Selective Retransmission of MPEG Video Streams over 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

{huszak, imre}@hit.bme.hu

Abstract- **Internet multimedia streaming is becoming increasingly popular to access multimedia information. Two major issues arise in spite of its popularity. First, limited bandwidth restricts high bit rate video transmission. Second, wireless transmission by its nature may introduce higher rate of errors during transmission. The frequent errors should cause deterioration of the quality of multimedia streams. In this paper we present a selective retransmission algorithm for video streaming over IP networks where the video is streamed using DCCP transport protocol. The goal is to minimize the number of the corrupted bits, with acceptable transmission delay. The performance of the algorithm was confirmed by analytical examinations. The result shows that the quality of MPEG streams is increasing significantly compared with the currently used methods.**

Keywords: selective retransmission, DCCP, video streaming

I. INTRODUCTION

Today, with the rise of multimedia and network technologies, multimedia has become an indispensable feature on the Internet. Animation, voice and video clips become more and more popular on the Internet. Multimedia networking products like Internet telephony, Internet TV, video conferencing have appeared on the market. The obstacles of the expansion are the high bit error ratio of the radio link and the limited bandwidth of the mobile links. New technologies like UMTS [1], WiMAX [2] with high bandwidth capability should facilitate mobile multimedia services to come into general use.

The convergence of the different services is more and more appreciable, because not only files are forwarded over IP networks, but voice and video too. The requirements of various services are different. While file transfer needs a totally reliable transmission, in case of multimedia applications the delay is the most important parameter. Of course, the goal is to provide the highest quality with the lowest delay possible. However, current IP networks do not provide Quality of Service (QoS), which is an important obstacle to overcome in order to assure a proper delivery of real-time video, with stringent delay, bandwidth and packet loss requirements. Several efforts are being made to achieve better performance and provide QoS like Integrated Services [3] and Differentiated Services [4].

The generally used transport protocols (TCP, UDP) [5,6] were designed for wired networks, hence they do not work properly on unreliable wireless channels. New protocols were developed like the Lightweight User Datagram Protocol (UDP-Lite) [7] and the Datagram Congestion Control Protocol (DCCP) [8,9]. These protocols seem advantageous for streaming applications in IP based mobile networks due to the partial checksum method. Some real-time applications encode audio/video in a format that handles single-bit errors in the data payload better than the loss of a full packet.

DCCP is an unreliable transport protocol like UDP, UDPLite, but DCCP provides congestion controlled flow of unreliable datagrams. The congestion control mechanism needs to know if packet is lost, hence the DCCP header includes a sequence number field that identifies the packet. Our algorithm will use this feature to retransmit some special packets if they are damaged. We use the general DCCP for our selective retransmission algorithm, but there are other special protocols to provide retransmission [10], [11]. Another method is available to provide increased quality of MPEG streams using DCCP, but this solution realizes selective discard [12] of packets.

We focused on MPEG [13] format where the errors in the key-frame proceed to the other frames, so it is advisable to handle these frames on a different way. Error in a key-frame propagates to all other frames till the next key-frame. Retransmission of the damaged part of the key-frame will significantly raise the quality of the video stream.

The paper is organized as follows. First we present the DCCP transport protocol compared to other protocols. A brief description of MPEG media format is presented in section 3. Our selective retransmission algorithm is introduced in section 4. Performance evaluation of our method follows in Section 5. The conclusion and the plans of our future work are reviewed in the last section.

II. DATAGRAM CONGESTION CONTROL PROTOCOL

The Datagram Congestion Control Protocol is a newly defined transport protocol by the IETF that implements bidirectional, unicast connections of congestion controlled, unreliable datagrams. Presently the reliable TCP is the only alternative protocol to provide congestion control, but the retransmission mechanism is a strong disadvantage for multimedia services due to high end-to-end delay it causes. In delay-sensitive applications, such as streaming media and telephony, timeliness is also preferred. The main goal is to keep the delay and its variance on a minimum level, but on the other hand providing high service quality wit low bit error ratio.

