On the Feasibility of Integrated MPEG Teleconference and Data Transmission, over IEEE 802.11 WLANs

Ioannis Broustis and Michael Paterakis

Information & Computer Networks Laboratory
Department of Electronic & Computer Engineering
Technical University of Crete
73100 Chania, Greece
{broustis, pateraki} @ telecom.tuc.gr

Abstract. The most widespread Wireless Local Area Networks (WLANs) are based today on the IEEE 802.11 standard and its various versions, especially the IEEE 802.11b. In this article we first briefly explain the IEEE 802.11 architecture and the possible implementations of the Data Link Layer. We then present and discuss the results from simulation experiments we performed in order to evaluate the protocol performance. We studied cases of integrated MPEG teleconference and computer data transmissions, over the IEEE 802.11b WLAN. Our simulation results clearly demonstrate the difficulty of the protocol to support time sensitive applications with a large number of wireless users under the same Access Point.

Index Terms: IEEE 802.11b Wireless Local Area Networks, MPEG Teleconference, Video, Network Protocols, Performance Evaluation.

1 Introduction

A WLAN provides its connected users with the capability of communicating with each other, by sending packetized data, without using cables that connect their devices. The WLANs focus on a small geographical area, such as a building, an airport or a part of a university campus. Every time a mobile user wishes to connect to the Internet or to any other network from such place, the connection is almost always carried out via a WLAN. For this kind of access, the user's host is a node belonging to the network, and the LAN provides access to the Internet, via a router. The connected to the WLAN users transmit their data packets via a common wireless channel. Every time a mobile station transmits a packet, this packet "goes out in the open", so every other mobile user may receive it. However, a user usually sends a packet to one recipient only (unicast). In order for a station to be able to send a packet to a specific receiver, it must know the recipient's network address. As a result, we must correspond addresses to users, and transmitted packets must contain the receiver's WLAN address in their headers. By this way, a wireless station is able to know whether a sent packet is designated to it, or to some other station.

In the following sections we briefly explain the possible implementations of the IEEE 802.11b MAC layer, and we indicate the basic concepts concerning multimedia applications and their transmission problems. We especially focus on the integrated teleconference and data transmission over IEEE 802.11b. We perform an extensive simulation study on various scenarios of teleconference video and computer data traffic, and we present and explain results from these simulations.

2 Multimedia Applications and Video

During the last decade, a remarkable explosion of network-multimedia applications has taken place. Such applications have been developed so that users can send/receive audio and video, especially via the Internet. The IP telephony, the prerecorded or the real-time video transmission, the teleconference, the Internet radio and the network games are some of the most popular ones. We can generally distinguish these applications in three different classes: prerecorded audio and video streaming, real-time audio/video streaming, and interactive real-time audio/video streaming.

Multimedia has an important characteristic: it can be compressed. Without compression, multimedia consumes a large amount of disk storage and network bandwidth. A good compression algorithm must not only save disk space, but also avoid multimedia quality loss. Such a good and modern algorithm is MPEG-4 [9]. MPEG-4 manages to put by enough storage, without diminishing the reproduction quality. A 2-hour DVD movie, encoded using the MPEG-2 algorithm [10], consumes almost 7,5 - 8 Gigabytes. If the MPEG-4 algorithm is used, the same movie will now consume about 2,5 Gigabytes, that is 2/3 less disk space.

The multimedia applications differ from other standard widespread Internet applications (such as FTP, e-mail etc), in that they are delay sensitive, and tolerant to few packet losses. As a result, wireless networks that have been developed for fast data transfer, and which provide reliability and security, may not be appropriate for multimedia transmission, because of the various packet delays that are introduced. This time sensitivity generates many problems in multimedia transmission over some wireless media, because, if the packet delay becomes larger than a specific value (usually of the order of few tens of milliseconds), those packets will then be considered as old and will be dropped, as the receiver's application layer cannot wait more for these packets. However, multimedia applications are packet loss tolerant, and, according to how strict they are about the maximum percentage of packets that may be lost, we can determine a QoS *(Quality of Service)* level for the transmissions. Furthermore, if we do not lose packets very often, the gaps from the lost packets may be partially or completely covered (in some occasions), using various techniques [4].

