

# Source Controlled Semi-reliable Multimedia Streaming Using Selective Retransmission in DCCP/IP Networks

Árpád HUSZÁK, Sándor IMRE

Budapest University of Technology and Economics, Department of Telecommunications, Mobile Communications and Computing Laboratory, Magyar Tudósok krt.2., H-1117 Budapest, Hungary  
Tel.: +36 70 514 5925, Fax: + 36 1 463-3263  
Email: [huszak@hit.bme.hu](mailto:huszak@hit.bme.hu), [imre@hit.bme.hu](mailto:imre@hit.bme.hu)

## Abstract

In the past few years we have witnessed an explosive growth in the usage of media streaming applications. The newly appeared audio/video applications are becoming increasingly popular in IP networks, while in mobile environment the limited bandwidth and the higher error rate arise in spite of its popularity. Retransmission-based error recovery is considered inappropriate for multimedia applications, because of its latency. However, this solution can be attractive because it requires minimal network bandwidth, processing cost and efficiently improves the quality. Despite its latency, retransmission can be used successfully in many cases, especially if playout buffering is employed. Only the successfully retransmitted packets will improve the quality parameters of the multimedia stream, therefore it is worth to examine which packets should be retransmitted.

In this paper a source controlled selective retransmission algorithm is presented with a decision algorithm based on the actual RTT and sending rate determined by the TFRC. In our scheme the transmitter determines the playout delay caused by the playout buffer using the proposed Flood method. The needed information about the network congestion state and the network delay are provided by the TFRC (TCP Friendly Rate Control) algorithm. Our proposal does not need additional administration messages because the decision procedure and its inputs are at the transmitter. The obtained results show that significant quality improvement can be achieved with the proposed selective retransmission scheme.

Keywords: Selective retransmission, DCCP, TFRC, Quality of Service

## I. INTRODUCTION

Nowadays, with the rise of multimedia and network technologies, multimedia has become an indispensable feature on the Internet. The newly appeared applications that rely on the real-time delivery of data, such as video conferencing, Internet telephony, and streaming audio/video players are gaining prominence on the Internet. These applications are not only used in reliable wired networks but also in wireless environment where the obstacles of the expansion are the higher bit error ratio of the radio link and the limited bandwidth of the mobile links.

A packet loss generally degrades the performance of any Internet data transfer especially on compressed data. Inter-frame-video compression algorithms such as MPEG exploit temporal correlation between frames to achieve higher compression therefore errors in a reference frame will propagate to the dependent difference frames. To minimize the end-to-end packet loss ratio the packet loss should be either prevented or subsequently handled.

Late retransmissions in real time application are undesirable because the receiver side process already skipped the lost packets. The unrequired retransmissions waste network bandwidth and CPU cycles, contribute to congestion and may delay new data. The time available for recovery may be increased with no perceptible deterioration in quality to the user, by introducing limited buffering at the receiver. This is called playout buffering and the buffering delay is called playout or control delay. For the retransmission to be successful, retransmitted packet must arrive at the receiver in time for playback and should not be lost again due to congestion. When the network is in congested state or the RTT (round-trip-time) is so high that the retransmitted packet will not arrive in time, the retransmission will not improve the quality; moreover will increase the load and latency. To minimize the probability of wastefully retransmitted packets, a playout buffer is usually set up at the receiver side to prefetch a certain amount of data before playback. The buffered data provides additional time to absorb the retransmission delay

making the retransmission acceptable for one-way pre-recorded and one-way live media applications. The generally used transport protocols (TCP, UDP) [1,2] were designed for reliable wired networks, hence large number of packetloss will significantly decrease their performance. New protocols were developed like the Lightweight User Datagram Protocol (UDP-Lite) [3] and the Datagram Congestion Control Protocol (DCCP) [4]. Among the unreliable transport protocols only the Datagram Congestion Control Protocol (DCCP) supports congestion control mechanisms (TCP-Like Rate Control [5] and TCP Friendly Rate Control [6]). DCCP combines the best features of the TCP and UDP protocols within media transmission context, supporting congestion control mechanisms. It may be useful to think of DCCP as TCP minus bytestream semantics and reliability, or as UDP plus congestion control, handshakes, and acknowledgements.

In our proposal we applied DCCP as transport protocol because it uses sequence numbering, acknowledgements and congestion control algorithms. Sequence numbers and acknowledgement are needed to identify the lost packets while the congestion control algorithms manage the actual Round-Trip Time (RTT) measurements.

In this paper a source controlled selective retransmission algorithm is presented with a decision algorithm based on the actual RTT and the sending rate determined by the TFRC. In our scheme the transmitter determines the playout delay caused by the playout buffer using the proposed Flood method. After a congestion period the number of lost packets should be so high that the retransmission of all the lost packets is not worthy. In this case the differentiation of the packets should be made based not only on the playout delay limitations but also on the packet content. In MPEG [7] format the errors in the key-frame proceed to the other frames therefore the retransmission of the damaged part of the key-frame will significantly raise the quality of the video stream.

The rest of the paper is organized as follows. A review of related work in selective retransmission is presented in Section II. In Section III we introduce our source controlled selective retransmission method for multimedia applications. Performance evaluation of our method follows in Section IV. Finally, we summarize our paper and make the conclusions in the last section.

## II. RELATED WORK

Related works base their retransmission algorithms either on the packet importance or on the delay introduced by the retransmission. Some prior work has been done to develop error recovery and concealment for real-time video. The method of

these works can be divided to content based and network characteristic based selective retransmission algorithms.

### A. Content based retransmission

The content based decision algorithms utilize the structure of the data stream. Video sequences are compressed in a format such as MPEG to achieve bandwidth efficiency. Video compression exploits redundancy between frames to achieve higher compression. However packet loss can be detrimental to the compressed video with interdependent frames because errors potentially propagate across many frames. The MPEG video frame structure consist of, I picture, intra coded, coded independently of other frames, P or predictive picture, predicted from the previously decoded picture and B or bi-directionally predictive picture, predicted from one previous and future picture. This is the motivation to protect the important frames like I pictures to avoid the propagation of errors.

Content based retransmission methods retransmit only the important data of the bitstream. These approaches describe a method that includes categorizing groups of packets in order of importance. It takes advantage of the motion prediction loop employed in most motion compensation based codecs. Correcting errors in a reference frame caused by earlier packet loss, prevents error propagation.

