Proceedings of the 5th WSEAS Int. Conf. on APPLIED INFORMATICS and COMMUNICATIONS, Malta, September 15-17, 2005 (pp404-410)

Perceptually Enhanced ARQ for Live Video Streaming over Congested 802.11e Networks

P. BUCCIOL[1], E. MASALA[1], E. FILIPPI[3], J. C. DE MARTIN[2]
[1]Dipartimento di Automatica e Informatica / [2]IEIIT-CNR
Politecnico di Torino, Corso Duca degli Abruzzi, 24 — I-10129 Torino
[3]Advanced System Technologies, STMicroelectronics S.r.l. — Cornaredo (Milano)
ITALY

Abstract: - In this paper we present a perceptual ARQ algorithm specifically designed for real-time 802.11 video communications. A fundamental characteristic of the algorithm is its ability to take into account the perceptual and temporal importance of each packet at the same time. A priority value is associated to each packet to determine which one to retransmit at each retransmission opportunity. Compared to the standard 802.11 MAC-layer ARQ scheme, the proposed technique delivers higher perceptual quality because only the most perceptually important packets are retransmitted. Simulations of H.264 live video streaming in a realistic 802.11e infrastructured scenario show that the proposed method consistently outperforms the standard link-layer 802.11 retransmission scheme, delivering gains up to 10 dB of PSNR as well as very low transmission delays.

Key-Words: - Perceptual ARQ, H.264 live video streaming, 802.11e wireless LAN's

1 Introduction

The number of devices equipped with an 802.11 wireless network interface [1] is dramatically increasing. Consumer electronics devices, especially the ones with multimedia capabilities, are expected to integrate such interfaces in a very short time. Hence it is very important for the 802.11 standard to efficiently transport different kinds of traffic, and multimedia in particular. In 802.11, radio link noise and MAC-level collisions are addressed by an automatic link-layer retransmission scheme. While data-agnostic, link-layer ARQ is both fast and simple to implement, for the specific —and increasingly important— case of multimedia traffic, more advanced ARQ techniques could use network resources more efficiently as well as deliver higher perceptual quality.

Two of the most important characteristics of multimedia streams are their highly non-uniform perceptual importance and their strong time sensitivity. One or both characteristics are usually considered by most ARQ techniques designed for multimedia communications. The *Soft ARQ* proposal [2], for instance, avoids retransmitting late data that would not be useful at the decoder, thus saving bandwidth. Variants of the Soft ARQ technique have been developed for layered coding [2].

Compressed multimedia bitstreams are composed of syntax elements of varying perceptual importance. Some techniques exploit this feature by assigning different priorities to the syntax elements. In [3] video packets are protected by error correcting codes whose amount depends on the kind of frame to which the video packets belong. Channel adaptation is achieved by an additional ARQ scheme that privileges the most important classes of data. Scheduling of video frames according to the priority given by their position inside the Group of Pictures (GOP) in presented in [4], coupled with the assignment of different priorities to the various kinds of data (i.e. motion and texture information) contained in each packet.

Further improvements are possible optimizing the transmission policy for each single packet, rather than relying on a priori determination of the average importance of the elements of the compressed bitstream [5]. For instance, packets could be retransmitted or not depending on whether the distortion caused by their loss is above a given threshold, as in the low-delay wireless video transmission system presented in [6]. However, it is not clear how to optimally determine such threshold. Given a way to associate distortion values to each packet, rate-distortion optimization of the transmission policies has also been proposed [7] [8].

In this paper, we focus on the specific case of real-time video transmission over 802.11e networks. Unlike the 802.11 MAC-level ARQ which retransmits all packets regardless of their importance, we propose a perceptual ARQ scheme, implemented at the application level, which exploits information about the *perceptual* and the *temporal* importance of each packet. In our proposal, a set of retransmission opportunities is determined at the beginning

Proceedings of the 5th WSEAS Int. Conf. on APPLIED INFORMATICS and COMMUNICATIONS, Malta, September 15-17, 2005 (pp404-410)

of each GOP, then the algorithm retransmits unacknowledged packets according to their priority. Each packet's priority is computed using a simple and flexible formula, that combines perceptual importance and maximum delay constraint. Perceptual importance is evaluated using the analysis-by-synthesis technique [8], which is detailed in Section 3.

