ing.. degree in computer sciences from University of Miskolc, in 2006. From 2006 he worked at Ericsson Ltd., Budapest, in 2008 he joined Magyar Telekom PLC as IT manger. He started his PhD studies at the Technical University of Košice, in 2006 in the field of electronics and multimedia communications. His research interest include image processing, digital signal processing and applications of FPGA.



József VÁSÁRHELYI (ass. prof, Ing., PhD.) was born in Cluj, in 1958. He received his Ing.. degree in electronics and telecommunications from Technical University of Cluj, in 1983. He received a PhD. degree in the field of electronics at Technical University of Cluj in 2004. He joined the Technical University of Miskolc since 1992. His research interest includes applications of FPGA, dynamic reconfiguration systems and their applications, system on chip and embedded systems.

The week when Budapest becomes the ICT capital of the world

Rolland Vida, Head of International Affairs, HTE

Between May 15–20, 2011, Budapest will become the ICT capital of the world, hosting a series of high-standard scientific events related to Information and Communication Technologies and the Future of the Internet, attracting the world's top researchers in these fields and a large international audience. But let us summarize briefly the different elements of this extremely busy and interesting week that awaits us.

VTC2011-Spring
BUDAPEST
Beyond the Generations Game

The 73rd IEEE Vehicular Technology Conference – VTC2011-Spring

15-18 May 2011, Budapest, Hungary

The 73rd IEEE Vehicular Technology Conference will be held 15-18 May 2011 at the Budapest Marriott Hotel. The IEEE VTC conferences have established themselves over the last six decades as one of the main international conferences in vehicular mobile and wireless communications. They are conducted twice each year (Spring and Fall) in key international cities. Budapest follows thus on the steps of Ottawa (Fall 2010), Taipei (Spring 2010), Anchorage (Fall 2009), Barcelona (Spring 2009), Calgary (Fall 2008), Singapore (Spring 2008), Baltimore (Fall 2007), Dublin (Spring 2007), Montréal (Fall 2006) or Melbourne (Spring 2006), and precedes the VTC2011-Fall conference in San Francisco (September 5-8).

The Conference is the flagship event of the IEEE Vehicular Technology Society with over 60 years of history and more than 5000 members. This year's first VTC conference will propose more than 660 oral and poster presentations grouped into 12 tracks: ad hoc and sensor networks, antennas and propagation, cognitive radio and cooperative communications, mobile applications and services, mobile satellite and positioning systems, multiple antenna systems and space-time processing, signal processing for wireless communications, transmission technologies, transportation, vehicular electronics and telematics, wireless access and wireless networks.

Besides these, the program will also feature keynote talks and panel discussions, involving world-class researchers and highly recognized executives from corporate, government and educational institutions and organizations worldwide. Moreover, the first day of the conference will comprise 6 half-day tutorials, on topics such as green communications and networking, cognitive radio based on UWB technology, participatory sensing crowdsourcing data from mobile smartphones in urban spaces, cooperative active and passive localization and tracking, mobility models and social networks, or low-complexity algorithms for large-MIMO detection. Finally, the same day will feature 5 thematic workshops as well: the International Workshop on Self-Organizing Networks (IWSON), the Second Green Wireless Communications and Networks Workshop (GreeNet), Cognitive radio and Cooperative strategies for POWER saving (C2POWER), Broadband evolved Femtocell Technologies (BeFEMTO), and the First International Workshop on Cross-Layer Operation Aided Multimedia Streaming. The organization of these workshops is partly sponsored by the National Innovation Office (NIH).

For more information, please refer to the conference website: (www.ieeevtc.org/vtc2011spring). Registration is already open.

Future Internet Week

16-19 May 2011, Budapest, Hungary

To accomplish the goals of the EU 2020 Strategy and its flagship initiatives, Innovation Union and Digital Agenda, there is a need for advanced communication infrastructures, information technologies, smart applications and services that should be based on a new generation of networks, on the Internet of the Future. The events of the Future Internet Week in Budapest will investigate the policy, technological, and socio-economic aspects of the Future Internet involving many participants from the industry, academia as well as stakeholders from the European Commission and from the Member States.

