Supporting Multimedia Traffic in 802.11e WLANs

C. Casetti, C.-F. Chiasserini, L. Merello, G. Olmo
CERCOM - Dipartimento di Elettronica, Politecnico di Torino
C.so Duca degli Abruzzi 24, 10129 Torino - Italy
Ph.: +39 0115644183, Fax: +39 0115644099, Email: {casetti, chiasserini, luca.merello, olmo}@polito.it

***Abstract*— We study the performance of WLANs with infrastructure, when the IEEE 802.11e EDCA function is employed. We focus on VoIP, video and TCP data traffic, and run experiments with ns-2 to assess the performance of different classifications of the traffic flows. Our aim is to investigate the capacity of a WLAN system in terms of number of traffic connections that can be supported, given the QoS requirements of voice and video flows. We show that, using a proper set of access parameters, we can obtain acceptable performances for VoIP while maintaining a good video playout quality, which has been estimated by means of realistic visual comparisons. As for video traffic, we propose a technique to exploit the synergies between application layer (H.264 codec) and MAC layer, and show that this approach can significantly improve the system performance.**

***Index Terms*— Wireless Quality-of-Service, Multimedia traffic, 802.11 WLANs.**

I. INTRODUCTION

IEEE 802.11 Wireless Local Networks (WLANs) have emerged as one of the most successful wireless technologies and are becoming essential in the wireless communications area. As the demand of integrated multimedia services is increasing, high data rates and an efficient Medium Access Control (MAC) scheme become important criteria for the design of WLAN systems.

A new standard IEEE 802.11e that aims at supporting (Quality of Service) QoS in WLANs is specified in [1]. IEEE 802.11e introduces an enhanced, distributed, contention-based access scheme, called Enhanced Distributed Channel Access (EDCA). EDCA offers the possibility to define four different classes of service at the MAC layer so that multimedia traffic can be supported in addition to data traffic.

Several studies on performance of IEEE 802.11e have recently appeared in the literature; some simulation results can be found in [2], [3], [4], while an analytical model of EDCA is presented in [5].

In this paper we investigate the performance of WLANs with infrastructure when the IEEE 802.11e EDCA function is employed [1]. We consider Voice-over-IP (VoIP), video and TCP data traffic, and run experiments with *ns-2* to assess the performance of different classifications of the traffic flows. Our aim is to investigate the capacity of a WLAN system in terms of number of traffic connections that can be supported, given the QoS constraints of voice and video flows. Then, we focus on video traffic and present a technique that exploits the synergies between application layer and MAC layer to improve the system performance. Some preliminary simulation results are shown, confirming our intuition that a significant performance gain can be obtained by adopting a cross-layer approach.

TABLE I
VOICE-OVER-IP QOS REQUIREMENTS

Quality	Delay (ms)	Jitter (ms)	Losses
good	0-150	0-75	$< 3\%$
medium	150-400	0-125	$< 7\%$
poor	>400	0-225	$> 7\%$

TABLE II
H.264 ENCODER SETTINGS

Intra / Inter frame sequence	IBPBP
Period of Intra frames	1 every 6 s
Reference frames	5
R-D optimization	ON
Entropy coding method	CABAC
Average bit rate	256 Kb/s

II. SYSTEM SCENARIO

We consider a wired-cum-wireless network scenario, where the wireless part is represented by a WLAN with Access Point (AP). All stations within the WLAN communicate with a fixed node, which is located in the wired part of the network.

We model VoIP calls so as to mimic a two-speaker interaction. During a talkspurt, the active source sends data following a real-life trace collected at the output of a g.729A vocoder, with an encoded bit rate of 8 Kb/s and an average rate of 2.8 Kb/s resulting from the introduction of VAD (Voice Activity Detection); the framing time is 20 ms. The quality of the recorded performance can be related to voice applications by cross-checking Table I [11].