Other differences between DCCP and TCP are the different acknowledgement formats and distinguished kinds of loss. A Data Dropped option declares that a

packet was dropped because of corruption, because of receive buffer overflow, and so on. This facilitates research into more appropriate rate-control responses for these non-network congestion losses (although currently such losses will cause a congestion response).

For real-time applications the time constraints are more important than reliability, so media transmissions typically use transport protocols like UDP, where no retransmission occurs, providing minimal packet delay. UDP avoids long delays, but applications that use UDP as transport protocol, must implement congestion control on their own to prevent packet network flooding with a miss-behaved application and ensure fairness in bandwidth share.

DCCP combines the best features of the two 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.

DCCP is a connection-oriented protocol with special packet types to establish, close and maintain connections. The main packet type for transmitting data is the DCCP-Data. In general use only the 12 or 16 byte length Generic Header is mandatory (Figure 1.).

Source port			Dest port
Data offset	CCVal	CsCov	Checksum
Res	Type	X	Sequence Number (low bits)

Figure 1. DCCP Generic Header

The size of the header depends on the length of the Sequence Number field. The X bit indicates whether 24 or 48 bit long sequence numbers are used. Unlike TCP sequence numbers, which are byte-based, DCCP sequence numbers increment by one per packet. In our selective retransmission algorithm we will utilize this feature.

DCCP similarly to UDPLite is designed to provide a partial checksum that only covers as much of the user data that the sending application specifies in the DCCP Generic Header. Errors in the rest of the packet are ignored because they are assumed to be acceptable for the destination application. To avoid complexity, the protocol requires that the sensitive data in a packet start at the beginning. The CsCov field specifies how many bytes are sensitive to errors. This feature will be also utilized in our algorithm.

DCCP connections are congestion controlled, but unlike in TCP, DCCP applications have a choice of congestion control mechanism. DCCP uses Congestion Control Identifiers (CCID) to determine the congestion control mechanism. Currently two identifiers are being defined: CCID 2 that implements a TCP-like Congestion Control [14] and CCID 3 that implements a TCP-Friendly Rate Control (TFRC) [15], but DCCP is easily extensible to further forms of unicast congestion control.

III. MPEG OVERVIEW

MPEG is an encoding and compression system for digital multimedia content defined by the Motion Pictures Expert Group (MPEG). MPEG-2 extends the basic MPEG system to provide compression support for TV quality transmission of digital video. The MPEG-2 video compression algorithm achieves very high rates of compression by exploiting the redundancy in video information.

MPEG video streams consist of a sequence of sets of frames known as a GoP (Group of Pictures). The three types of frames in MPEG are: I-frame, P-frame and B-frame. In I-frame there is no reliance on previous or future frames for coding. The coding is done only on the frame itself and so there is only intra-frame coding used. The P-frame is coded using prediction of the previous I-frame or P-frame. This prediction is usually accomplished by means of motion compensation and motion predication. The last type of frames is the B-frame that is coded by means of forward and backward prediction to I-frames and P-frames or even an estimate of a value between of them.

Figure 2. GoP structure

I-frames are much more important than the other ones. An error in the key-frame propagates toward P-, and B-frames, because these frames are derived from I-frames. It would be very advantageous to protect the key-frames.

IV. SELECTIVE RETRANSMISSION ALGORITHM

Each DCCP packet has an individual sequence number that makes possible to detect packet losses. Packet loss occurs when the link is congested or a bit is corrupted. On wireless links bit corruption plays significant role. DCCP does not specify the retransmission of missing packets, but in some cases it would be advantageous.

The main idea is to differentiate the application data because not all type of data has the same importance. As introduced in the previous section MPEG's key-frames are more important then the other frames so these frames should be handled on a different way than other frames. The P- and B-frames are predicted from the I-frame so errors in the key-frame have effect on all frames in the GoP. If the damaged part of the I-frame is retransmitted the quality of the streamed video should increase considerably because the MPEG's prediction method will be based on correct I-frame. The selective retransmission can be solved by DCCP easily due to sequence numbering and the partial checksum. The other currently used transport protocol can not be used for this purpose. The UDP does not have sequence numbers so the identification of packets is not possible. The other disadvantage of the UDP is that the checksum covers the entire packet, hence any error in the packet will cause packet drop which is not advantageous in MPEG multimedia streaming. The UDPLite solves the second problem by partial checksum method but the identification of packets is also impossible. TCP is not capable for multimedia streaming due to high delays what is not acceptable for streaming purposes. The model of the