Feamster and Balakrishnan [8] analyzed this approach with SR-RTP [9]. This RTP extension provides semantics for requesting the retransmission of independently processible portions of the bitstream and a means for reassembling fragmented portions of independently processible units. They have shown that, by recovery of only the most important data in the bitstream, significant performance gains can be achieved without much additional penalty in terms of latency.

In [10] the selective retransmission of MPEG stream was analyzed with DCCP. This transport protocol provides indispensable information to apply semi-reliable transfer of MPEG video. The results show that the effectiveness of this protocol is considerable when selective retransmission is deployed.

Zheng and Atiquzzaman [11] proposed a new selective retransmission scheme for multimedia transmission over noisy wireless channel using the ATM ABR service. They analyzed the system requirements and minimum receiver buffer size for providing acceptable QoS to the user.

These proposals are effective in networks with high bandwidth where no congestion occurs. For all that it makes no sense to retransmit a lost packet if it

will be lost again or it will cause the loss of other packets.

### B. Network characteristic based retransmission

This type of methods investigates the possibility of successful packet retransmissions in the function of the actual network delay and bandwidth. The decision algorithms decide weather to retransmit a packet without late reception.

Attempts were made to implement a selective retransmission protocol with a decision algorithm [12]. This algorithm decides weather or not to request a retransmission for a packet that was detected as lost. The decision is made by the Euclidean distance calculated by the loss and latency ratio. This method does not use playout buffer and does not implement congestion avoidance mechanisms.

In the scheme introduced in [13], the server decides whether it retransmits packets or not based not only on importance of each packet but also playout time of each packet. In this proposal when the client detects a packet loss, it sends a retransmission request packet to the server. The request message contains some necessary information to make the retransmission decision successful.

The priority of the two type of retransmission method is not the same. The importance of the network characteristic based method is higher because the successful data transmission is the main goal. The packet content should be taken into consideration only if the possibility of the correct receipt of the retransmitted packet is high enough. It makes no sense to retransmit a high importance packet if it will not arrive in time.

In most of the related works the receiver controls the retransmission procedure. The decision algorithm is implemented at the receiver therefore additional administration messages must be sent to the sender. Most of the prior works use NACK (Negative ACKnowledgement) or Retransmission Request messages. In our proposal no administration messages are needed because the decision procedure is located at the transmitter. The other advantage of the transmitter side decision is that the input parameters of the decision algorithm (RTT, estimated link bandwidth, etc.) are available at the source using the DCCP transport protocol.

## III. SOURCE CONTROLLED RETRANSMISSION SCHEME

We propose a DCCP/IP based selective retransmission scheme which disable or enable the retransmission of lost packets according to the current state of the network and the packet content. The Datagram Congestion Control Protocol is a newly defined transport protocol by the IETF that

implements bidirectional, unicast connections of congestion controlled, unreliable datagrams. For our purpose this protocol is ideal because it provides all the needed information for the decision algorithm. The DCCP header contains sequence number field that is indispensable to recognize and identify the lost packets. DCCP connections are congestion controlled, but unlike in TCP, DCCP applications have a choice of congestion control mechanism. Currently two mechanisms are defined: TCP-like Congestion Control, TCP-Friendly Rate Control (TFRC). All kind of congestion control mechanisms need information about the network, like RTT and the packet loss. These are the necessary input for our decision algorithm too.

To make the retransmission possible the receiver must employ a playout buffer while at the DCCP source a retransmission buffer is needed. DCCP does not have a packet retransmission function therefore the server buffers the transmitted packets into a retransmission buffer. To measure the elapsed time from the first packet transmission to the detection of loss the transmission time must be stored.

One of the inputs of the decision algorithm is the delay introduced by the playout buffer at the receiver. In our scheme the playout delay is set up by the transmitter using so-called Flood method. To make the decision of retransmission the elapsed time end the actual playout buffer level will be used.

### 3.1 FLOOD METHOD

To make the retransmission controlled by the source without administrative messages the transmitter must know the playout buffer level and its delay. The proposed solution is called Flood method.

At the beginning of the video transmission the transmitter will not deliver the data packets immediately. It will heap up some data and transmit it all together after  $T_{bd}$  time as illustrated in Fig. 1. With this method the receiver will receive more data than it should process therefore certain amount of data will be heaped up in the receiver's playout buffer. The process rate of the receiver is known by the transmitter because it is equal with the bitrate of the current video so the duration of the Flood method can be calculated easily in theoretical case.

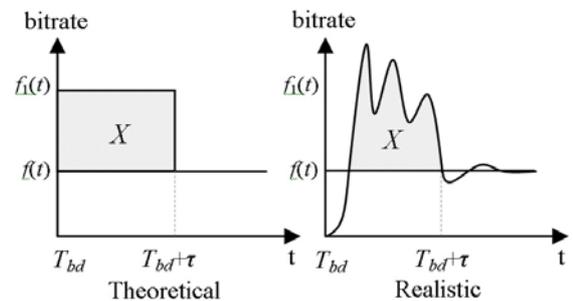


Fig. 1. Flood method

The video source bitrate is  $f(t)$  and the increased sending rate of the transmitter during the flood period is  $f_1(t)$ . In the theoretical case  $f_1(t)$  is constant during the  $[T_{bd}, T_{bd} + \tau]$  time interval. The heaped up data can be calculated as follows:

$$f_1(t)\tau - f(t)\tau = X \quad (1)$$

This extra amount of data ( $X$ ) will be sent in  $\tau$  time.

$$\tau = \frac{X}{f_1(t) - f(t)} \quad (2)$$

Actually the duration of the Flood method depends on the actual network conditions. In the case of DCCP transport protocol the congestion control algorithm (TFRC) will determine the sending rate during this period therefore  $f_1(t)$  is not constant. After all the heaped up data is delivered, the sending rate will be similar to the bitrate of the video stream and the receiver's playout buffer will contain  $X$  amount of data. The time that a data packet will spend in the playout buffer is  $T_{bd}$  that is equal with the heap up time before the transmission.

$$X = T_{bd}(t) \cdot f(t) \quad (3)$$

We assume that the video bitrate  $f(t)$  is constant therefore the buffer delay  $T_{bd}(t)$  will not vary intensively. With this method the playout delay is set up by the transmitter so one of the input data for the retransmission decision algorithm is given.

The arising question is that how to determine the  $X$  amount of data to introduce adequate delay for retransmissions. The playout buffer delay should be long enough to make the packet loss detection and the retransmission feasible but it should be as short as possible. To answer this question first the decision algorithm must be reviewed.