As for video traffic, we consider five clips recorded from television: a television news, a football match, an action movie, a cartoon and a talk show (i.e., scenes from low motion to high motion). This choice is justified by observing the video demand for mobile terminals in emerging 3G systems. All clips have the same characteristics: resolution of 352x288 colored pixels, frame rate equal to 25 fps, length of 4 minutes. Video has been compressed using H.264/AVC (v7.6), i.e., the most efficient video coder available; the encoder parameters are presented in Table II, and allow for the best compromise between quality and bit rate.

This work was supported by the Italian Ministry of University and Research through the VICOM and the TWELVE project.

For both voice and video, we assume the usual multimedia protocol stack, i.e., RTP/UDP/IP. In particular, all bits belonging to a coded frame of video are divided by the application level into UDP packets with a maximum size of 1500 bytes. This process is carried out so as to ensure that a packet will not include data taken from two consecutive frames. The packets are then sent to underlying levels following a uniform temporal distribution over a 40 ms time period, which is the frame duration.

Finally, we consider TCP traffic sources exhibiting an *on-off* behavior. More specifically, data sources are modelled so as to represent a client-server interaction, such as Web browsing. All wireless stations on the simulated WLAN are TCP clients, while the fixed node is a TCP server.

III. IEEE 802.11E: A SUMMARY

Like the 802.11 Distributed Coordination Function (DCF), EDCA is based on the CSMA/CA scheme and employs the concept of Inter Frame Space (IFS) as well as the backoff mechanism. However, the following innovations are introduced.

- Various Access Categories (ACs) are defined, each of which corresponds to a different priority level and to a different set of parameters to be used for contending the channel. In particular, an 802.11e station operating under the EDCA function includes up to four MAC queues; each queue corresponds to a different AC and represents a separate instance of the CSMA/CA protocol. A queue employs the following parameters to access the channel: (i) the Arbitration Inter Frame Spacing (AIFS[AC]), similar to the DIFS used in DCF, (ii) the Minimum and the Maximum Contention Window ($CW_{min}[AC]$, $CW_{max}[AC]$), (iii) and the TXOP_Limit[AC][1]. The higher the AC priority is, the smaller the AIFS[AC], $CW_{min}[AC]$ and $CW_{max}[AC]$ are. In the following, we do not exploit the TXOP_Limit[AC] parameter.
- Within every 802.11e station, a scheduler solves *virtual collisions* among the AC queues, i.e., among the various CSMA/CA instances, by always enabling the queue associated with the highest priority to transmit.

IV. EXPERIMENTAL RESULTS

We consider 20 wireless stations transmitting at 11 Mb/s. All stations are involved in both a bidirectional TCP connection and a bidirectional VoIP traffic flow; moreover, we assume a number of video flows varying from 0 to 5.

By using *ns-2*, we simulated the EDCA scheme in presence of three different sets of access parameters, namely, *configurations 1, 2 and 3*, as presented in Table III. *Configuration 1* sets the following priority levels (in decreasing order[2]): (0) downlink VoIP and video, (1) uplink VoIP, (2) TCP traffic. With *configuration 2*, we have: (0) downlink VoIP, (1) uplink VoIP, (2) video, (3) TCP traffic. With *configuration 3*, we have: (0) downlink VoIP, uplink VoIP and video, (1) TCP traffic. Notice that *configuration 1* gives higher priority to downlink VoIP and video with respect to uplink voice, thus compensating for the uplink/downlink unfairness phenomenon observed in [6]. Furthermore, we assume that both the header of the compressed file (176 bits) and the header of every single video frame (on average 10 bytes) are always received correctly.