During these 4 days, the program will include the following events:

- Future Internet Research & Experimentation (FIRE) event – May 16, 2011;
- Internet of Things Conference – May 16, 2011;
- Future Internet Conference – May 17, 2011;

- Future Internet Assembly – May 18-19, 2011;
- Future Internet Forum – May 18, 2011;
- The European Network of Living Labs (ENoLL) event – date to be announced;
- Coordination of the European Future Internet Forum of Member States (ceFIMS) workshop – date to be announced.

The Scientific Association for Infocommunications (HTE) is involved as a local organizer of the Future Internet Assembly, an event partly sponsored by the National Innovation Office (NIH) For more details, please visit the Future Internet Week website (www.fi-budapest.eu).

ICT Proposers' Day
19-20 May 2011,
Budapest, Hungary

The ICT Proposers' Day is Europe's largest matchmaking event dedicated to networking and promoting research and development in the fields of Information and Communication Technologies. Participants have the opportunity to learn about funding possibilities in the fields of ICT offered by the European Union, and about the upcoming Calls 8 and 9 of the FP7 ICT Work Programme, offering more than one billion euros of EU funding. The two half-day sessions (Thursday afternoon – Friday morning) will provide academia and business sectors, SMEs and government actors from all over Europe a platform for building partnerships and exchange ideas for participating in calls for proposals. About 2500 participants are expected to attend. In addition, over 100 Commission officials will be present to provide information on the content of the calls for proposals and advice you on how to submit your proposal.

The event is free of charge; registration will open early February 2011. For more details, please visit the event's website (ec.europa.eu/information_society/events/ictproposersday/2011).

* * *

Besides the fact that in the first half of 2011 Budapest became the Capital of Europe (next to Brussels and Strasbourg) due to Hungary giving the EU presidency, we can also consider it as the ICT capital of world, at least for this much-awaited week in May. We encourage everyone to take this excellent opportunity to meet the world's and Europe's leading professionals in the fields of Information and Communication Technologies, and attend these high quality events. We are looking forward to welcoming you in Budapest!

2010/I *Infocommunications Journal*

2010/II *Infocommunications Journal*

2010/III *Infocommunications Journal*

2010/IV *Infocommunications Journal*

Guidelines for our Authors

Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on
http://www.ieee.org/publications_standards/publications/authors/authors_journals.html#sect2,
"Template and Instructions on How to Create Your Paper".

Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by "1"), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

Accompanying parts

Papers should be accompanied by an *Abstract* and a few *index terms (Keywords)*. For the final version of accepted papers, please send the *short cvs* and *photos* of the authors as well.

Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue

[11] Boggs, S.A. and Fujimoto, N., "Techniques and instrumentation for measurement of transients in gas-insulated switchgear," IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984.

Format of a book reference:

[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.

All references should be referred by the corresponding numbers in the text.

Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. "see Fig. 2."
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

Contact address

Authors are requested to send their manuscripts via electronic mail or on an electronic medium such as a CD by mail to the Editor-in-Chief:

Csaba A. Szabo
Dept. of Telecommunications
Budapest University of Technology and Economics
2 Magyar Tudosok krt.
Budapest 1117 Hungary
szabo@hit.bme.hu

2ND INTERNATIONAL CONFERENCE ON COGNITIVE INFOCOMMUNICATIONS, COGINFOCOM 2011

July 7-9, 2011 Budapest, Hungary

Web: http://www.hte.hu/coginfocom2011

CALL FOR PAPERS

ORGANIZERS

Important dates
Full paper submission: **May 9, 2011**
Acceptance notification: **June 10, 2011**
Camera ready: **June 20, 2011**

Contact
Workshop Secretary
Anna Szemereki
e-mail: anna.szemereki@tmit.bme.hu; coginfocom@gmail.com

Registration Secretary
Mária Tézsla
e-mail: info@hte.hu

The 2nd International Conference on Cognitive Infocommunications (CogInfoCom2011) will be held in Budapest, Hungary, at the premises of the Budapest University of Technology and Economics, in July 7-9, 2011.