### 3.2 DECISION ALGORITHM

The decision algorithm will decide whether to retransmit a packet or not. The algorithm will work properly if all the retransmitted packets arrive in time and there is no lost packet that was not retransmitted although it would arrive before the playout. The retransmitted packet should not cause the loss or corruption of other packets. First a delay sensitive approach is investigated where the duration of the retransmission process is relevant and after a congestion avoidance retransmission scheme is introduced based on TFRC. The two algorithms should be used simultaneously; therefore the

retransmission of a packet should be disabled if one of the two algorithms blocks the retransmission (logical OR function).

#### A. Delay Sensitive Selective Retransmission Scheme

The available time for the first transmission, retransmission and loss detection is equal to the playout buffer delay, therefore the stipulation to successfully retransmit a packet is

$$\frac{3}{2}RTT + \delta < T_{bd}, \quad (4)$$

where  $\delta$  is the loss detection delay. The upper bound of the loss detection delay is determined by the congestion control algorithm because this algorithm specifies the frequency of the DCCP acknowledgements. The receiver sends DCCP-Ack packets at least once per Round-Trip-Time acknowledging the data packets, unless the sender is sending at a rate of less than one packet per RTT, as indicated by the TFRC specification [RFC 3448]. If the protocol sends ACKs for every packet, the lost packets should be identified with gap detection. According to the specification the loss detection delay is

$$\frac{1}{f_p} < \delta < RTT, \quad (5)$$

where  $f_p$  is the packet sending frequency so the elapsed time between two packets is  $1/f_p$ .

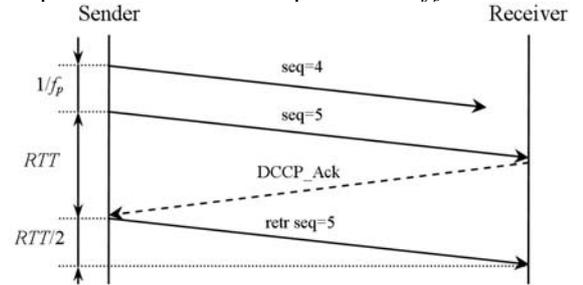


Fig. 2. Transmission sequence

The actual value of the loss detection delay varies, therefore the packet sending time ( $t_1$ ) and the acknowledgement reception time ( $t_0$ ) is used to determine the elapsed time.

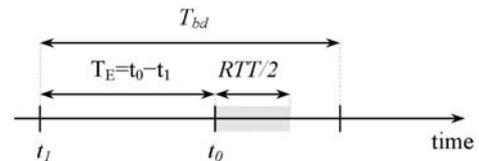


Fig. 3. Time sequence

After receiving an acknowledgement, that contains the sequence numbers of lost packets, the

transmitter should decide which packet is worth to retransmit. The decision algorithm calculates the remaining time that must be less then the one-way network delay for successful retransmission.

$$(t_0 - t_1) - T_{bd} < RTT \quad (6)$$

The limitations of the playout buffer delay ( $T_{bd}$ ) are determined by the streaming application and the network delay. The upper bound of  $T_{bd}$  is the maximal acceptable delay that is few seconds in the case of one-way live media applications but it should not be higher then 150-200ms for two-way interactive applications. In the retransmission point of view the lower bound of the playout buffer delay is determined by the network delay. Our goal is to find lowest delay possible to make the retransmission feasible.

According to (4) and (5) the upper bound of the sum of the time needed for the transmission and the time needed for the loss detection is

$$\frac{3RTT}{2} + \delta \leq \frac{3RTT}{2} + RTT = 2.5RTT \quad (7)$$

Several researchers have analyzed the characteristic of the RTT so far. The Round-Trip-Time is usually modeled with heavy-tailed Gamma [14] or with Normal distribution [15]. First we have used the Normal distribution model  $N(\mu, \sigma^2)$  where  $\mu=RTT$  and  $\sigma^2=(0.1RTT)^2$  to determine the playout buffer delay in our retransmission scheme. The used RTT variance is relatively high according to measured values of [15] because our goal was to minimize the probability of late retransmissions. We have determined the playout buffer delay in such a way that the probability of successful retransmission should be higher then 95%.

$$F(x; \mu; \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^x \exp\left(-\frac{(u-\mu)^2}{2\sigma^2}\right) du = 0.95$$

$$\Rightarrow x = 1.65$$

$$X = \mu + x\sigma = 2.91RTT \approx 3RTT \quad (8)$$

$$X = \mu + x\sigma = 2.91RTT \approx 3RTT \quad (9)$$

In our retransmission scheme we propose to set the playout buffer delay to  $T_{bd}=3RTT_0$ , where  $RTT_0$  is the first RTT measured between the source and destination. If the RTT is too high the  $T_{bd}$  should be set to the highest acceptable delay depending to the application demands. According to the calculated playout buffer delay the amount of the heaped data can be determined as given in (3).

We have calculated the maximum variation of the RTT to find the limitations of the algorithm. According to the following equation the highest

RTT variation is  $(0.124 RTT_0)^2$  to enable the 95% of the retransmissions.

$$F(x; \mu; \sigma) = F(3; 2.5; 0.124) = 0.95 \quad (10)$$

We have investigated the proposed method modeling the network delay with heavy-tailed Gamma distribution too.

$$\Phi(x, \alpha, \beta) = \frac{x^{\alpha-1} \cdot e^{-\frac{x}{\beta}}}{\beta^\alpha \Gamma(\alpha)}, \quad x > 0 \quad (11)$$

The mean value and the variance can be calculated as follows:

$$E(x) = \alpha \cdot \beta \quad (12)$$

$$\text{Var}(x) = \alpha \cdot \beta^2 \quad (13)$$

To use the same mean value ( $RTT$ ) and variation ( $0.1RTT$ ) for the Gamma distribution as for the Normal distribution, the Gamma distribution parameters should be set to  $\alpha=100$  and  $\beta=0.01$ .

$$E(x) = \alpha \cdot \beta = 1, \quad \text{Var}(x) = \alpha \cdot \beta^2 = 0.1^2 \quad (14)$$

To successfully retransmit the 95% of the lost packets the inverse Gamma function should be calculated considering the upper bound of the sum of the time needed for the transmission and the time needed for the loss detection ( $2.5RTT$ , as given in (7)).

