Digital Entertainment Delivery in a Wireless House: Time for a MAC Tuning

Claudio E. Palazzi*†, Giovanni Pau*, Marco Roccetti†, Mario Gerla*
***Computer Science Department,**
University of California, Los Angeles, CA 90095, USA
Email: {cpalazzi, gpau, gerla}@cs.ucla.edu
† Dipartimento di Scienze dell'informazione,
Universit`a di Bologna, 40126, Bologna, Italia
Email: roccetti@cs.unibo.it

ABSTRACT

Computer-centered services and broadband wireless connectivity are going to allow a direct delivery of digital entertainment from the Internet to in-house mobile devices. Yet, current architecture and protocols are not optimized for an efficient coexistence of downloading flows and real-time ones. With current systems, in fact, real-time applications suffer from delays caused by the interference with the transmission behaviors of downloading ones. In this paper, we investigate the impact of MAC layer design choices on the distribution of in-house entertainment contents showing how current settings in the IEEE 802.11 protocol do not correspond to the optimal choice in this context. Finally, we provide directions to set MAC layer parameters in order to solve the tradeoff relationship between the performance requirements of downloading and real-time applications.

Key words: digital entertainment, wireless house, MAC tuning

I. INTRODUCTION

Digital entertainment is going to play a major role among applications run in a domestic environment. Indeed, the already popular TiVo and Media Center [2] are just two simple exemplars of what we call a *Home Entertainment Center* (HEC). A HEC can be seen as a technological hub able to retrieve and store all of a consumer's digital content and to deliver it to users located in various parts of the house[1]. It is expected that HECs will expand their features becoming, within few years, *Home Servers* controlling the whole home connectivity and information system [3].

As the availability of this technology is increasing rapidly, the need for interconnecting digital entertainment devices is felt as ever more urgent, as well as the necessity to extend the reach of HECs to the wireless domain. In the today's market, IEEE802.11[1] based wireless LANs are emerging as the best candidate to lead the "home networks" revolution providing wireless connectivity and advanced functions in terms of flexibility, security and throughput to support new emerging entertainment applications ranging from networked games to in-house digital audio/video distribution and live conferencing etc.[4]. Yet not much work has been done to understand whether the Internet native language (i.e. the TCP/IP protocol) will be able to support this complex scenario.

To this aim, we are aimed at studying several realistic home networking scenarios for entertainment over IEEE802.11g networks. In particular, we focus our

This work is partially supported by the Italian Ministry for Research via the InterLink project and the ICTP/E-Grid Initiative, the National Science Foundation through grants CNS-0435515/ANI-0221528, the UC Discovery Grant Core com 04-10170 private sponsor STMicroelectronics and the UC-Micro Grant MICRO 04-33 private sponsor STMicroelectronics. Correspond-ing author: Claudio E. Palazzi, Department of Computer Science -University of California Los Angeles, CA 90095, e-mail:cpalazzi@cs.ucla.edu.

[1]Currently the IEEE802.11 based WLAN offers two different transmission rates: (i) 11Mbit/s (IEEE802.11b) and (ii) 54Mbit/s (IEEE802.11g/a). The next generation the IEEE802.11n, based on MIMO technologies, designed to support a transmission rate of 100Mbit/s has been already announced and should be available by the end of 2006.

attention on the cross-layer interactions between transport and MAC layer when digital contents are distributed to entertainment devices. We also show how (UDP-based) real-time applications suffer from delays caused by the interference with (TCP-based) downloading ones. This represent the reverse of the well known argument by which UDP's lack of congestion control would harm TCP, whereas we will show how even a single persistent TCP-based flow (i.e., a download of a medium size file) can deteriorate the performance of UDP-based applications (i.e., real-time ones).

Finally, we propose a tuning of the currently deployed off-the-shelf low-cost wireless gateways to provide a solution to the tradeoff between the different types of traffic patterns that may exist during the activity of entertainment distribution in a home network.

This paper is organized as follows. In Section II we design a typical home networking scenario. Section III depicts the background of this work. In Section IV we provide the simulation assessment, while Section V presents and discusses results. Finally, Section VI concludes this paper.

II. HOME NETWORKING SCENARIO

The current market is heading toward a wireless house where all the devices will be wirelessly connected to the home network and possibly controlled by the HEC. The wireless connectivity could be provided by the popular IEEE 802.11g technology featuring a nominal rate of 54Mbit/s and a factually available one that can reach 20Mbit/s.

In this context, a family presumably owns several networked personal portable devices such as PDAs, MP3 players, game consoles and digital cameras; all these being also connected to the home network.