Location: **Building of Informatics, Budapest University of Technology and Economics**
Magyar tudósok krt. 2., H-1117 Budapest, Hungary

Cognitive infocommunications (CogInfoCom) investigates the link between the research areas of infocommunications and cognitive sciences, as well as the various engineering applications which have emerged as the synergic combination of these sciences.

The primary goal of CogInfoCom is to provide a systematic view of how cognitive processes can co-evolve with infocommunications devices so that the capabilities of the human brain may not only be extended through these devices, irrespective of geographical distance, but may also interact with the capabilities of any artificially cognitive system. This merging and extension of cognitive capabilities is targeted towards engineering applications in which artificial and/or natural cognitive systems are enabled to work together more effectively.

We invite investigators to submit original research papers as well as review articles on the subject which will help stimulate further contributions to the research field.

Potential topics include, but are not limited to:

- **Affective computing**
- **Augmented cognition (AugCog)**
- **Cognitive infocommunications**
- **Cognitive informatics**
- **Cognitive media**
- **Cognitive robotics**
- **Cognitive science**
- **Etho-engineering**
- **Etho-informatics**
- **Human-robot interaction**
- **Interactive systems engineering**
- **iSpace research**
- **Multimodal interaction**
- **Sensory substitution & sensorimotor extension**
- **Teleoperation**
- **Scientific Visualisation**
- **Virtual Reality Technologies**
- **Media Informatics**

The authors are encouraged to submit full papers describing original, previously unpublished, complete research, not currently under review by another conference or journal, addressing state-of-the-art research and development in all areas of computer networking and data communications. All papers will be reviewed and accepted papers will appear in the conference proceedings, published in IEEE Xplore (confirmation pending).

Papers must be submitted electronically via EDAS in double column IEEE format, 4-6 pages.

Authors of selected outstanding papers will be invited to submit extended versions of their papers for consideration of publication in Infocommunications Journal or Journal of Advanced Computational Intelligence and Intelligent Informatics.

Patrons

Technical Co-sponsors

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its more than 1300 individual members, the Scientific Association for Infocommunications (in Hungarian: Híradástechnikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society. HTE is corporate member of International Telecommunications Society (ITS).

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **Prof. Dr. Gyula Sallai** • *sallai@tmit.bme.hu*

Secretary General: **Pál Horváth** • *phorvath@anotel.com*

Managing Director: **Péter Nagy** • *nagy.peter@hte.hu*

International Affairs: **Rolland Vida, PhD** • *vida@tmit.bme.hu*

Addresses

Office: H-1055 Budapest, V. Kossuth Lajos square 6-8, Room: 422.

Mail Address: 1372 Budapest, Pf. 451., Hungary

Phone: +36 1 353 1027, Fax: +36 1 353 0451

E-mail: *info@hte.hu*, Web: *www.hte.hu*

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its more than 1300 individual members, the Scientific Association for Infocommunications (in Hungarian: Híradástechnikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society. HTE is corporate member of International Telecommunications Society (ITS).

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **Prof. Dr. Gyula Sallai** • *sallai@tmit.bme.hu*

Secretary General: **Pál Horváth** • *phorvath@anotel.com*

Managing Director: **Péter Nagy** • *nagy.peter@hte.hu*

International Affairs: **Rolland Vida, PhD** • *vida@tmit.bme.hu*

Addresses

Office: H-1055 Budapest, V. Kossuth Lajos square 6-8, Room: 422.

Mail Address: 1372 Budapest, Pf. 451., Hungary

Phone: +36 1 353 1027, Fax: +36 1 353 0451

E-mail: *info@hte.hu*, Web: *www.hte.hu*