In this paper we extend our previous work [9], simulating a congested home network scenario based on the 802.11e standard, in which the access point represents the home access gateway. We simulate the transmission of an H.264 video sequence from the access point to a PC, in presence of several concurrent interfering flows, which lead to very congested network conditions. Detailed results are presented, in terms of both perceptual (measured by PSNR) and network performance. The results show that consistent gains are achieved by the proposed scheme with respect to the standard 802.11 retransmission technique. Moreover, they show that the proposed perceptual ARQ technique presents a very low transmission delay even in very congested network conditions.

This paper is organized as follows. Section 2 and Section 3 review the H.264 standard and analysis-by-synthesis distortion estimation, respectively. In Section 4 the proposed perceptual ARQ technique is presented in detail. Section 5 presents the simulation setup, followed by the discussion of the results in Section 6. Conclusions are drawn in Section 7.

2 H.264 Video Transmission

We focus on the transmission of video data compressed according to the new ITU-T H.264 standard. In the H.264 Video Coding Layer (VCL), consecutive macroblocks are grouped into *slices*, that are the smallest independently decodable units. They are useful to subdivide the coded bitstream into independent packets, so that the loss of a packet does not affect the ability of the receiver to decode the others. To transmit the video data over an IP network, the H.264 provides a Network Adaptation Layer (NAL) [10] for the Real-Time Transport Protocol (RTP), which is well suited for real-time wired and wireless multimedia transmissions.

Some dependencies exist between the VCL and the NAL. The packetization process is an example. Error resilience, in fact, is improved if the VCL is instructed to create slices of about the same size of the packets and the NAL told to put only one slice per packet, thus creating independently decodable packets. Note that in H.264 the subdivision of a frame into slices can vary for each frame of the sequence. However slices cannot be too short due to the resulting overhead that would reduce coding efficiency.

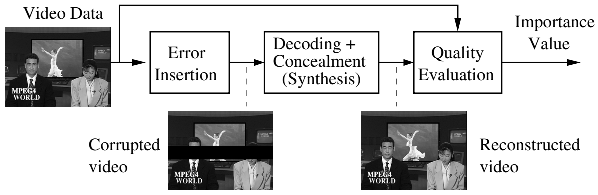

Fig. 1. Block diagram of the analysis-by-synthesis technique.

3 Analysis-by-Synthesis Distortion Estimation

Multimedia data, and video in particular, exhibit non-uniform perceptual importance. When video is transmitted over a noisy channel, each loss event causes a decrease of the video quality that depends on the perceptual importance of the lost data. Such importance can be defined *a priori*, based on the average importance of the elements of the compressed bitstream, as with the data partitioning approach.

At a finer level of granularity, the importance of a video coding element, such as a macroblock or a packet, could be considered proportional to the distortion that would be introduced at the decoder by the loss of that specific element. The distortion estimate associated to each packet could, therefore, be computed as follows:

1. decoding (including concealment) of the bitstream simulating the loss of the packet being analyzed (synthesis stage);
2. computation of the distortion (e.g. MSE) between reconstructed and original sequence;
3. storage of the obtained value as an indication of the perceptual importance of the analyzed video packet.

Figure 1 shows the block diagram of the above described analysis-by-synthesis approach.

The analysis-by-synthesis distortion estimation scheme is independent of the video coding standard. Since it includes the synthesis stage in its body, it can accurately evaluate the effect of both the error propagation and the error concealment. Note that this method assumes isolated packet losses; nevertheless, this leads to a useful approximation as demonstrated by some applications of the analysis-by-synthesis approach to MPEG coded video [5] [8].

The complexity and delay of the analysis-by-synthesis classification technique depend on the frame types the sequence is composed of. If only I-type frames are present, the technique is quite simple since each frame is coded independently of the others. If the sequence contains also predicted frames such as in the case of H.264, the algorithm is more complex because error propagation must be taken into account until the end of the GOP; a model-based approach, however, can be used to drastically reduce complexity [11]. Moreover, note that in the case of stored video (e.g. non-live streaming scenarios), the distortion values can be precomputed and stored.