3 The 802.11 Architecture

In order for a mobile station to be able to connect to an 802.11 WLAN, a wireless card must be available. The fundamental building block of the 802.11 is the BSS (*Basic Service Set*). The BSS is a set of stations that use one of the following coordination functions: DCF (*Distributed Coordination Function*), or PCF (*Point Coordination Function*), which we briefly explain later in this section. The geographical area, covered by a BSS is called BSA (*Basic Service Area*) and corresponds to the notion of a "cell" that is used in mobile telephony. Stations that belong to the same BSS are able to communicate directly to each other, as in an Ad-Hoc network. In contrast to the Ad-Hoc network topology, *Infrastructure networks* have been developed to provide specific services and cover wider ranges. The basic equipment includes an AP *(Access Point*), which corresponds to the BTS (*Base Transceiver Station*), used in mobile telephony networks [3], [5]. Using APs we manage to interconnect stations outside the limits of a BSS. As a result, APs connect a BSS with the rest of the network world, so as to create an ESS (*Extended Service Set*). An ESS consists of many interconnected BSSs through a DS *(Distribution system*). The Distribution System may be considered as a *backbone* network, responsible for the data link layer transmissions [3].

3.1 MAC Layer

In 802.11, CSMA-CA (*Carrier Sense Multiple Access with Collision Avoidance*) is used: when a node wants to transmit, it senses the medium and if it does not detect any activity, it will wait for an additional time interval and will sense the medium again. If the medium is still idle, then the station will assume that nobody uses the medium, so it will send a data packet.

The mechanism that stations follow to gain access to the medium may alternate between two different modes: Contention-free and Contention. While in contention-free, (*CFP – Contention Free Period*), access to the channel is determined by an AP (*Access Point*), in that the AP decides which station has the right to transmit during a certain period of time. During this period, other stations only hear the medium and do not have authorization to transmit. In contrast to CFP, while in CP *(Contention Period*) the CSMA-CA algorithm is followed.

3.1.1 Distributed Coordination Function

DCF is the fundamental channel access method, based on the CSMA-CA protocol, and is used to support asynchronous data transfer. All stations are required to be able to operate in DCF, which may coexist with the PCF (described later), or operate solely. While in DCF, every station with a packet to transmit must contend to gain access to the medium. In every access, the wireless node may transmit one data frame only and, after the end of this transmission, it must re-contend, in order to transmit another frame from its queue [1], [2].

The main advantage of this algorithm is fairness: every station must re-contend for the channel after every packet transmission. All nodes have equal probability of gaining access to the medium after each *DCF-InterFrame Space* (DIFS) time interval. The basic disadvantage though, is that it does not guarantee a minimum access delay to stations running time-sensitive applications.

3.1.2 Point Coordination Function

PCF is an optional connection-oriented operation, based on a PC (*Point Coordinator*), whose responsibility is to select the station that transmits during a specific time period. Wireless nodes do not have to contend with each other in order to gain access to the medium. The AP usually performs the role of the PC. The way stations are selected is based on an algorithm, chosen by the PC (e.g. Round Robin). PCF provides contention-free (CF) frame transfers and is usually used for time-bounded services.

PCF needs Access Points. All nodes obey the medium access rules determined by the AP/PC. Every time the AP contacts a station, giving it the order to transmit, the station may transmit only one frame, destined to any other station (inside the BSA or not), and not just to the AP/PC. PCF controls the CFP *(Contention Free Period*). At the nominal start of every CFP, the AP/PC senses the medium. If the medium is considered to be idle for a *PCF_InterFrame Space* (PIFS) time period, it will broadcast a beacon frame to all nodes, informing them that CFP has began. If neither the stations nor the PC have packets to transmit, the PC may terminate the PCF immediately after the beacon frame transmission [1].

The entire channel operation, alternating between PCF and DCF periods, is shown in the following figure. Finishing with the brief protocol description, we notice that because of the alternating operation (PCF - DCF), we are able to transmit different kinds of data during each function. Thus, we may use CFP to transmit delay sensitive data, such as audio and video, and leave other kinds of data to be transmitted during the CP.

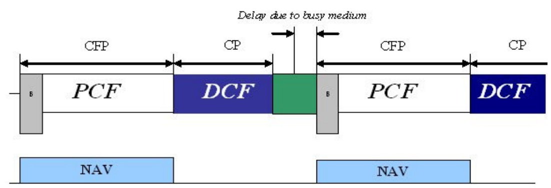

Fig. 1. Alternation between Point Coordination and Distributed Coordination Functions

Indeed, delay sensitive data packets cannot be efficiently transmitted, if they collide frequently, as additional delays are incurred in such cases. In contrast, data packets that can put up with some delay (such as FTP or email data) may wait more, so, they can be transmitted during the CP.