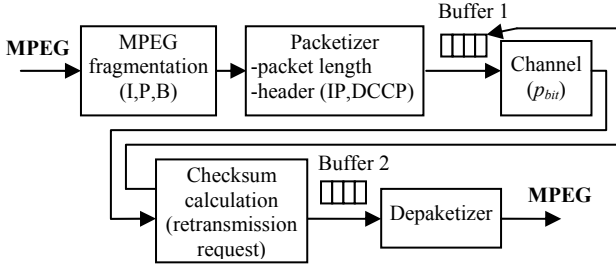selective retransmission algorithm is introduced in Figure 3.

Figure 3. Selective retransmission model

The sender application fragments the MPEG stream and creates packets that will be transmitted over the noisy channel with bit-error probability p_{bit}. The packetizer module determines the packet length and adds the headers (IP, DCCP). Every I-frame must begin in a new packet because the sensitive area always follows the header.

The packet that contains I-frame data should be buffered in *Buffer1* (Fig. 3.) and deleted only when the acknowledgement arrives generated automatically by congestion control algorithm, thus the sender is informed about the damaged packets.

The receiver recognizes bit errors due to recalculation of the checksum field and stores the datagrams in *Buffer2*. The depaketizer module must wait till the retransmitted packets arrive.

In our selective retransmission algorithm only the I-frames and the packet headers are covered by checksum using the partial checksum method supported by DCCP. The other parts of the packet (P-, B-frame data) will be processed without checking. If the recalculated checksum is not correct, the packet will be retransmitted, but this time only the header will be covered by checksum without retransmission request, hence at retransmission the checksum must be recalculated.

To analyze the delay effects of the retransmission algorithm, it is useful to divide streaming media applications into three subtypes:

One-way pre-recorded media
One-way live media
Two-way interactive media

In case of recorded and one-way live media the selective retransmission is adaptable because the additional delay is not significant. In case of interactive media the RTT must be analyzed to decide whether the total delay is acceptable. In our solution the delay is:

$$t_{delay} = t_{transmission} + t_{retransmission} = \frac{3}{2} RTT \qquad (1)$$

Of course the retransmitted packet should be damaged too, but the correctly arrived ones have significant effect on the quality on the cost of additional delay.

To increase the quality of the MPEG stream, the average packet loss must be reduced. In usual the packet loss probability is

$$p_{packetloss} = 1 - (1 - p_{bit})^{CsCov}, \qquad (2)$$

where *CsCov* is the size of data covered by checksum and p_{bit} is the bit-error probability of the noisy channel. In UDP the *CsCov* parameter is equal to the packet size (*PS*), but in our method it varies. If only P- or B-frames are in the packet the *CsCov* is equal to the header size (*H*), but if the packet contains I-frame data the *CsCoV* field is set to

$$CsCov = H + I , \qquad (3)$$

where *I* stands for the size of the I-frame data in the packet payload. In a MPEG stream the average I-frame size is signed with *size(I)* and the whole GoP's size with *size(GoP)*, so the average I-frame data length in a packet is

$$avg(I) = (PS - H) \cdot \frac{size(I)}{size(GoP)} . \qquad (4)$$

Using (4) the packet drop probability is

$$p_{packetloss} = 1 - (1 - p_{bit})^{\frac{PS-H}{size(GoP)} \cdot size(I) + H} \qquad (5)$$

The packet will be retransmitted only if the packet payload contains I-frame data, hence the retransmission probability is calculated as follows:

$$p_{retransmission} = \frac{size(I)}{size(GoP)} \cdot (1 - (1 - p_{bit})^{\frac{PS-H}{size(GoP)} \cdot size(I) + H}) \qquad (6)$$