4 Cross-Layer Perceptual ARQ

To take into account the perceptual and temporal importance of each multimedia packet, an application-level, end-to-end ARQ technique using the IP-UDP-RTP/RTCP protocol stack is proposed. Every packet is transmitted once, then it is stored in a retransmission buffer RTX$_{buf}$ waiting for its acknowledgment. The receiver periodically generates RTCP receiver reports (RR) containing an ACK or a NACK for each transmitted packet. A NACK is generated when the receiver detects a missing packet by means of the RTP sequence number. Packets in the retransmission buffer are sent in the order given by their combined temporal-perceptual priority, as defined in Section 4.2. The performance of the proposed technique depends on a few key parameters, such as the maximum amount of bandwidth B_{max} granted to retransmissions and the relative weights given to temporal and perceptual importance.

4.1 The Retransmission Scheduling Algorithm

At the beginning of each GOP, the transmission time of each packet produced by the encoder is determined by equispacing the packets of each frame inside their respective frame interval. Let B_{GOP} be the bandwidth needed to transmit the current GOP and B_{max} the maximum amount of bandwidth granted to retransmissions. N_{rtx} retransmission opportunities are available for the current GOP, where $N_{rtx} = (B_{max} - B_{GOP})/\overline{S}_{pck}$ and $\overline{S}_{pck}$ is the average packet size. The time instants corresponding to the retransmission opportunities are determined as follows. The total size of each frame is first computed and then the smallest one is identified. The time instant of the first retransmission opportunity is set to be midway between the time instant of the first packet of the smallest frame interval and the last packet of the previous frame. The procedure is repeated until N_{rtx} opportunities have been determined, considering at each step the opportunities filled by packets of size $\overline{S}_{pck}$. This procedure may create retransmission bursts between each frame, but has the advantage to be simple to implement; if desired, a more uniform distribution of the retransmission opportunities is achievable. Note also that the opportunities will not be necessarily completely used.

The algorithm used by the sender to implement the retransmission policy is based on a retransmission buffer RTX$_{buf}$. When a packet is sent, it is placed in the RTX$_{buf}$, waiting for its acknowledgment, and marked as *unavailable* for retransmission. When an ACK is received, the corresponding packet in the RTX$_{buf}$ is discarded because it has been successfully transmitted. If a NACK is received, the packet is marked as *available* for retransmission. Packets belonging to the RTX$_{buf}$ that will never arrive at the decoder in time for playback are discarded. To limit the impact of receiver report losses, the sender piggybacks the highest sequence number for which it received an ACK or NACK. The receiver always repeats in the receiver reports the status information for all the packets whose sequence number is less than the piggybacked one.

When a retransmission opportunity approaches, a priority function (see Section 4.2) is computed for each packet marked as *available* in the RTX$_{buf}$ and the one with the highest priority is transmitted. It is important to stress that the retransmission opportunities computed according to B_{max} *not necessarily* will be actually used by the algorithm, leading to an actual bandwidth usage which can be considerably lower than B_{max}.

4.2 The Priority Function

In a real-time streaming scenario each packet must be available at the decoder a certain amount of time before it is played back to allow the decoder to process it. Let t_n be the time the n-th frame is played back. All packets containing data needed to synthesize the n-th frame must be available at the decoder at time $t_n - T_P$ where T_P is the decoder processing time. Note that the temporal dependencies present in the coded video (e.g. due to B-type frames) must also be taken into account.

For each packet i belonging to the n-th frame we define its deadline (i.e. the time instant by which the packet must reach the decoder) as $t_{i,n} = t_n - T_P$. If a packet never arrives, or arrives after $t_{i,n}$, it produces a distortion increase $D_{i,n}$ that can be evaluated using the analysis-by-synthesis technique. The sender should always select a packet for transmission only among the ones that can arrive before their deadline, i.e. $t_{i,n} > t_s + FTT$, where t_s is the instant of the next retransmission opportunity and FTT (Forward Trip Time) is the time needed to transmit the packet, which is typically time-varying, due to the network state. Defining the distance from the deadline as $\Delta t_{i,n} = t_{i,n} - t_s$, the previous condition can be rewritten as $\Delta t_{i,n} > FTT$.

At any given time a number of packets satisfy the condition $\Delta t_{i,n} > FTT$. A policy is needed to choose which packet must be retransmitted and in which order. Consider the packets containing the video data of a certain frame: each packet has the same $\Delta t_{i,n}$. Within a frame the sender should transmit, or retransmit, the packet with the highest $D_{i,n}$ that has not been yet successfully received. The decision is not as clear when choosing between sending an element A with low distortion $D_{A,n-1}$ in an older frame and an element B with high distortion $D_{B,n}$ in a newer frame. In other words, there is a trade-off between the importance of the video data and its distance from the deadline (which

Table 1. Characteristics of the concurrent streams.