4 The Simulator

In order to determine the protocol's efficiency and the network capacity, we developed a software simulator. It is generally difficult to develop such simulator from scratch, because of the significant complexity of the protocol. Our simulator does not cover cases of noisy environment and hidden nodes. Some of protocol's parameters either could not be found, or are left to the implementer - administrator. Thus, we assumed certain values for these parameters, shown in Table 1, based on the relevant bibliography and the values that have been used in protocol simulations by others. We varied these values in a number of experiments and noticed that they do not significantly affect the results.

Table 1. Values of some important parameters, used in our simulations

DIFS	34 μsec
SIFS	16 μsec
PIFS	25 μsec
Slot time	9 μsec
DCF period	1000 μsec
PCF period	10000 μsec

The simulator views the wireless network as follows. Every single station maintains two separate queues: one for video packets and one for data packets. The teleconference application, assumed to run at each station, adds fragmented video frames into the station's video queue. The video frame lengths (in bytes) are being read from 1-hour playback duration trace files, corresponding to MPEG-4 teleconference movies [7]. Every node has its own trace file, which is different from the others', meaning that every 40 msecs, stations read a new different (for every station) video frame, which they fragment and insert into their video queues. Moreover, every station's data queue is being filled with ATM size packets, according to a Poisson arrival process with arrival rate $\boldsymbol{\lambda}$. The video system is represented with a circular queue, nodes of which are the stations' separate video queues. In addition, the data system is represented with another circular queue, nodes of which are the separate stations' data queues. For the representation of the AP we follow a similar design: There is a circular queue, nodes of which are video queues, one for every destination station. Every time station A wants to send a video packet to station B, it sends the video packet to the AP. This packet is stored in B's queue at the AP. When the AP contacts station B, it will send packets to it from its queue, maintained at the AP. There is no need for a data circular queue at the AP, since we assume that data transfers (DCF) occur between stations of the same BSS, therefore, data packets are sent directly to the receivers. The simulator is time – triggering; we assume the existence of a minimum time period of 1 μsec, and represent all time values as multiples of this time period [8]. Finally, the simulator was developed in C++.

4.1 Scenario 1: Full Teleconference

In this scenario we assume that we have a number of users in a BSA, which use a common teleconference application to communicate, under the IEEE 802.11b wireless environment, with a transmission rate of 11Mbps. Every user has a video camera connected to a portable PC (laptop), and either can move inside the BSA, or may stay still at a certain point. We also assume that every station is running several other applications, among which is a data transfer application, used to send ATM size data packets to the other teleconference users. The video produced from the teleconference application is encoded using the MPEG4 algorithm, and the time elapsed between two video frame arrivals is 40 msecs. Furthermore, video packets have a maximum life time of 40 msecs in the system. The ATM data packets are generated according to a Poisson arrival process, with arrival rate $\boldsymbol{\lambda}$. We also assume that the protocol alternates between CP and CFP, so both PCF and DCF are used. Stations exchange video packets during the PCF period and data packets during the DCF period of time.

In order for a station to participate in the teleconference, it must send its video packets to all the other teleconference users, and receive video packets from all other users. As a result, a packet that has been received by a node, has been sent to all the other teleconference nodes, except the one who sent it. Such functionality could be easily performed if the sender sent the packet to the AP, and the AP broadcasted or multicasted it to all other teleconference participants, by attaching a broadcast/multicast address corresponding to the set of them. However, in our scheme we do not select this broadcast/multicast functionality, since it is unreliable. Indeed, if a packet is broadcasted or multicasted to a set of nodes, then those nodes - recipients do not send acknowledgements back to the sender. That is, the sender does not have any clue about whether a video packet reached correctly all teleconference participants, so as to decide whether to retransmit it or not. Because of the latter, we avoid using the broadcast/multicast feature, even though it appears to be the easiest way to send the video packets to the teleconference participants. Thus, we have to find another way to reliably send video packets from a node to all the other teleconference nodes, before the 40 msecs lifetime deadline is exceeded. We assume that we have a transport protocol, responsible for the end-to-end packet delivery, such as UDP, which sets up the path that packets follow. As a result, every node has a separate UDP connection with every other teleconference node in the BSS. A solution to the problem of sending a packet to all other stations is to send that packet separately to each station ("*multicast through unicast"*). To do this, every video packet must be copied as many times as its recipients, that is all nodes minus one – the sender, and every copy must be sent to a recipient. There are two ways to implement this network operation:

1. The originator of the packet performs the copy procedure: When a station contacts the AP, it must send as many packet copies as the number of the other teleconference users. The problem here appears when the sender is given the authorization to transmit. It can send only one packet upstream. Thus, if we have, for example, 8 teleconference users, the sender must transmit 7 packet copies, one copy for every time it contacts the AP. So it will take seven round robin cycles until the packet has completely left the transmitter. This is a significant amount of time for the video packets waiting in queues.

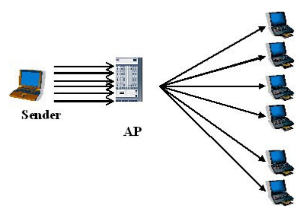

Fig. 2. The packet originator sends copies of the packet to the AP, which further delivers them

The 40 msecs lifetime deadline is frequently exceeded and video packets are dropped while inside the sender's queue - that is while in the process of being transmitted on the uplink.

2. The copy procedure is performed by a copy function behind the AP/PC: In this case, a packet copy (replication) function must operate behind the AP. Every time a video packet arrives at the AP, the copy function replicates the packet as many times as the number of destination stations. The AP stores the copies in its local buffers and sends these copies to the stations (one by one) every time it contacts them.

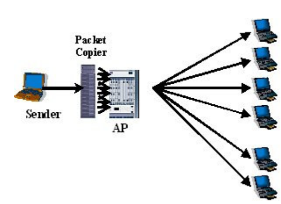

Fig. 3. A packet copier supplies the AP with copies of the packet sent by the originator

Even in this case, however, we observe packet loss, *on the downlink*, rather than on the uplink: the Access Point has stored many packet copies, which does not manage to transmit - at least not all of them, as their lifetimes expire, while still stored in AP's buffers. In the sequel, we assume the existence of a copy function behind the AP/PC.

Initially we assume that the teleconference application fragments video frames to packets with maximum length of 1000 bytes. The video *mean bit rate* is 42 Kbps and the *peak bit rate* is 690Kbps. Moreover, as we have mentioned before, the ATM data packets are transmitted during the DCF, they have a constant length of 53 bytes and the Poisson arrival rate is $\lambda=4*10^{-4}$, that is 4 data packets every 10 msecs. Furthermore, we require a quite small video packet loss, specifically, we require: *Packet Video Drop (PVD)* $\leq 10^{-4}$, where PVD stands for the number of the overall lost video packets, divided by the number of the overall video packets that have been generated. The following figures represent the PVD and the average video packet delay as we add more teleconference stations in the network, respectively:

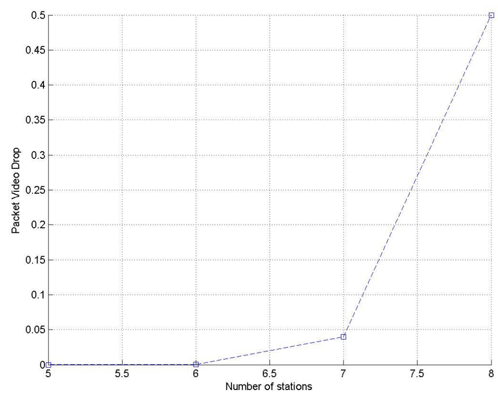

Fig. 4a. For 7 users or more, the system does not operate efficiently; the PVD increment is prohibitive for our QoS requirements