From (6) it is unambiguous that the retransmission probability is similar to UDP packet loss probability if only key-frames are used. It is noticeable that no retransmission occurs if there are no I-frames. Figure 4. introduces the retransmission probability in the function of *size(I)/size(GoP)* ratio according to the actual channel bit-error probability.
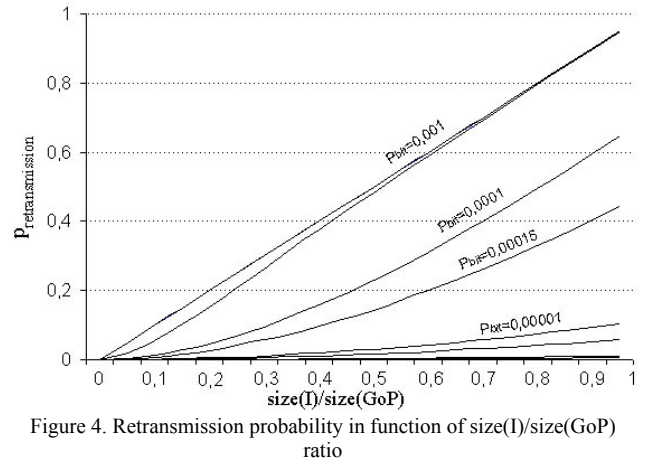

Figure 4. Retransmission probability in function of size(I)/size(GoP) ratio

As the figure shows the retransmission probability does not change significantly when the channel's bit-error ratio is low, hence the performance of our algorithm is higher when the link conditions are bad.

From equation (6) it is observable that the bit-error ratio has impact on the retransmission probability. Figure 5. shows the relation between the two probability.

Using these relations, constant retransmission probability can be achieved, with adaptive MPEG coder. In function of p_{bit} the coder must adapt the I-frame frequency (*N*), which has heavy impact on the quality. With constant retransmission probability the load of a link

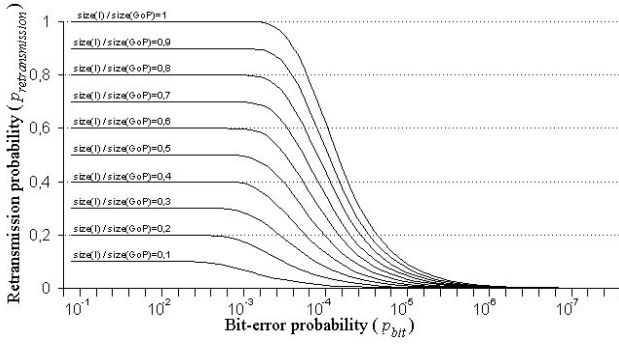can be estimated so the link could be efficiently managed between numerous sources.

Figure 5. Retransmission probability in function of bit-error probability

V. ANALYTICAL EXAMINATION

In this section we will compare the selective retransmission algorithm with conventional methods for multimedia streaming. In general UDP, UDPLite and DCCP transport protocols are used for this purpose. UDPLite and DCCP are very similar except that DCCP provides congestion control, hence calculations on UDPLite are the same for the general DCCP without retransmission.

First we have examined UDP in the aspect of bit-error ratio. While the UDP covers the entire packet by checksum a single bit error will cause packet drop losing all bits in the packet. The probability of losing PS bits is $P(A)=p_{packetloss}$, where PS stands for the packet size. The probability of losing zero bits is $1-p_{packetloss}$. We introduce an indicator probability variable X which denotes the packet loss occurrence. The relative frequency of damaged packets and the expected value of number of damaged packets are equal to the packet loss probability due to the weak law of large numbers.

$$\mathbf{E}(X) = 1 \cdot P(A) + 0 \cdot (1 - P(A)) = P(A) = p_{packetloss} \quad (7)$$

Every packet loss occurrence leads to the loss of PS bit, hence the expected value of the number of damaged bits (#$dbits$) for every packet is given in the following equation:

$$\mathbf{E}(\#dbits)_{UDP} = p_{packetloss} \cdot PS \quad (8)$$

In case of using UDPLite or general DCCP the expected value of damaged bit is lower due to partial checksum method. Loss of PS bits will occur only if the header is corrupted (occurrence B, $P(B)=p_{packetloss}$) otherwise only a single bit will be lost (occurrence C, $P(C)=p_{bit}$), so $B \cap C \neq \{0\}$. The expected value of lost bits is:

$$\mathbf{E}(\#dbits)_{UDPLite} = P(B) \cdot PS + $$
$$+ P(C) \cdot (PS - H) - P(B) \cdot P(C) \cdot (PS - H) \quad (9)$$