Stream	Access Category	Bandwidth
Tested H.264	AC1	765–1304 kbit/s
Video1	AC2	1.5 Mbit/s
Video2	AC2	3 Mbit/s
Video3	AC2	6 Mbit/s
FTP	AC0	variable
VoIP	AC3	70 kbit/s
RTP RR	AC3	3–6 kbit/s

can be seen as a sort of temporal importance.) A reason in favor of sending A is because its playback time is nearer ($\Delta t_{A,n-1} < \Delta t_{B,n}$), that reduces the number of opportunities to send it. On the other hand, if B arrives at the decoder, it will reduce the potential distortion of a value greater than A (because $D_{B,n} > D_{A,n-1}$.) A detailed study of the problem can be found in [2].

A retransmission policy is needed to select at each retransmission opportunity the video packet that optimizes a given performance criterion. We propose to compute, for each packet, a priority function of both its potential distortion and its distance from the deadline:

$$V_{i,n} = f(D_{i,n}, \Delta t_{i,n}). \qquad (1)$$

The retransmission policy consists of sending packets in decreasing order of priority $V_{i,n}$. The issue is to find an effective, and, if possible, simple, function that combines the distortion value with the distance from the deadline. We propose to use the following function:

$$V_{i,n} = D_{i,n} + wK\frac{1}{\Delta t_{i,n}}, \qquad (2)$$

where K is a normalization factor, computed as the product of the mean value of the distortion and the receiver buffer length T_B in seconds as in the following formula

$$K = \overline{D_{i,n}} \cdot T_B. \qquad (3)$$

The normalization factor, K, is designed to balance the perceptual and temporal importance of the packet for the average case. The size of the receiver buffer T_B is, in fact, approximately equal to the mean value of the distance from the deadline, assuming that the receiver buffer is almost full. The weighting factor w in Eq. (2) is introduced to control the relative importance of the perceptual and temporal terms of the formula.

5 Simulation Setup

The proposed technique has been implemented and tested using *ns*. The simulator implements an 802.11e MAC layer over an 802.11a physical layer with a channel bandwidth of 36 Mbit/s. A packet error model has been implemented

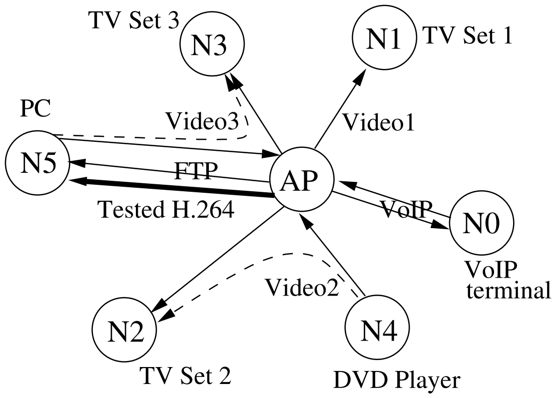

Fig. 2. The 802.11 network topology. The tested H.264 video stream is transmitted from the Access Point to the destination node. The solid lines show the actual path of the transmitted packets, while the dashed lines indicate logical connections.

in *ns* based on BER curves obtained from 802.11 channel measurements, with different noise levels and packet sizes.

A realistic home network scenario, shown in Figure 2, has been simulated. Many wireless devices (three TV sets, a DVD player, a PC and a VoIP terminal) are connected to the same Access Point. Three concurrent video transmission, a VoIP call and an FTP transfer are active at the same time, as well as the H.264 live video transmission under test. The H.264 transmission is originated from the access point, that represents the home access gateway; packets are directly sent to the destination (PC) without using intermediate hops.

Three standard CIF test sequences have been encoded using version 6.1e of the H.264 test model software with a fixed quantization parameter. The GOP encoding scheme is IBBPBBPBBPBB. The characteristics of the tested video sequences are shown in Table 2. Each sequence is concatenated with itself to reach a length of approximately 500 s. The video encoder is instructed to make RTP packets whose size is approximately constant. The decoder implements a simple temporal concealment technique that replaces a corrupted or missing macroblock with the macroblock in the same position in the previous frame.