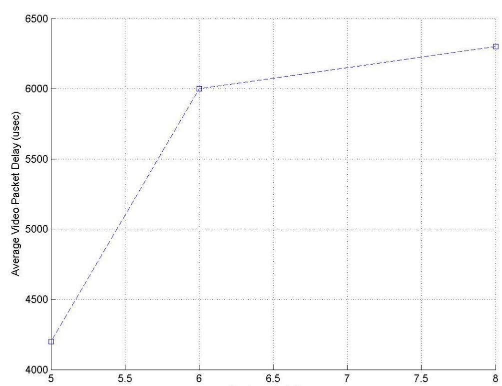

Fig. 4b. Average video packet delays are acceptable

From figures 4a and 4b we notice that the system operates efficiently with a maximum capacity of 6 stations. If we increase the number of stations to 7 or more, we observe a severe PVD increase, something unacceptable for the assumed QoS level. Moreover, as we expect, the average video packet delay increases too, since many more video packets are waiting to be transmitted now. Of course, a maximum number of 6 users may be a bit small for certain teleconference scenarios. This relatively small capacity is due to the protocol's significant time overheads. All those time safety valves consume quite large time that otherwise could have been used for actual packet transmissions. As a result the channel throughput decreases significantly. As we also notice, concerning the DCF, by adding more stations we observe large packet delays leading the system to be unstable, (something that we thoroughly explain later). For 6 users, however, packet delays and loss are acceptable, as we see from figures 4a, 4b and 5.

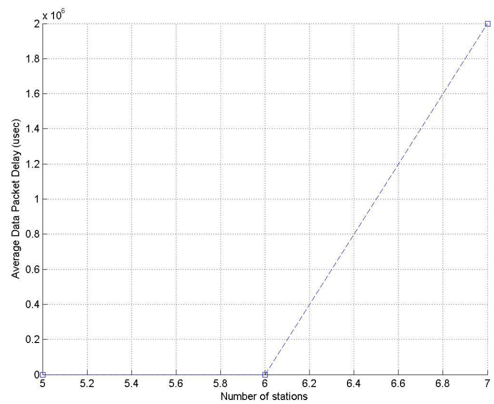

Fig. 5. Average data packet delays are acceptable

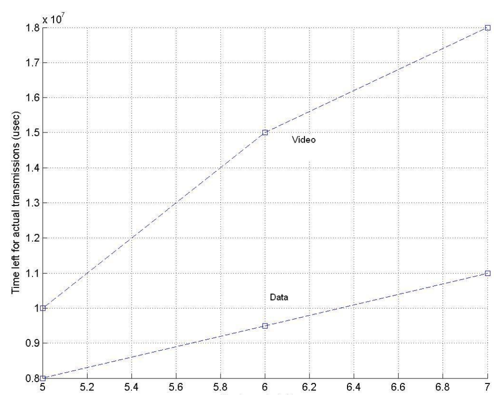

Fig. 6. The amount of time left for video and data packet transmissions is quite small

The channel time used for actual packet transmissions is quite small. From the simulations we see that for 6 stations and 100 secs of network operation, video packets are transmitted for 15 seconds (overall) only and data packets for 10 seconds only. The remaining 75 seconds constitute the time overhead and the CP periods. These numbers do not seem to change dramatically for a different number of wireless stations, as we see from figure 6 above. It is quite interesting to see how the average data packet delay and loss change, for different data packet arrival rates λ. In the simulation we assumed 6 teleconference users and a maximum video packet size of 1000 bytes (payload). *The values are aggregate, (i.e. for the packets of all stations).*

Table 2. Average data packet delays and data packet losses for various values of λ

λ	*Arrivals*	*Losses*	*Average Data Packet Delay*
10^{-4}	59908	0	1439,50 μsec
$2*10^{-4}$	120278	1	2968,09 μsec
$3*10^{-4}$	179301	16	4975,05 μsec
$4*10^{-4}$	240134	178	11054,0 μsec
$5*10^{-4}$	299446	1050	2868192 μsec

From the results in Table 2 we observe that, for $\lambda \geq 5*10^{-4}$ and for the maximum sustained number of 6 teleconference users, the system is unstable: data packet loss (due to experiencing a number of consecutive collisions higher than the protocol's threshold) and average data packet delays are quite large and increase as the simulation time is prolonged. We must also mention here that the changes in **λ** do not affect the *video packet loss*, while they slightly affect the average video packet delays, as shown in Table 3.

Table 3. Average video packet delays and video packet losses for various values of λ

λ	*Losses*	*Average Video Packet Delay*
10^{-4}	5	5781,70 μsec
$3*10^{-4}$	5	5969,13 μsec
$4*10^{-4}$	5	6008.20 μsec
$5*10^{-4}$	7	6010,20 μsec

This behavior is due to the following reason. For larger values of **λ** every data queue fills with more packets. This means that the medium is busy for a longer time during the DCF. It is thus likely that consecutive transmissions will be taking place, even at the time point that DCF ends. As a result, the protocol has to extend the DCF period, so that the current data packet transmission can be completed. This time extension acts burdensomely for the delays of the video packets waiting to be transmitted during the following CFP. We conclude by saying that the DCF period extensions generally increase the data and video packet delays. Respectively, the same conclusion stands for the PCF extension.