Replacing the probability $p_{packetloss}$ the expected value is

$$\mathbf{E}(\#dbits)_{UDPLite} = (1 - (1 - p_{bit})^H) \cdot PS + $$
$$+ p_{bit} \cdot (PS - H) - (1 - (1 - p_{bit})^H) \cdot p_{bit} \cdot (PS - H) \quad (10)$$

Adopting the selective retransmission algorithm calculation of expected value of damaged bits is very similar. It must be taken into consideration that the retransmitted packets should be damaged, but this time only the header corruption leads to packet loss therefore the average number of lost bits is less or equal to the previously calculated value in case of UDPLite

$$\mathbf{E}(\#dbits)_{SelRetrAlg} = \mathbf{E}(\#dbits) + $$
$$+ p_{retransmission} \cdot \mathbf{E}(\#dbits)^2 - p_{retransmission} \cdot \mathbf{E}(\#dbits) \quad (11)$$

In excessive case when the packets do not contain I-frame data the number of damaged bits is similar to the calculated value given in equation (10) because in this case the retransmission probability is zero. It is perceptible that the number of correctly received bits is increasing when the $size(I)/size(GoP)$ ratio is higher. This is due to the retransmission of damaged I-frame data because some of retransmitted packets will be received correctly. In worst case the number of damaged bits will be similar to UDPLite.

The comparison of different methods (UDP, UDPLite and the DCCP Retransmission Algorithm) is illustrated on Figure 6. We examined the probability of bit loss on the receiver side calculated as follows:

$$p_{bitloss} = \frac{\mathbf{E}(\#dbits)}{PS} \quad (12)$$

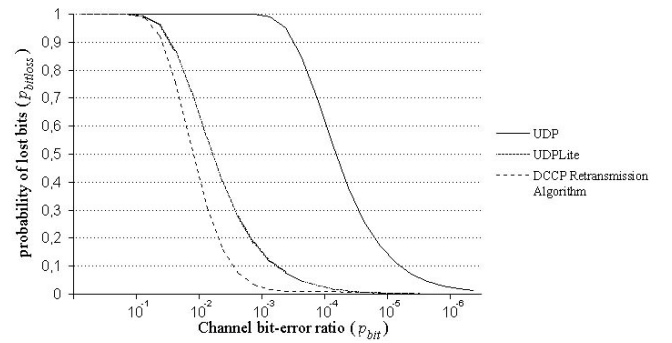In our example half of the transmitted data is I-frame data.

Figure 6. Comparison of UDP, UDPLite and DCCP Retransmission Algorithm

The highest rate of bit corruption happens in case of UDP. A single bit error leads to the drop of the entire packet, while using UDPLite a single bit error leads to loss of one bit if it belongs to the application data. The entire packet will be dropped only if the header is damaged, therefore the probability of bit loss is lower then in UDP. Using our selective retransmission algorithm the bit loss probability is more smaller and only at the case of $size(I)/size(GoP)=1$ is equal to UDP. This probability means the general packet loss probability but it is very important that using our algorithm most of the lost bits belongs to P- and B-frames. These bit errors have no such effect on the quality as the errors in the I-frame.