$$\Phi(x; 2.5\alpha; \beta) = \frac{x^{2.5\alpha-1} \cdot e^{-\frac{x}{\beta}}}{\beta^{2.5\alpha} \Gamma(2.5\alpha)} = 0.95 \quad (15)$$

$$x = 2.76RTT \approx 3RTT \quad (16)$$

The Gamma model provides similar results as the Normal distribution model. We have showed that  $T_{bd}=3RTT_0$  is sufficient playout buffer size to realize the retransmissions. Setting the playout buffer to  $T_{bd}=3RTT_0$  higher RTT variance should be accepted. The  $x=3 RTT_0$  and 0.95% probability with Gamma distribution should be reached with  $\alpha=28.5$  and  $\beta=0.035$  (mean value  $E(x) = \alpha \cdot \beta = 1$ )

$$\Phi(x; \alpha; \beta) = \Phi(3; 28.5; 0.035) = 0.95 \quad (17)$$

With setting the playout buffer size to  $3RTT_0$  even  $(0.18RTT_0)^2$  variance is acceptable with 95% of retransmissions (mean value is  $RTT_0$ ). This variance level is high enough to make the delay sensitive retransmissions successful so both distribution models confirmed the proposed playout buffer setup process.

## B. TFRC-based Selective Retransmission Scheme

The retransmission should be enabled or disabled according to the stream bitrate and the offered TFRC sending rate. When the network is in congested state or near to this state the retransmissions should be disabled. When the buffers of the network routers are overloaded the additional load will make the things worst. A retransmitted packet will be dropped at the routers or it will cause the loss of other packets. The proposed method uses the TFRC congestion avoidance algorithm to decide whether the lost packet should be retransmitted.

TFRC is an equation based congestion control algorithm and it can only estimate the free bandwidth in the network. The algorithm proposes a sending rate in function of the network parameters like RTT, packet loss ratio.

$$T = \frac{s}{R\sqrt{\frac{2p}{3}} + 4R(3\sqrt{\frac{3p}{8}} \cdot p \cdot (1 + 32p^2))} \quad (18)$$

The equation uses weighted average of the parameters, to avoid the radical decrease or increase of the sending rate. Hereby the effect of a single packet loss is not so significant like in the TCP-like mechanism. Single packet losses usually occur due to wireless channel failure without need to reduce the sending rate.

The bitrate of a MPEG source may vary but we suppose that it is usually below the calculated TFRC sending rate when the network is not overloaded. When it is overloaded the estimated packet error ratio and the RTT (which may fluctuate during congestion) will be higher causing the decrease of TFRC sending rate. The video bitrate and the TFRC sending rate is independent so in the case of network congestion the determined TFRC sending rate should be lower than the multimedia stream rate. Our proposal is to disable the retransmission when the TFRC rate is under the video bitrate.

$$X_{MPEG}(t) > \frac{s}{R\sqrt{\frac{2p}{3}} + 4R(3\sqrt{\frac{3p}{8}} \cdot p \cdot (1 + 32p^2))} \quad (19)$$

To save on bandwidth, such video streams are often compressed, which leads to highly bursty, variable bit-rate (VBR) output streams. Transmitting a VBR stream over packet switched networks is difficult without packet loss due to congestion, or without wasting substantial bandwidth with a peak rate reservation.

Our mechanism proves to be effective when the TFRC sending rate varies near the video bitrate or

the video bitrate is high enough to reach the TFRC rate. In other situations the TFRC-based selective retransmission method is applicable too, but of course it will enable the retransmission for the whole duration of the video transmission when the TFRC rate is much higher than the video bitrate and disable when it is lower.

Investigations were made to regulate video quality to adjust video rate to the desired sending rate which is determined by TCP-Friendly Rate Control algorithm [16]. Although it is recommended that the TFRC system regulates sending rate more than once a RTT, it is unrealistic to control video coding quality so frequently, in some cases, at the rate higher than video frame rate. With adaptive coding mechanisms difference between the TFRC rate and the video rate is minimal but not zero. In these solutions the retransmission determination function varies its output frequently.

$$A(t) = \text{sgn}(X_{MPEG}(t) - X_{TFRC}(t)) \quad (20)$$

The retransmission should be enabled according to variable  $A(t) \in \{-1, 1\}$ . The retransmission is enabled when  $A(t)$  is -1 and it is disabled when  $A(t)=1$ .

The TFRC-based retransmission method does not need any additional traffic load to manage its functionality. All the needed information is provided by the DCCP protocol and the integrated congestion control algorithms. The transmitted packets should be stored in buffer on the server side to later retransmit the lost ones if possible.

#### IV. SIMULATION RESULTS

In order to test the performance of the selective retransmission scheme, described in the previous section, we analyzed some scenarios with ns-2 [13] network simulator. The simulation environment made it possible to adjust the link characteristics and analyze the effects of additional background traffic. We used a constant bitrate video stream in the simulations but of course our method is applicable for variable bitrate streams too. The simplex test network is illustrated in the following figure (Fig. 4.).

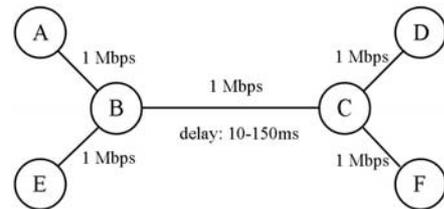


Fig. 4. Network topology

The analyzed video stream is transmitted from node A to node D in DCCP/IP packets, while the background traffic is generated by node E and

received by node F. This background traffic uses UDP with variable bitrate and TCP. The TFRC-based selective retransmission method is implemented in node A. The bandwidth of the links is 1 Mbps that is high enough for all test scenarios. Node B uses a DropTail (FIFO) queue with length of 10 which should be overloaded in case of congestion.

In the first scenario we have analyzed the retransmission probability of the lost packets using the delay sensitive retransmission scheme. The packet loss probability was set to 1% while the network delay was varied between 10ms and 150ms; the video stream bitrate was 384kbps. During the 150s simulation time about 5000 packets were sent. The packet size was 1500 bytes.

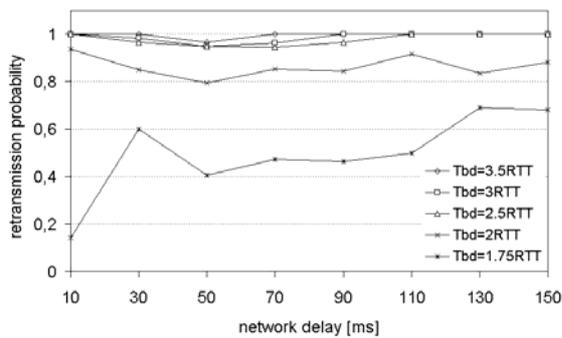


Fig. 5. Retransmission probability

On ratio of retransmitted packets to all lost packets, we compare different playout buffer delay setups. The results illustrated in Fig. 5. show that with  $T_{bd}=3RTT_0$  most of the lost packets can be retransmitted.