4.2 Scenario 2: Teleconference for Pairs of Users

This case differs from the previous one, in that every pair of users holds its own teleconference. We have as many teleconferences in progress, as the number of users divided by two (fig. 7). Of course we assume an even number of users. As before, we assume that, all parallel teleconferences begin at the same time, so, new video frames arrive at the same time to the wireless stations. Because of the nature of this scenario, a video packet sent by a node is destined to one node only, the teleconference partner of the transmitter. As a result, we do not have here to replicate packets destined to a number of receivers. As expected, the system capacity increases. From the results presented in table 4, we can see that the system can now support up to 16 users (8 pairs), that is 10 more users than in the previous scenario.

As we expect, if we have more teleconference user pairs, video and data packet delays are increased. As we add more users, we observe a severe increase in the average data packet delays, during the DCF period.

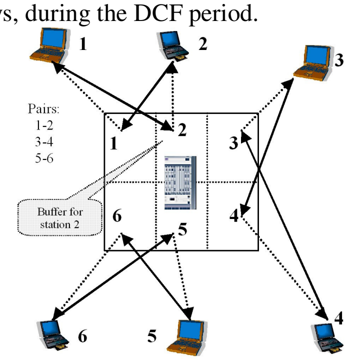

Fig. 7. Teleconference for pairs of users

This is because, in a system with many users, it is more difficult for a station to gain access to the medium through the CSMA-CA algorithm. A large number of users implies frequent packet collisions and thus, increased data packet delays.

Table 4. Average video packet delays and video packet losses for pairs of teleconference users

Users	*Arrivals*	*Losses*	*Average Video Packet Delays*
6	16252	0	2378,303 μsec
8	21654	0	3196,960 μsec
10	27098	0	3991,544 μsec
12	32766	0	4837,170 μsec
14	38207	0	5304,612 μsec
16	43640	0	6148,205 μsec
18	49060	372	6603,696 μsec

4.3 Scenario 3: Teleconference for Groups of Users

In the last scenario we assume groups of teleconference users under the same AP. For the groups with more than two users, we must use a copy function that replicates video packets behind the AP. These packets suffer higher delays (scenario #1). In contrast, for teleconference groups with two users only, we do not have to copy packets. As a result, video packets exchanged between these pairs of users do not suffer from time delays, at least not as much as packets in larger teleconference groups. Indeed, packet loss (because of lifetime deadlines) occurs more frequently in groups with many users. The advantage of the system is that, groups of different sizes do not interfere with each other. Large groups suffer from delays and losses, while small ones hold teleconference more efficiently. In the following tables we provide representative simulation results, supporting all of the above statements.

Table 5. One group of three users and another of two users

	Groups	***Losses***	***Arrivals***	***Average Delay***
VIDEO	0 – 1 – 2	0	0: 2709, 1: 2709, 2: 2709	6055.66 μsec
	3 – 4	0	3: 2708, 4: 2708	2055 μsec
DATA	***λ***	***Losses***	***Arrivals***	***Average Delay***
	0.0004	0	199460	1016.298 μsec

Table 6. One group of three users and two other groups of two users each

	Groups	***Losses***	***Arrivals***	***Average Delay***
VIDEO	0 – 1 – 2	0	0: 2707, 1: 2707, 2: 2707	8101.66 μsec
	3 – 5	0	3: 2708, 5: 2708	2821 μsec
	4 – 6	0	4: 2707, 6: 2707	3051.5 μsec
DATA	***λ***	***Losses***	***Arrivals***	***Average Delay***
	0.0004	91	279303	4447,46 μsec

From tables 5 and 6 we can see that the simulation results are quite satisfactory and agree with intuition. Thus, in the first sub-scenario the group with the three nodes encounters larger time delays, because we are obliged to replicate all video packets at the AP. In contrast, the other two stations, involved in the second group, encounter smaller delays, as video packets are sent directly between partners. In the second sub-scenario we notice similar results. We should point here, that in this second case, packets exchanged between pairs of users in groups 2 and 3 encounter larger time delays from the packets exchanged between pairs of users in the first sub-scenario. This is because in the second case we have added two more users (which constitute a new teleconference pair) under the same AP. The following results are shown to further support our observations and conclusions.