Correction of a single bit error in the key-frame has impact on the other frames. There are N frames in a GoP, therefore single I-frame bit correction leads to the improvement of N bits increasing the MPEG quality significantly. To calculate the correction effectiveness first the bit correction probability must be introduced. The

probability of a single bit correction in the I-frame is given in equation (14), where p_{bit_error} is the probability of a single bit error in the case of UDPLite

$$p_{correction} = p_{retransmission} \cdot p_{bit\_error}^2 - p_{retransmission} \cdot p_{bit\_error} =$$
$$= p_{retransmission} \cdot p_{bit\_error} (p_{bit\_error} - 1) \quad . \quad (13)$$

Using the last equation we can calculate the expected value of corrected bit in a GoP that contains N frames due to the correction of I-frame bits

$$\mathbf{E}(corr\_bit)_{GoP} = \frac{size(I)}{size(GoP)} \cdot p_{correction} \cdot N \quad . \quad (14)$$

The results show that our selective retransmission algorithm is a very efficient method for increasing the quality of MPEG multimedia streams. The algorithm is especially efficient in wireless networks with high bit-error ratio. Compared with nowadays used UDP our algorithm has equal bit corruption ratio (2%) at p_{bit}=10^{-3} while UDP reaches this minimal ratio at p_{bit}=10^{-6} (see Figure 6.).

VI. CONCLUSIONS

In this paper a new selective retransmission algorithm was presented and studied. The results obtained show that the algorithm radically reduces the bit corruption ratio and achieves significant increase of quality especially in wireless networks with high bit-error ratio. We showed that it is possible to achieve results better than those offered by unreliable protocols such as UDP or UDPLite without sacrificing performance. It is important that the algorithm does not add significant overhead or implementation complexity.

The evaluations were done on MPEG streams, but the selective retransmission algorithm is capable for other data type transmissions where the data can be differentiated. The algorithm can be extended to allow multiple retransmissions realizing TCP-like reliable transmission.

As future plans analytical investigations will be followed by simulation examinations to achieve practical results. The simulation will allow testing the I- and B-frame data retransmissions and not only the I-frame data. An interesting future work is a DCCP testbed development to examine our selective retransmission algorithm in real conditions.

REFERENCES

[1] ETSI UMTS: Universal Mobile Telecommunications System: ES 201 385 V1.1.1 (1999).
[2] The WiMAX Forum. http://www.wimaxforum.org.
[3] R. Braden, D. Clark, and S. Shenker, "Integrated services in the Internet architecture: An overview", RFC 1633, Internet Engineering Task Force, July 1994.
[4] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services", RFC 2475, Internet Engineering Task Force, December 1998.
[5] J. Postel: "Transmission Control Protocol", RFC 792, Internet Engineering Task Force, September 1981.
[6] J. Postel: "User Datagram Protocol", RFC 768, Internet Engineering Task Force, August 1980.
[7] Larzon, Degermark, Pink: "The Lightweight User Datagram Protocol (UDP-Lite)", RFC3828, Internet Engineering Task Force, July 2004.
[8] Kohler, Handley, Floyd: "Datagram Congestion Control Protocol (DCCP)", draft-ietf-dccp-spec-11.txt, March 2005.
[9] T. Phelan : "Datagram Congestion Control Protocol (DCCP) User Guide", draft-ietf-dccp-user-guide-03.txt, October 2005
[10] M. Piecuch, K. French, G. Oprica, M. Claypool: "A Selective Retransmission Protocol for Multimedia on the Internet", Proceedings of SPIE International Symposium on Multimedia Systems and Applications, Boston, MA, USA, November 5-8, 2000.
[11] E. Mulabegovic, D. Schonfeld, R. Ansari: "Lightweight Streaming Protocol (LSP)", Proceedings of the tenth ACM international conference on Multimedia, Juan-les-Pins, France, December 2002.
[12] R. N. Vaz, M. S. Nunes, "Selective Frame Discard for Video Streaming over IP Networks", Proceedings of the 7th Conference on Computer Networks (CRC2004), Leiria, Portugal, October 2004
[13] I. I. S. 13818. "Generic Coding Of Moving Pictures And Associated Audio Information", 1994.
[14] Sally Floyd: "Profile for DCCP Congestion Control ID 2: TCP-like Congestion Control", draft-ietf-dccp-ccid2-06.txt, 18 July 2004.
[15] Sally Floyd: "Profile for DCCP Congestion Control ID 3: TFRC Congestion Control", draft-ietf-dccp-ccid3-06.txt, 18 July 2004.
[16] D. Chiu and R. Jain: "Analysis of the increase/decrease algorithms for congestion avoidance in computer networks", Computer Networks and ISDN, June 1989.