Video streaming, video chat, online games, and mp3 down-loads are only a few of the vast plethora of possible applications run on these devices. However, each of them is characterized by specific sensitiveness to certain problems and hence metrics to evaluate their performances.

In particular:

- Video Streaming. Streaming applications are affected by the jitter phenomenon[2] while are resilient to some packet loss; a network designed mainly for video streaming should minimize the jitter.
- Video Chat and Massive Multiplayer Online Games: Both this applications require a high degree of interactiv-ity, they greatly suffer from delays and packet jitter while may tolerate some packet loss.
- iTunes Music download: A music download activity is typically performed using TCP, hence this type of application is resilient to jitter and delays but decreases the sending rate in presence of losses: it hence does not tolerate any error losses (losses that are not generated because of congestion).

Finally, we assume that several family members will be accessing the household network at the same time according to their work or leisure needs. For instance: (a) one teenager is watching the movie “Star Wars”, streaming it from the close-by HEC; (b) the other one is playing with his latest online game against a crowd of buddies across the Internet; (c) the dad is having a conversation through an IP based video chat; and (d) the mum is downloading the last U2 greatest hit compilation from the Apple iTunes© music store. In the above everyday-life picture, it is important that all the applications result able to coexist without interfering each other, all achieving good performance levels.

III. TECHNICAL BACKGROUND

A. The transmission control protocol

The most popular TCP version, TCP New Reno, implements a congestion control algorithm, known as AIMD (Additive Increase, Multiplicative Decrease). In this context, the conges-tion window represents the number of packets that the sender can send towards the destination without having yet received any acknowledgment about their delivery from the receiver. Indeed, for every packet received, the receiver sends back to the sender an acknowledgment (ack) that identifies the next packet expected and implying that all the preceding packets in the sequence where successfully delivered.

The very basic concept of the congestion control scheme can be summarized as follows:

- The number of packets sent out without having yet received back their corresponding acks cannot be higher than the current congestion window;
- For each successfully delivered packet, a new one is sent;
- When a whole congestion window of packets has been successfully delivered, the congestion window value is increased by 1;
- If the sender receives three consecutive duplicate acks (dupacks) from the receiver, a packet loss is inferred, the corresponding packet is sent again and the congestion window is halved.

This scheme has been developed following the end-to-end paradigm by which the two involved end-nodes do not have any explicit information about the links connecting them. In essence, the Internet is seen as a black box whose content remains unknown and all the intelligence is left at the edge. Sender and receiver are unaware of the available bandwidth on the links among them and of the possible presence of other flows along the same path. The sender has hence to continuously probe the channel to make use of bandwidth that might be available and to back off when congestion is detected.

The resulting TCP traffic continuously oscillates and reaches peaks that exceeds the available bandwidth on the channel. In presence of persistent TCP connections it is hence very likely to happen that buffers are steadily fully utilized, thus periodically slowing down the delivery time of every packets transiting through them and deteriorating the performance of time-sensitive applications.

B. The IEEE 802.11 MAC protocol

The 802.11 MAC protocol attempts to face the packet loss problem over a wireless channel by implementing its own retransmission scheme [5]. In particular, lost packets are retransmitted after a certain period of time without having received any corresponding ack. Successive retransmissions for the same packet are repeated up to a maximum number of time, which is by default set to 4 in the standard IEEE802.11, or until receiving a successful ack. A backoff mechanism determines the retransmission timeouts.

This scheme hides wireless error losses from the TCP's congestion control mechanism, thus avoiding deleterious multiple reductions of the data sending window. On the other hand, local retransmissions affect packet delivery delay by increasing its variability and thereby affecting time-constrained applications such as audio or video stream.

C. Related work

In recent years, many researchers have focused their studies on the problems that TCP encounters in a wireless environ-ment [6]. Experimenting TCP and UDP over IEEE802.11 with different signal levels, Nam et al. showed that without retransmission implemented at the link layer, loss rates become unacceptable for any application [7], [8]. The claim that MAC layer retransmissions improve TCP performance is confirmed by Xylomenos and Polyzos, who experimented TCP and UDP on a WLAN and analyzed their behavior with different interfaces and bidirectional TCP traffic [9]. On the other hand, a high number of repeated retransmissions can cause TCP to timeout anyway and retransmit the same data as the MAC layer. Moreover, MAC retransmissions can be wasteful and potentially harmful for time-sensitive applications, such as real-time video or audio over UDP.