Traffic has been assigned to the 802.11e Access Categories as shown in Table 1. This assignment follows the Wi-Fi alliance recommendation for multimedia [12]. The FTP stream is assigned to the lowest priority class (Access Category 0). The tested H.264 stream is assigned to AC1, while all the remaining video flows are sent as AC2. The VoIP flows and the receiver reports are assigned to AC3 — which provides the highest available QoS — to achieve the maximum protection against receiver report losses. The maximum number of MAC retransmissions is three for all the classes except AC1, for which no MAC level retransmissions are used. We assigned the tested H.264 video stream and the other video flows to different access cate-

Table 2. Characteristics of the sequences used in the simulations.

Sequence	Avg. bitrate (kbit/s)	Encoding distortion (dB)	Resolution	Frame rate
paris	765	35.68	CIF (352×288)	30 fps
tempete	1205	34.23	CIF (352×288)	30 fps
bus	1304	34.25	CIF (352×288)	30 fps

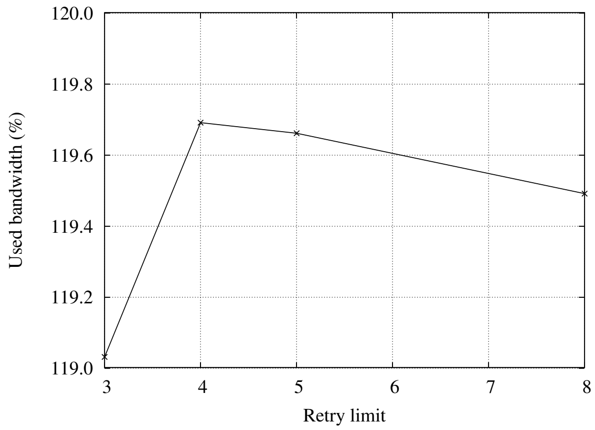

Fig. 3. Used bandwidth as a function of the retry limit for the MAC-level ARQ case; *paris* sequence.

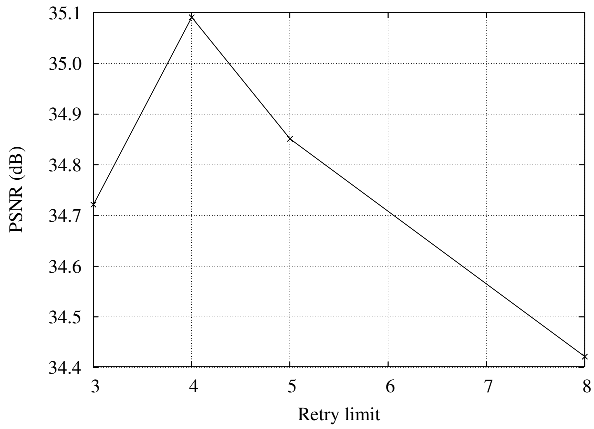

Fig. 4. PSNR as a function of the retry limit for the MAC-level ARQ case; *paris* sequence.

gories because the retry limit can be specified only for each access category and not for each flow. To ensure fairness in the comparisons, however, the tested H.264 stream flow has been assigned to an access category whose priority is lower than the other video streams. Table 1 also reports the bandwidth of all the flows. Note that the rate of the RTCP flow due to the receiver reports is very modest. It ranges from 3 to 6 kbit/s for a 100 ms receiver report interval, and, if needed, could be further improved by packing ACK and NACK information more efficiently than the current implementation.

6 Results

The first set of results shows the performance of the standard MAC-level ARQ technique, as it is implemented in the current 802.11 standard. In the remaining part of the section the best performance of the MAC-level ARQ will be compared to the one of the proposed perceptual ARQ.

In the MAC-level ARQ simulations, we varied the retry limit of the AC1 class to assess its impact on the performance. Figures 3 and 4 assess the performance of the MAC-level ARQ scheme as a function of the retry limit, in terms of used transmission bandwidth and PSNR performance, for the *paris* sequence. The first graph clearly shows that the used bandwidth saturates if the retry limit is increased over a certain threshold, that is four in our simulations. In this condition the used bandwidth is about 120% of the average bitrate of the sequence, i.e. 20% is used for retransmissions. The PSNR presents a maximum when the retry limit is equal to four. For higher values the PSNR performance decreases due to the higher packet delay caused by severe network congestion. The higher packet delay, in fact, results in the expiration of the MAC-level timeout of many packets.