Examinations were made to find the limits of the playout buffer delay in the retransmission point of view. Fig. 6. shows the maximal playout buffer delay when no retransmission is possible and the minimal playout buffer delay when all the packets can be retransmitted.

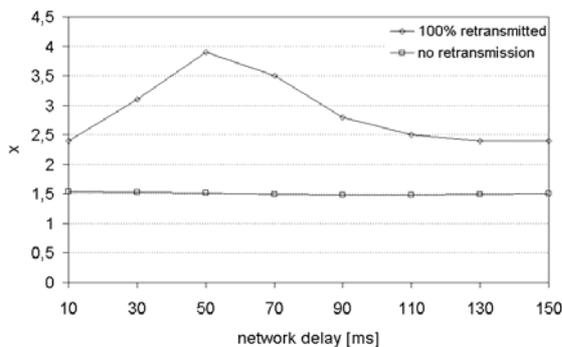


Fig. 6. Retransmission probability,  $T_{bd}=xRTT_0$

The lower bound of  $T_{bd}$  is  $1.5RTT_0$ . In the simulations there were always some packets that were detected as lost in short time. The other reason

of receiving some retransmitted packets in near  $1.5RTT_0$  time is that the  $T_{bd}$  is defined by the RTT at the beginning of the transmission. If the initial RTT is higher than the actual, some retransmission should be enabled if  $T_{bd}=1.5RTT_0$ .

The main goal of our proposal is to improve the quality of video streams. To see the effectiveness our source controlled selective retransmission scheme we measure the average peak signal-to-noise ratio (PSNR) of the transmitted video stream with different  $T_{bd}$  settings. PSNR values measured against the original frames with varying degrees of packet loss. As the packet loss increases the frame-by-frame PSNR drops dramatically. In the PSNR measurements the network delay was 90ms.

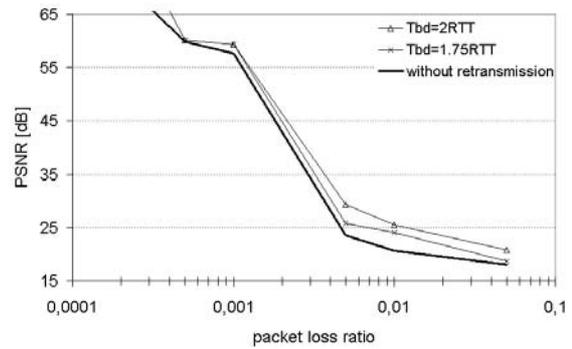


Fig. 7. Average PSNR of 150 seconds of 24 fps video as a function of packet loss

We propose to set the playout buffer delay to  $3RTT_0$  because in this case almost all of the lost packets can be retransmitted; therefore the measured average PSNR value is similar to the original video stream. The video quality improvement with  $T_{bd}=2RTT_0$  is between 2-3dB while in the case of  $T_{bd}=1.75RTT_0$  it was 1-2dB. The frame-by-frame PSNR comparison of the video stream with retransmission ( $T_{bd}=1.75RTT_0$ ) and without retransmission is illustrated in the next figure (Fig. 8.).

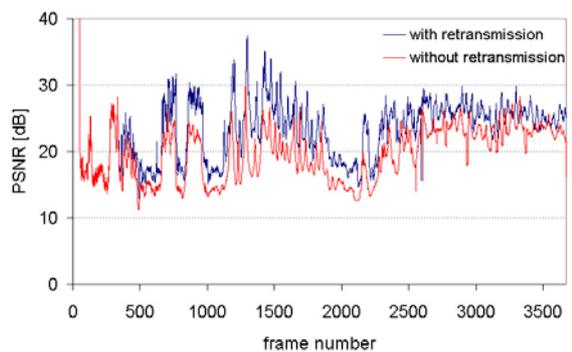


Fig. 8. Frame-by-frame PSNR

The acceptable playout buffer delay is limited by the type of the streaming application. In some situations it is not possible to set the  $T_{bd}$  to  $3RTT_0$

but it is worth to set to its maximal acceptable value. Fig. 8. shows that even with  $T_{bd}=1.75RTT_0$  quality improvement can be achieved.

To obtain these results no additional messages were needed, because the DCCP transport protocol has provided all the necessary information for the Flood method and for the decision algorithm.

In order to analyze the TFRC-based retransmission scheme we used a constant bitrate video stream in the simulations but of course our method is applicable for variable bitrate streams too. In the case of variable bitrate streams the current coding rate must be known. In the simulations the video bitrate was always lower than bandwidth of the links therefore congestion was caused only due to the background traffic.

In the next scenario the background traffic is off and the links are reliable therefore no loss occurs due to channel unreliability. The calculated sending rate by TFRC and the actual sending rate of the MPEG stream are illustrated in the next figure (Fig 9.).

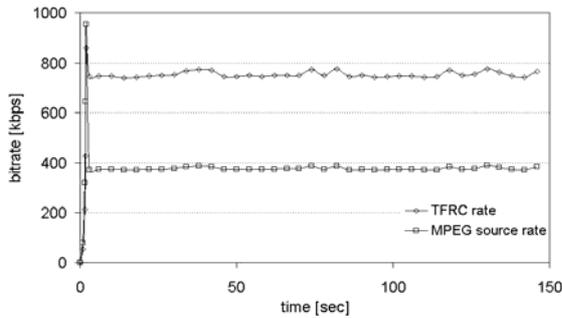


Fig. 9. TFRC rate and MPEG source rate without background traffic and without packet loss

The difference between the calculated TFRC rate and the video stream rate is high; hence according to our method the retransmission was enabled for the whole duration of the simulation.

The TFRC sending rate is significantly influenced by the packet loss ratio. The reason of the packet loss is not differentiated by the source therefore the loss due to congestion and channel unreliability has the same effect on the loss ratio parameter used in the TFRC equation. The only difference is in the RTT variation so it should be taken into consideration what is already done by the TFRC algorithm. It uses weighted average of loss ratios where a single loss has no significant effect on this estimation. The weighted average function smooths the variation of loss ratio so the TFRC-based algorithm should not care on the RTT in addition.

We have analyzed the TFRC-based retransmission scheme with different packet loss ratios on the B-C link to find the limits of the method. The background traffic is still off but the TFRC

algorithm radically reduces its suggested sending rate.