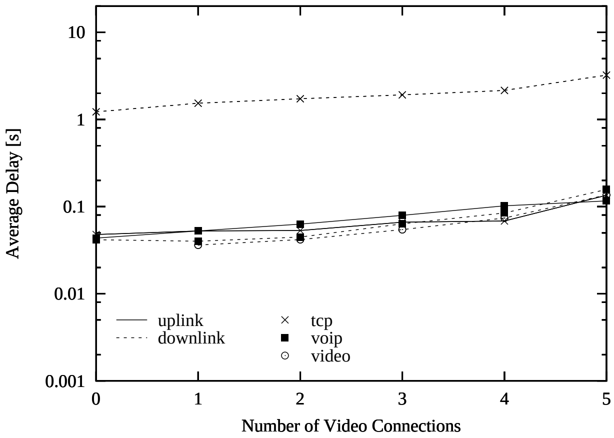

Fig. 1. Average packet delay vs. number of video flows, when *configuration 1* is employed

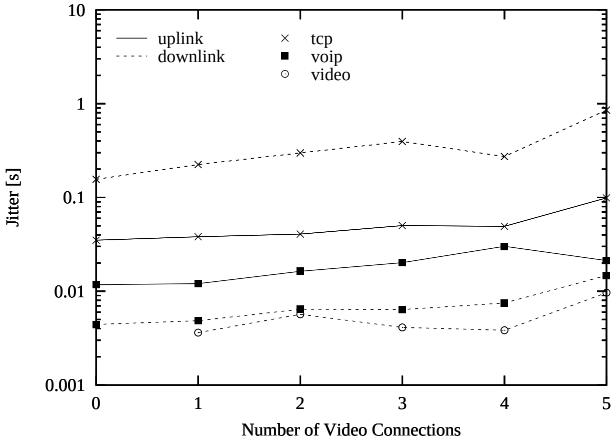

Fig. 2. Delay jitter vs. number of video flows, when *configuration 1* is employed

Our results show that *configuration 1* provides at most four concurrent video streams with acceptable quality, while offering a medium VoIP quality. Indeed, both VoIP and video packets never exceed an average delay of 100 ms or a delay jitter of 10 ms, as shown in Figures 1–2. As for packet loss probability, Figure 3 shows the obtained results as functions of the number of video streams. We highlight that VoIP packet loss always remains below 7%, while video packet losses are always below 1%, thus allowing for a good video playout at the receiver side.

With *configurations 2* and *3*, acceptable performances are achieved only with two video flows or less, otherwise packet loss quickly reaches 10% (results are not shown here for the sake of brevity). Such a high value of packet losses forces the decoder to keep displaying on the screen the last frame

---

[1] The value of AIFS must be at least as long as the DIFS interval; the only exception is for the AP that can use an AIFS 30 μs long.

[2] I.e., (0) corresponds to the highest priority.

TABLE III

CONFIGURATION NUMBER VS. THE EDCA PARAMETER SETTINGS IN TERMS OF AIFS [μs] AND (CW$_{min}$,CW$_{max}$)

Config.	AIFS Downlink			AIFS Uplink		
	TCP	VoIP	Video	TCP	VoIP	Video
1	50	30	30	70	50	-
2	50	50	50	50	50	-
3	50	50	50	50	50	-

Config.	(CW$_{min}$,CW$_{max}$) Downlink			(CW$_{min}$,CW$_{max}$) Uplink		
	TCP	VoIP	Video	TCP	VoIP	Video
1	$(31, 1023)$	$(3, 255)$	$(3, 255)$	$(63, 4095)$	$(7, 511)$	-
2	$(63, 4095)$	$(3, 255)$	$(31, 1023)$	$(63, 4095)$	$(7, 511)$	-
3	$(63, 4095)$	$(3, 255)$	$(3, 255)$	$(63, 4095)$	$(3, 255)$	-

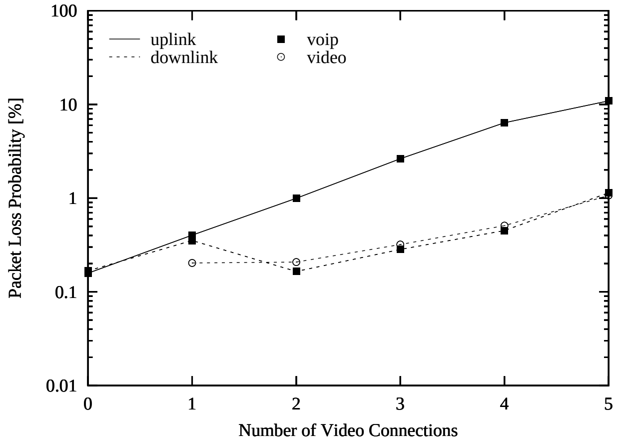

Fig. 3. Packet loss probability vs. number of video flows, when *configuration 1* is employed