The standard MAC-level ARQ technique is now compared with the proposed perceptual ARQ technique, analyzing its PSNR performance, the used bandwidth and the average packet delay. The impact of the two main parameters of the proposed ARQ algorithm, namely the weighting parameter (w) and the maximum available bandwidth (B_{max}), will also be examined in the following.

Figure 5 shows the PSNR performance of the proposed ARQ technique as a function of the maximum available bandwidth parameter, which is expressed as a percentage of the sequence average bitrate. In this graph, the horizontal solid line represents the best performance achieved by the MAC-level ARQ technique in our simulations. Consistent performance gains with respect to the standard MAC-level ARQ technique are achieved. The performance increase is about 0.5 dB for the case of the *paris* sequence. For other sequences, such as *bus*, the gain reaches up to 10 dB due to the very poor performance of the MAC-level ARQ scheme, caused by the congestion level of the network. For the *paris* sequence, the PSNR performance nearly reaches the encoding distortion, reported in Table 2.

The performance of the perceptual ARQ algorithm has

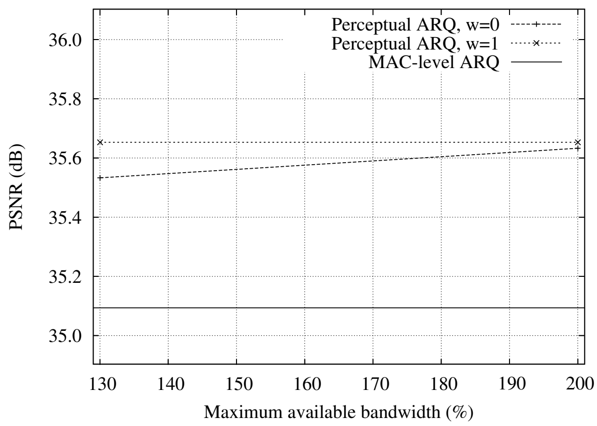

Fig. 5. PSNR as a function of the maximum available bandwidth for the proposed ARQ scheme; *paris* sequence. The horizontal line represents the best performance achieved by the MAC-level ARQ.

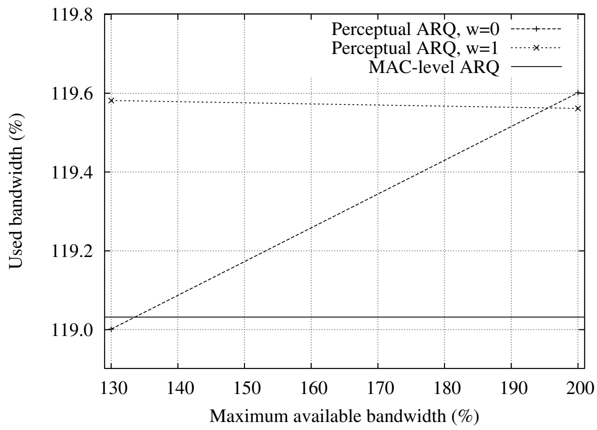

Fig. 6. Used bandwidth as a function of the maximum available bandwidth for the proposed ARQ scheme; *paris* sequence. The horizontal line represents the performance achieved by the MAC-level ARQ.

been reported for two different values of the weighting parameter (w), which determines the relative importance of the perceptual and temporal terms in Equation (2). As shown in Figure 5, when the w value is zero, the temporal constraints of the packets are neglected, hence the retransmission priority of the packets is only based on the perceptual importance. This leads to a lower performance when the maximum available bandwidth is low (about 130% of the sequence average bitrate), because the limited amount of instantaneous available bandwidth for retransmission requires to take into account also the temporal importance of the packets. On the contrary, when the maximum available bandwidth is enough to absorb nearly all instantaneous bandwidth peaks due to retransmissions, the influence of the weighting parameter is more limited.

Note that the average bandwidth used by the algorithm is much lower than the one indicated by the maximum available bandwidth parameter (B_{max}). This value, is, in fact, the *peak* transmission bandwidth, fully used only when a GOP is particularly difficult to transmit. Therefore, the PSNR gain comes from the peak bandwidth increase that allows the algorithm to timely retransmit a higher number of packets when it is more needed. Figure 6 shows the average value of the used bandwidth, expressed as a percentage of the sequence average bitrate. Both the MAC-level ARQ values and the proposed perceptual ARQ values are reported. The perceptual ARQ algorithm presents a slightly higher (1%) bandwidth usage, but its quality performance is consistently better as shown in the previous graph.