Table 7. Two groups of three users each and a group of two users:

	Groups	***Losses***	***Arrivals***	***Average Delay***
VIDEO	0 – 1 – 2	0	0: 2709, 1: 2709, 2: 2709	9618.61 μsec
	3 – 4	0	3: 2708, 4: 2708	3145.7 μsec
	5 – 6 - 7	0	5: 2707, 6: 2707, 7: 2709	9668.8 μsec
DATA	***λ***	***Losses***	***Arrivals***	***Average Delay***
	0.0004	2164	319443	*1479940.1* μsec

Table 8. Three groups of two users each and a group of five users:

	Groups	***Losses***	***Arrivals***	***Average Delay***
VIDEO	0 – 1 – 2 – 7 – 8	0:629, 1:615 2:622, 7:652, 8: 651	0: 2710, 1: 2710 2: 2710, 7: 2710 8: 2710	24721,56 μsec
	3 – 6	0	3: 2710, 6: 2710	4367,2 μsec
	5 – 9	0	5: 2710, 9: 2710	4750 μsec
	4 – 10	0	4: 2710, 10:2710	4383 μsec
DATA	***λ***	***Losses***	***Arrivals***	***Average Delay***
	0.0004	205	218578	7074,45 μsec

From the results of the last sub-scenario we observe that the group with the five nodes suffers severe packet delays and loss. In contrast, the other groups hold teleconference quite efficiently.

5 Conclusions

The purpose of the work presented in this paper was to examine the capability of the widespread IEEE 802.11b protocol, to efficiently support time-sensitive applications. We chose the integrated MPEG teleconference and data exchange, and we examined scenarios of full teleconference, teleconference with pairs of users and teleconference with groups of different numbers of users. In the first scenario (full teleconference) the system may support up to 6 wireless stations. As it is expected, in the 3rd scenario (groups of teleconference users) we cannot transmit video and data efficiently, if we have a group with more than 5 users, as this will be an extension to the first scenario. In reality, according to the number of groups and the number of stations per group, the system may support a larger or a smaller number of parallel teleconferences. This is clear in the second scenario, in which every group consists of 2 stations only, thus the system may support 16 users. Potentially, a maximum number of wireless nodes in one group is 5, however if we have a few users per group, we may support more teleconferences. As for the data packet transmissions during DCF, the simulation results are as expected. By adding more stations we observe an increase in the average data packet delay and in the packet losses.

In conclusion, our simulation results clearly demonstrate the difficulty of the protocol to support time-sensitive applications with a large number of wireless users, under the same Access Point.

References

1. ANSI/IEEE 802.11 Standard, 1999 Edition.
2. IEEE 802.11b Standard, 1999, (Supplement to ANSI/IEEE 802.11 Standard, 1999 Edition).
3. Brian P. Crow, Indra Widjaja, Jeong Geun Kim, Prescott T. Sakai: "IEEE 802.11 Wireless Local Area Networks", IEEE Communications Magazine, 1997.
4. James F. Kurose, Keith W. Ross: Computer Networking: "A Top Down Approach Featuring The Internet", Addison – Wesley Publications, 2001.
5. F. Cali, M. Conti, E. Gregori: "IEEE 802.11 Wireless LAN: Capacity Analysis and Protocol Enhancement", Proceedings of the 1998 IEEE Infocom Conference.
6. Breeze Wireless Communications Ltd.: IEEE 802.11 Technical Tutorial, Israel 2000.
7. Video traces: http://www.tkn.ee.tuberlin.de/research/trace/trace.html
8. Averill M. Law, W. David Kelton: "Simulation modeling & Analysis", 2nd Edition, McGraw Hill International Editions, 1991.
9. Frank H. P. Fitzek, Martin Reisslein: "MPEG-4 and H.263 Video Traces for Network Performance Evaluation (Extended Version)", TU Berlin, Dept. of Electrical Engineering, Telecommunication Networks Group, Technical Report: TKN-00-06, October 2000, http://www.tkn.ee.tu berlin.de/research/trace/pub.html
10. P. N. Tudor: "MPEG-2 Video Compression", Tutorial, Electronics and Communications Journal, December 1995, http://www.bbc.co.uk/rd/pubs/papers/paper_14/paper_14.html