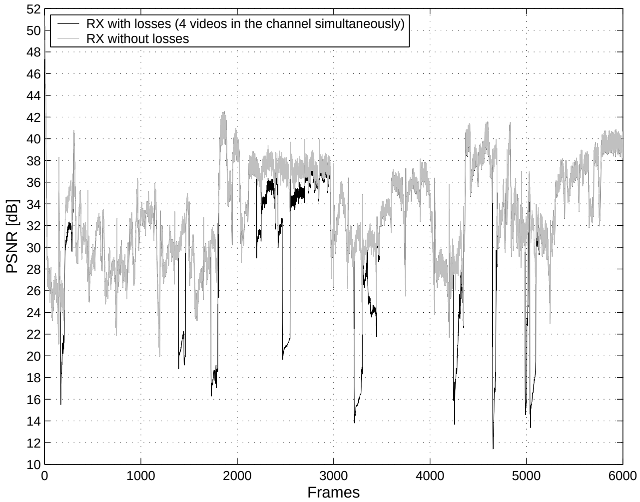

Fig. 4. PSNR Y of video 'television news' vs. frame number, when *configuration 1* is used

received correctly, whose quality is impaired by blocking artifacts caused by corrupted or lost data. The re-establishment of the correct flow of frames occurs slowly, often exhibiting an annoying concurrence of details belonging to the present and of corrupted data related to the past.

Next, we focus on the performance of the video representing television news. Figure 4 presents the PSNR (Peak Signal to Noise Ratio) Y of the original video stream representing television news (gray curve), compared with the stream received at a station (black curve), when the number of video flows is equal to four and *configuration 1 is employed*. The difference between the two curves is due to packet losses during transmission. We observe that the difference becomes more evident as the number of supported video flows increases. Also, from all of the obtained results on PSNR, we can infer that an increase in the number of clips offered by the AP amplifies both frequency and duration of PSNR falls; i.e., both the number of packets losses and the length of loss bursts increase. PSNR reductions may be small (less than 1 dB) or huge (till 22 dB), depending on the importance of the corrupted data. For example, if a packet loss occurs in an area of the bit stream representing high motion, then less time is needed to restore original quality, but the visual impact is very remarkable.

Figure 5 shows, respectively, a frame of the original video stored at the AP and the same frame decoded at the receiver side when *configuration 1* is used and some packet losses occur. We note that, in correspondence of PSNR falls, blocking artifacts may be more or less evident, but, using *configuration 1*, they never compromise the intelligibility of the video message.

Fig. 5. A frame of video 'television news': original frame (left); decoded at the receiver side when *configuration 1* is used and some packet losses occur (right)

V. EXPLOITING INTERACTIONS BETWEEN APPLICATION LAYER AND MAC LAYER

Here we propose a technique to exploit the synergies between application layer (H.264 codec) and MAC layer. Similar synergies between higher and lower layers for the case of multimedia traffic over wired networks have been already explored in [7].

In the following, we briefly introduce the data partitioning error-resilience tool of H.264 and the RTP packetization that is applied, then the proposed cross-layer approach is described.

A. H.264 Features

Like all video codecs that have been recently developed, H.264 includes a number of error-resilience tools, whose application and adaptation is controlled by the encoder [8].

Typically, in H.264 all symbols within a macroblock (MB) are coded together into a single bit string forming a slice. Since some coded information is more important or valuable than other for representing the video content, H.264/AVC (Advanced Video Coding) allows the syntax of each slice to be separated into up to three different partitions, depending on a categorization of syntax elements. In particular, three different partition types are defined.