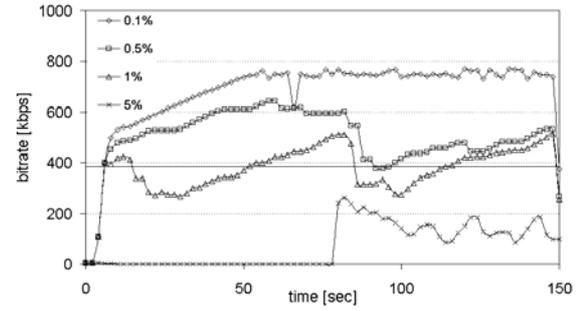


Fig. 10. TFRC sending rate in function of packet loss

The results show that the retransmission is enabled all the time when the packet loss is lower than 0.5%, the video stream rate is 386 kbps and the measured RTT is about 140 ms. The selective retransmission scheme will probably deliver all the packets in this case. Only those packets will be missing that are lost again during the retransmission. On extremely bad channels where the packet loss is 5% the connection should not be built up for a long time because the DCCP-Request and DCCP-Response packets were lost too. In the case of 1% packet loss ratio only a 270 kbps stream should be transmitted without disabling the retransmission.

The retransmitted packets significantly increase the MPEG video quality especially when I-frame data has been delivered correctly to client. We made our examinations for MPEG-2 video streams but of course obtained improvement of quality is true for other audio and video stream formats too.

As we mentioned before the TFRC-based selective retransmission is efficient when the TFRC sending rate varies near the video bitrate or the video bitrate is high enough to reach the TFRC rate. In the second scenario when the packet loss probability is 1%, two periods are determined when the retransmission is disabled. In spite of these periods the video quality is improved. Fig. 11. shows the evolution of video quality due to retransmissions in enabled periods.

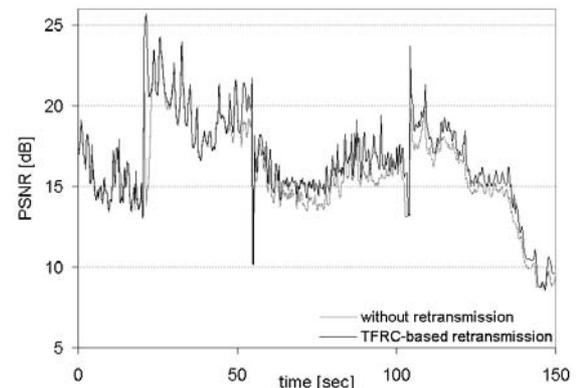


Fig. 11. MPEG video quality improvement

The average Peak Signal to Noise Ratio (PSNR) of the stream without retransmission is 15.86 dB. With the selective retransmission method it is 16.6 dB. Peak Signal to Noise Ratio is a coarse and controversial indicator of picture quality that is derived from the root mean squared error (RMSE). In these scenarios the RTT was roughly constant (about 140 ms) and the TFRC sending rate variation was due to packet loss occurrences. In the following tests the RTT varies according to the level of congestion. In the test network the level of congestion is equal with the buffer level of node B. To analyze the TFRC-based selective retransmission method in congested network the background traffic is set on. In this scenario the total bandwidth demand of the background traffic and the video stream is higher than the available link capacity in short periods. The available free capacity of the B-C link is shown in the next figure (Fig. 12.).

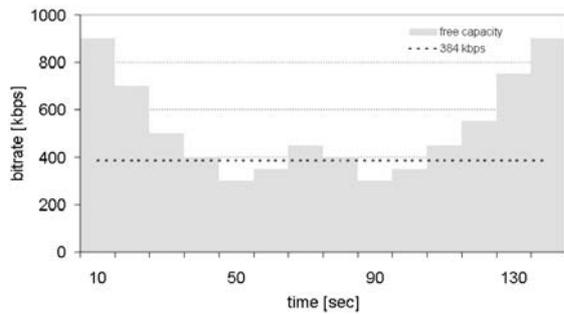


Fig. 12. Available bandwidth on B-C link

The packet drop probability of the B-C link is 0.1% but the large number of packet drops is due to the overflow of the buffer of node B. In 150 seconds about 4800 packets were transmitted from which 5 was corrupted due to channel corruption and about 50 due to congestion.

Due to the large number of packet drops and the increase of RTT the TFRC varies the sending rate to find the highest sending rate.

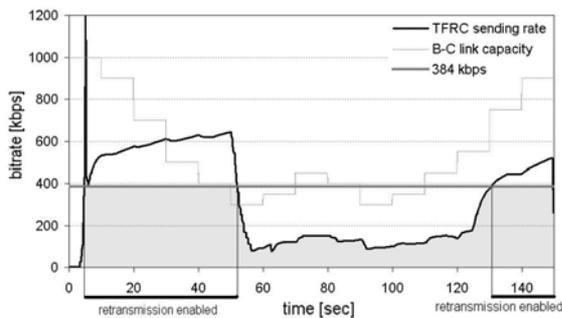


Fig. 13. TFRC controlled retransmission

The video rate is 384 kbps in the scenarios therefore this is the actual sending rate of the source although the TFRC specifies higher. Nevertheless the actual sending rate will be the TFRC rate when it is higher than the video rate. The periods when the retransmission is enabled according to our scheme is illustrated in Fig. 13.

The TFRC reduces the offered sending rate immediately below the video rate when congestion occurs. From this moment the retransmission is disabled. The TFRC will increase the offered rate after the buffer of node B is getting empty and the measured RTT is decreasing. It takes time to pour out the packets therefore the TFRC rate increase is restrained.

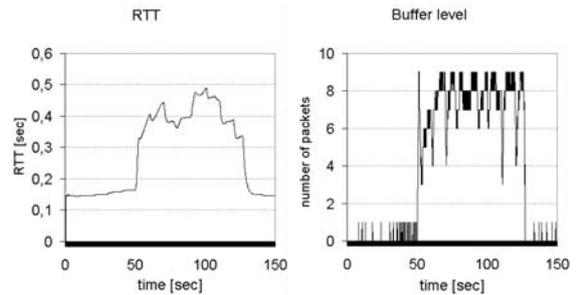


Fig. 14. RTT and buffer level of node B

Usually the two proposed algorithms make the same decision, because during congestion the network delay is too high, so the delay sensitive scheme will also disable the retransmission. In real time applications retransmission is not recommended when the RTT is high. Our TFRC-based selective retransmission method indirectly takes the RTT into consideration. The Fig. 14. shows that in the 50-130sec period the RTT values are extremely high. As Fig. 12. illustrates our method disables the retransmissions in the same time period. In congestion periods the probability of packet delivery is lower and the introduced delay is much higher so the retransmitted packets should not arrive to the client in time.