The last part of the results analyzes the average delay experienced by the video packets with both the MAC-level ARQ and the proposed perceptual ARQ techniques. For the case of the MAC-level ARQ scheme, the *paris* sequence presents a small average delay (about 150 ms), while *tempete* and *bus* shows an average delay of 0.85 and 1.3 s respectively. Such a high delay might be annoying in some situations, and certainly unsuitable for scenarios with very strict delay requirements. On the contrary, the proposed perceptual ARQ technique achieves a very low transmission delay for both the *paris* and *tempete* sequences, ranging from 50 to 80 ms. The average delay for the *bus* sequence is slightly higher, about 250 ms, which however greatly improves with respect to the 1.3 s average delay of the MAC-level ARQ technique. Hence the proposed perceptual ARQ algorithm can be very interesting in scenarios with very strict delay requirements.

7 Conclusions

In this paper we proposed and analyzed a perceptual ARQ algorithm to transmit video streams on 802.11 wireless networks. The technique computes a priority function for each packet to determine the best scheduling and transmission instants to retransmit packets. Live video streaming of H.264 sequences has been simulated with *ns* in a realistic high-traffic 802.11e infrastructured scenario. Results showed consistent performance gains (up to 10 dB PSNR) over the standard content-transparent 802.11 MAC-level ARQ scheme, with a very low transmission delay.

Acknowledgments

This work was supported in part by STMicroelectronics and by MIUR, Project FIRB-PRIMO, http://primo.ismb.it.

References:

[1] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *ISO/IEC 8802-11, ANSI/IEEE Std 802.11*, 1999.

[2] M. Podolsky, S. McCanne, and M. Vetterli, "Soft ARQ for layered streaming media," in *Tech. Rep. UCB/CSD-98-1024, University of California, Computer Science Division, Berkeley*, November 1998.

[3] Y. Shan and A. Zakhor, "Cross layer techniques for adaptive video streaming over wireless networks," in *Proc. IEEE Int. Conf. on Multimedia & Expo*, August 2002, vol. 1, pp. 277–280.

[4] S. H. Kang and A. Zakhor, "Packet scheduling algorithm for wireless video streaming," in *Proc. Packet Video Workshop*, Pittsburgh, PA, April 2002.

[5] F. De Vito, L. Farinetti, J.C. De Martin, "Perceptual classification of MPEG video for Differentiated-Services communications," in *Proc. IEEE Int. Conf. on Multimedia & Expo*, Lausanne, Switzerland, August 2002, vol. 1, pp. 141–144.

[6] S. Aramwith, C.-W. Lin, S. Roy, and M.-T. Sun, "Wireless video transport using conditional retransmission and low-delay interleaving," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 558–565, June 2002.

[7] J. Chakareski, P. A. Chou, B. Aazhang, "Computing rate-distortion optimized policies for streaming media to wireless clients," in *Proceedings of Data Compression Conference*, April 2002, pp. 53–62.

[8] E. Masala and J.C. De Martin, "Analysis-by-synthesis distortion computation for rate-distortion optimized multimedia streaming," in *Proc. IEEE Int. Conf. on Multimedia & Expo*, Baltimore, MD, July 2003, vol. 3, pp. 345–348.

[9] P. Bucciol, G. Davini, E. Masala, E. Filippi, and J. C. De Martin, "Cross-layer perceptual ARQ for H.264 video streaming over 802.11 wireless networks," in *Proc. of IEEE GLOBECOM*, Dallas, TX, November-December 2004, vol. 5, pp. 3027–3031.

[10] S. Wenger, "H.264/AVC over IP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 645–656, July 2003.

[11] F. De Vito, D. Quaglia, and J.C. De Martin, "Model-based distortion estimation for perceptual classification of video packets," in *Proc. IEEE Int. Workshop on Multimedia Signal Processing (MMSP)*, Siena, Italy, September 2004, vol. 1, pp. 79–82.

[12] Wi–Fi alliance, "Support for multimedia application with quality of service in Wi–Fi networks," *URL: http://www.wi-fi.org*, September 2004.