- *Header information:* This partition is referred to as type A partition. It contains the most important information: MB types, quantization parameters, motion vectors. Without it, symbols of the other partitions cannot be used.
- *Intra Partition:* It is called type B partition. It carries Intra Coded Block Patterns (CBPs) (i.e., indicating how MBs are internally coded) and Intra coefficients (i.e., DCT coefficients). The type B partition requires the availability of the type A partition of a given slice to be useful; also, it can stop possible drift in the video representation and, hence, is more important than the Inter Partition (see below).
- *The Inter Partition:* It is called type C and contains only Inter CBPs and Inter coefficients but is, in many cases, the largest partition of a coded slice. It is the least important because it does not help in re-synchronizing the encoder and the decoder. To be used, it requires the availability of the type A partition, but not the type B partition.

When data partitioning is used, the source coder places symbols of different types into three different buffers. The slice size is adjusted so that, at the transport layer, a partition can always be included in a packet smaller than the Maximum Transport Unit (MTU) size. At the decoder, all partitions need to be available to start standard-compliant reconstruction. However, in the case of H.264 with data partitioning, if the Inter or the Intra partitions are missing, the available header information can still be used to perform error concealment. More specifically, due to the availability of the MB types and the motion vectors, a comparatively high reproduction quality can be achieved, as it is only the texture information that is missing.

B. RTP packetization

The H.264/AVC design includes a Video Coding Layer (VCL), which efficiently encodes the video content, and a Network Abstraction Layer (NAL), which packetizes the VCL representation of the video and provides header information in an appropriate manner for conveyance to a given transport layer or storage media. Thus, all data are included in NAL units (NALU), each of which contains an integer number of bytes. A NALU presents a generic format to be used in both packet-oriented transport and bitstream delivery systems.

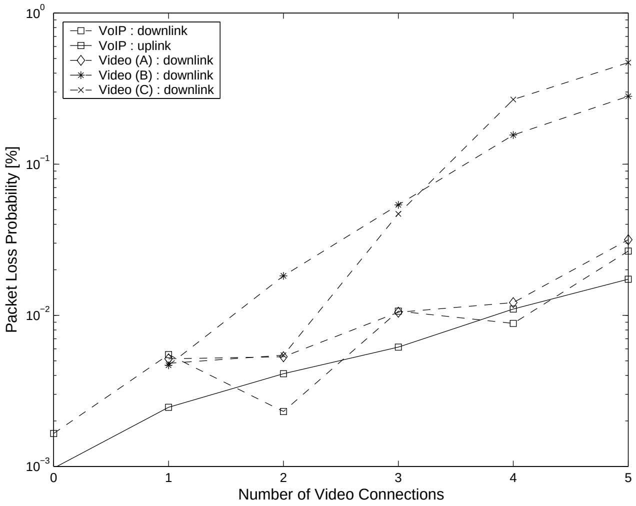

Fig. 6. Packet loss probability vs. number of video flows, when H.264 partitioning and *configuration 1* are employed

In our simulations we encode the video ‘television news’ with the data partition tool of H.264 enabled, and we perform a “simple packetization” scheme. The packetization rules are as follows: (i) Put exactly one NALU (including its header, which co-serves as payload header) into the payload of an RTP packet. (ii) Set the RTP header values as defined in the RTP specification (according to the payload type, sequence number, time-stamp, etc.).

C. Our Proposal

We propose to treat RTP packets differently, depending on their data partition number. We assign to $\{A,B,C\}$ RTP packets three different priority levels at the MAC layer, namely, $\{0,1,2\}$, respectively. This implies that data partitions are protected according to their importance for reconstruction: partition A is protected better than partition B, and partition C is sent as best-effort traffic.