According to presented simulation results the TFRC-based retransmission scheme effectively manages the additional load due to retransmissions and effectively improves the video quality. The attained gain of video quality in the analyzed scenario is presented in Fig. 15 and Fig. 16.

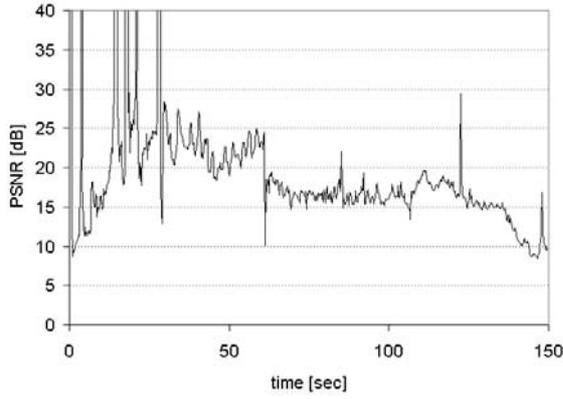


Fig. 15. MPEG video PSNR without retransmission

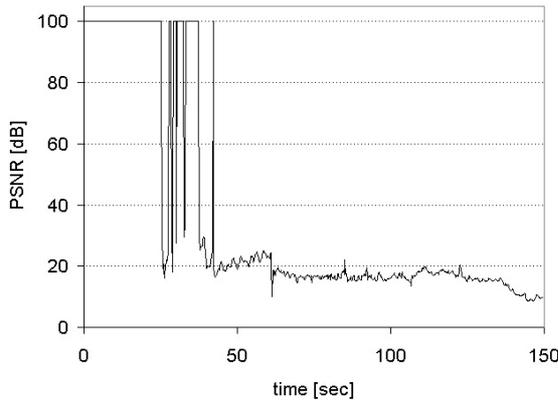


Fig. 16. MPEG video PSNR with TFRC-based selective retransmission

The analyzed network was congested in 50% of the time. In this scenario the video quality improvement is significant in the first 50 seconds while in the congested period there is no difference between the two cases (without retransmission and TFRC-based retransmission). In the last period of the simulation, when the retransmission is enabled again, only one packet was retransmitted that was lost due to channel unreliability. The PSNR analysis shows the significant difference in the first period. From the 130 second all the packets are delivered correctly therefore there is no difference from the original stream. The measured average PSNR using the TFRC-based retransmission scheme is 36 dB while without it 19.9 dB.

## V. TESTBED MEASUREMENTS

In order to analyze the efficiency of the proposed methods we have deployed a DCCP/IPv4 test network. The two DCCP (CCID3) terminals were located in the same local network. To insert packet drops we used *Netem* [17] which provides Network Emulation functionality for testing protocols by emulating the properties of wide area networks. It can emulate variable delay, loss, duplication and re-ordering. The configuration is as shown below:

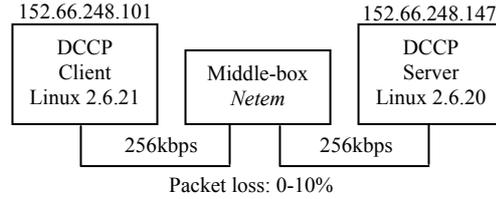


Fig. 17. Test configuration

The *Netem* middle-box can emulate link delays in order to analyze the communication of distant terminals. For the correct delay setup we have measured the RTT values (mean and deviation) of well-known streaming provider (youtube.com) and the RTT between the DCCP client and server without the *Netem* middle-box.

Table 1. RTT measurements

RTT	client – server	client – youtube.com
mean	0.125ms	182.041ms
variance	0.0066ms	0.972ms
	5.3% of the mean	0.5% of the mean

The result showed that 10% RTT variance is acceptable for the playout buffer calculations. The RTT variance was higher than  $0.1RTT_{mean}$  only for 1.5% of the packets in the case of client-server RTT measurements. For client-youtube.com there were no packets with RTT variance higher than  $0.1RTT_{mean}$ .

The delay sensitive selective retransmission is based on the RTT measurement of the TFRC algorithm. This RTT is used to schedule the acknowledgement sending, the receiver should send at least one ACK per RTT. This feature was used to estimate the loss detection time.

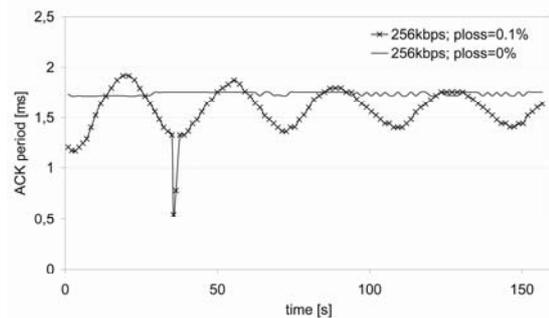


Fig. 18. ACK sending periods (RTT=1.71ms)

The figure shows the DCCP ACK sending period with 0.1% packet loss and without any loss. When there are no losses the time distance between the ACKs is very close to the RTT value. The ACK sending frequency in case of packet losses is generally below the RTT value that causes faster loss detection.

In order to analyze the TFRC-based retransmission we have measured the proposed TFRC rate on an unreliable channel. The packet loss and the delay

variations caused the adaptation of the sending rate to the current network conditions.

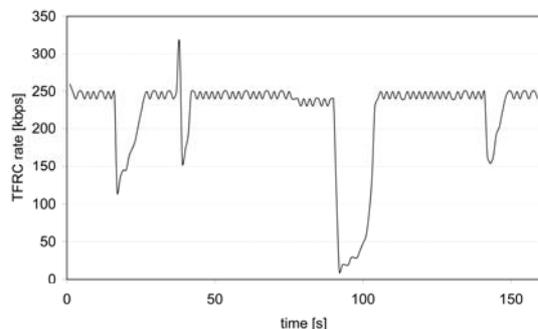


Fig. 19. TFRC sending rate

Using *Netem* network emulator tool 0.1% to 1% packet loss was set without additional delay. At the 90 second 10% loss and 10ms extra delay was set for few seconds to simulate congestion. Until the TFRC rate does not reach the stream rate again the retransmission is disabled.