We derived some preliminary results under a scenario similar to the one described in Section IV. In particular, we consider VoIP, video and data traffic, and adopt *configuration 1* (see Table III) so that traffic flows are classified as follows: (0) downlink VoIP and type A video, (1) uplink VoIP and type B video, (2) TCP traffic and type C video. As done before, we assume that the partition header is always correctly received (this can be achieved by repeating the header partition within the RTP packet two or three times).

Figure 6 shows the packet loss probability of VoIP, and video traffic (namely, type A, B and C video partitions), as a function of the number of video flows in the network. VoIP features very good performance with loss probability below 3% (i.e., good quality). Interestingly, loss probability of type A video is significantly lower than in the case of B and C partitions. Note that these results cannot be directly compared with those in Figure 3 since we employed a different packetization technique. While in the previous experiments the average packet size at the transport layer was around 855 bytes, this value drops to 400 bytes when the H.264 error-resilience tool with RTP packetization is applied, leading to a relevant increase in the number of video packets that are generated.

We conclude that combining different access priority at the MAC layer with data partitioning at the application layer is a very efficient way to improve network capacity [9], [10]. Further improvement could be achieved by using a more efficient packetization technique and taking into account that partitions B and C are useless when the corresponding A partition is missing. Indeed, an RTP receiver could discard RTP packets containing lone type B and/or C partitions. This feature may be particularly useful for media-aware gateways: when a gateway detects that a type A partition has been lost, it can discard type B and C partitions belonging to the same slice, thus reducing the network load.

VI. CONCLUSIONS AND FUTURE WORK

We studied the support of multimedia services over 802.11 WLANs. Our results suggest that it is possible to significantly reduce the PSNR degradation by exploiting synergies between application layer (H.264 codec) and lower layers. Indeed, the coder can inform the MAC and link layers about which data are of particular importance (e.g., headers or data with best Rate-Distortion tradeoff), so that they can better protected during transmission. Furthermore, lower layers could explicitly notify packet losses and corrupted packets to the decoder so that it could attempt to recover original data through ad hoc error concealment techniques.

REFERENCES

[1] IEEE 802.11 WG, Draft Supplement to Standard for Telecommunication and Information Exchange between Systems – LAN/MAN Specific Requirements – Part II: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: MAC Enhancements for Quality of Service, IEEE 802.11e Draft 11.0, October 2004.

[2] S. Choi, J. D. Prado, S. Shankar, and S. Mangold, "IEEE 802.11e Contention-based Channel Access (EDCF) Performance Evaluation," *IEEE ICC'03*, vol. 2, May 2003, pp. 1151–1156.

[3] A. Grilo and M. Nunes, "Performance Evaluation of IEEE 802.11e," *IEEE PIMRC'02,* Sept. 2002, pp. 511–517.

[4] A. Lindgren, A. Almquist, and O. Schelen, "Quality of Service Schemes for IEEE 802.11 Wireless LANs – An Evaluation," *Mobile Networks*, vol. 8, pp. 223-235, 2003.

[5] Z.-N. Kong, D. H. K. Tsang, B. Bensaou, and D. Gao, "Performance Analysis of IEEE 802.11e Contention-based Channel Access," *IEEE JSAC*, vol. 22, no. 10, Dec. 2004, pp. 2095–2106.

[6] C. Casetti and C.-F. Chiasserini, "Improving Fairness and Throughput for Voice Traffic in 802.11e EDCA," *PIMRC'04*, Barcelona, Spain, Sep. 2004.

[7] J. C. De Martin and D. Quaglia, "Distortion-based Packet Marking for MPEG Video Transmission over Diffserv Networks," *IEEE International Conference on Multimedia and Expo*, 2001.

[8] T. Wiegand, G. J. Sullivan, G. Bjøntegaard and A. Luthra, "Overview of the H.264/AVC Video Coding Standard", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 7, July 2003.

[9] S. Wenger, "H.264/AVC Over IP", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, July 2003.

[10] 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," *IEEE Globecom 2004*, Dallas, TX, Nov. 2004.

[11] R. Caputo, *CISCO Packetized Voice & Data Integration*, McGraw-Hill, 2000.