During the 160s examination time 27 packet loss was registered from which 6 was lost during the congestion period. The 216kbps video stream PSNR was 24.1dB without retransmission and 24.42dB with the TFRC-based retransmission algorithm.

The proposed semi reliable solutions are advantageous for applications with high bandwidth demand and of course causing higher probability of congestion. The deployed playout buffer also determinates the application types because it delays the multimedia stream with 3RTT. This delay is acceptable for one-way pre-recorded and for one-way live media applications but the acceptability for two-way real time applications highly depends on the actual RTT.

## VI. CONCLUSION

In order for video streaming to succeed on the Internet, systems must account for the anomalies of packet loss and changes in bandwidth and delay that make the delivery of real-time video on the Internet challenging. These multimedia applications can tolerate small amounts of data loss but with retransmissions the loss ratio can be minimized improving the quality of the video stream. For live multimedia applications the retransmission can be realized with playout buffer deployed at the receiver.

In this paper we proposed a new source controlled retransmission scheme with two type of decision algorithm. The main advantage of the proposed source controlled mechanism is that all the needed input parameters of the decision algorithm are available at the transmitter side without any additional administrative messages. We also defined the sufficient playout buffer delay to make

the retransmission suitable for the proposed scheme.

We have analyzed the effects of packet loss on the quality of MPEG video and proposed a model to improve the quality of service. We have shown that by recovering the corrupted or lost data in the bitstream, considering the current state of the network, significant performance gain can be achieved without much additional penalty in terms of latency. The algorithm for deciding whether or not to retransmit a lost packet, adapts itself not only to each application, but also to alternate network conditions. Thus, the flexibility and performance of the source controlled selective retransmission scheme, provides a potential framework for Internet multimedia applications to achieve better quality.

## VII. ACKNOWLEDGEMENT

This work was supported by the ETIK (Inter-University Cooperative Research Centre for Telecommunications & Informatics), the Mobile Innovation Centre (MIK) and the ANEMONE project which is partly funded by the Sixth Framework Programme of the European Commission's Information Society Technology.

## REFERENCES

- [1] J. Postel: "Transmission Control Protocol", RFC 792, Internet Engineering Task Force, September 1981.
- [2] J. Postel: "User Datagram Protocol", RFC 768, Internet Engineering Task Force, August 1980.
- [3] Larzon, Degermark, Pink: "The Lightweight User Datagram Protocol (UDP-Lite)", RFC3828, Internet Engineering Task Force, July 2004.
- [4] Kohler, Handley, Floyd, "Datagram Congestion Control Protocol", Internet Engineering Task Force, March 2006, RFC 4340.
- [5] S. Floyd and E. Kohler, "Profile for DCCP Congestion Control ID 2: TCP-like Congestion Control", draft-ietf-dccp-ccid2-10.txt, March 2005.
- [6] S. Floyd, E. Kohler, and J. Padhye, "Profile for DCCP Congestion Control ID 3: TFRC Congestion Control", draft-ietf-dccp-ccid3-11.txt, March 2005.
- [7] I. I. S. 13818. "Generic Coding Of Moving Pictures And Associated Audio Information", 1994.
- [8] M. Feamster, H. Balakrishnan, "Packet Loss Recovery for Streaming Video", 12th International Packet Video Workshop, Pittsburgh, PA, April 2002.
- [9] A. Miyazaki, A. H. Fukushima, K. Hata, T. Wiebke, R. Hakenberg, C. Burmeister, Matsushita, "RTP payload formats to enable multiple selective retransmission", Internet Engineering Task Force, draft-ietf-avt-rtp-selret-04.txt, Nov. 2001.

[10] A. Huszak, S. Imre, "Selective Retransmission of MPEG Video Streams over IP Networks", CSNDSP 2006, Patras, Greece, July 2006.

[11] Bing Zheng, Mohammed Atiquzzaman, "Network Requirement for Management of Multimedia over Wireless Channel", Lecture Notes In Computer Science Vol. 2496, Proceedings of the 5th IFIP/IEEE International Conference on Management of Multimedia Networks and Services: Management of Multimedia on the Internet, London, UK, 2002.

[12] Mike Picuch, Ken French, George Oprica, Mark Claypool, "A Selective Retransmission Protocol for Multimedia on the Internet", Proceedings of SPIE International Symposium on Multimedia Systems and Applications, Boston, MA, USA November 2000.

[13] H. Hagino, Y. Miyazaki, Y. Onoe, Y. Atsumi, H. Komaki, M. Taniguchi, N. Yamanouchi, "A Payout Time Oriented Retransmission Scheme for Multimedia Streaming Systems", HSNMC 2003, Estoril, Portugal, 23-25 July 2003., pp406-415

[14] Mark Kalman, Bernd Girod, "Modeling the Delays of Successively-Transmitted Internet Packets", IEEE Conference on Multimedia and Expo, Taipei, Taiwan, June 2004.

[15] Yasuhiro Sato, Shingo Ata, Ikuo Oka, Chikato Fujiwara, "Using Mixed Distribution for Modeling End-to-End Delay Characteristics", Proceedings of the 8th Asia-Pacific Network and Management Symposium (APNOMS 2005), Japan, Okinawa, September 2005.

[16] Naoki Wakamiya, Masaki Miyabayashi, Masayuki Murata, and Hideo Miyahara, "MPEG-4 Video Transfer with TCP-Friendly Rate Control", LNCS 2216, p. 29 ff., 2001

[17] Stephen Hemminger, "Network Emulation with NetEm", LCA 2005, Canberra, Australia, April 2005



Sándor Imre received the M.Sc. degree in Electronic Engineering from the Budapest University of Technology (BUTE) in 1993. He obtained dr. univ. degree in 1996 and the Ph.D. degree in 1999. Currently he is carrying his teaching activities as

Associate Professor at Dept. of Telecommunications of BUTE. He is member of Telecommunication Systems Committee of the Hungarian Academy of Sciences. He was invited to join the Mobile Innovation Centre as R&D director in 2005. His research interest includes mobile and wireless systems. Especially he has contributions on different wireless access technologies, mobility protocols and reconfigurable systems.



Árpád Huszák graduated with M.Sc. degree in Electronic Engineering from the Budapest University of Technology (BUTE) at the Department of Telecommunications, in 2003. He is currently a third year PhD student at the same university, student

member of the IEEE, and member of Mobile Communications and Computing Laboratory. His research interest includes mobile computing, multimedia communication over broadband and wireless networks and network protocols.