Table.1 Simulation Configuration (Wired Links)

Node 1	Node 2	One-way Delay	Capacity	Buffer Size
W1	W0	10ms	100Mbps	140pkts
W2	W0	20ms	100Mbps	140pkts
W3	W0	30ms	100Mbps	140pkts
W0	AP	10ms	100Mbps	140pkts

IV. SIMULATION ASSESSMENT

To analyze our scenario, we have utilized enhanced the NS-2 network simulator (version ns-2.28) with MAC layer parameters of the IEEE 802.11g. Since our interest for a wireless home, we utilized the Shadowing Model for the wireless environment as it represents the most realistic wireless model available in NS-2. Following directions provided by the official NS-2 manual, we represented a home environment partitioned into several rooms through setting the path loss exponent equal to 4 and the shadowing deviation equal to 9. Transmission signal attenuation grows with the increase of these parameters; by setting the shadowing deviation value with the highest value suggested by the NS-2 manual for an indoor environment (i.e., 9) we expect to face higher percentage of packet losses.

As depicted by Figure 1 we have considered a HEC positioned in the middle of the house and endowed with an AP to deliver entertainment to four mobile nodes. The Internet has been represented in a very simple way as described by Figure 1 and Table I. Indeed, we want to focus our analysis on problems that emerge in a wireless home scenario, the configuration of the links was hence chosen so as to have the bottleneck in correspondence of the last hop (i.e., the wireless connection at home).

As for the traffic simulated in our network topology, Table II provides details of the different applications run. Aimed at creating a realistic scenario, we have exploited real trace files for the movie stream and for the video-chat. Specifically, adopted trace files contain real packet size and rate for high quality MPEG4 Star Wars IV (for the movie stream), and two VBR H.

[2] The packet Jitter is defined by RFC 2598 as defined jitter as the absolute value of the difference between the arrival times of two adjacent packets minus their departure times, $|(a_j-d_j) - (a_i-d_i)|$.

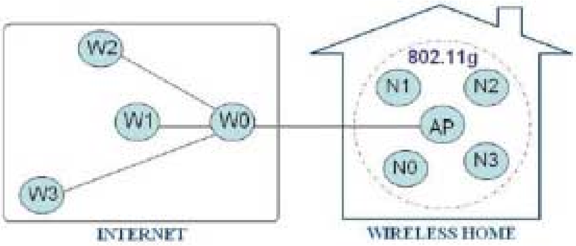

Fig.1 Simulated topology

Table.2 Simulated Application Layer Traffic Flows

From	To	Type	Transport Protocol	Start	End
AP	N0	Movie Stream	UDP	0s	180s
W1	N1	Game Traf_c	UDP	45s	180s
N1	W1	Game Traf_c	UDP	46s	180s
W2	N2	Video Chat	UDP	90s	180s
N2	W2	Video Chat	UDP	91s	180s
W3	N3	FTP	TCP	135s	180s

Table.3 Simulation Parameters

Parameter	Values	Comments
MAC data retransmissions	1, 2, 3, 4	default value = 4
user-AP distance (m)	5, 10	same room, different room
MAC buffer size (pkts)	50, 100	common default values

263 Lecture Room-Cam (for the video chat) [10]. Moreover, parameters characterizing the game-generated traffic have been chosen following the directions provided by scientific literature in this field. Assuming that the player in the house is engaged in the popular first person shooter game Quake Counter Strike, with other 25 players through the Internet, we took inspiration from real measurements reported in [11] to model the game traffic.

Simulation experiments were replicated to examine the effects generated by differently setting some of the parameters involved in the scenario. Table III lists all the variable parameters in the simulations; each combination of their possible values was simulated.

V. RESULTS

A. FTP interference on real-time applications

We demonstrate how even a single FTP flow can jeopardize the performance of real-time applications. To this aim, we have intentionally started the various flows at different times to better appreciate the impact of each one of them on a concurrent real-time application. In this context, Figure 2 and Figure 3 show the jitter experienced by the movie stream application (starting at 0s) and the online game application (starting at 45s), respectively.

As it is evident, the jitter remains low until it is the turn for the downloading application to take action (starting at 135s). At that point, the channel and the buffers along the path are quickly saturated and the jitter experiences peaks of tens of milliseconds. The exact correspondence between the jitter peaks of the real-time applications and the congestion window peaks in the TCP flow supporting the downloading application is evident in Figure 4. In this chart, cwnd, ssth, and RTTxBW represent the congestion window, the slow start threshold and the bandwidth per RTT product, respectively. When the congestion window reaches the pipe size of the chan-

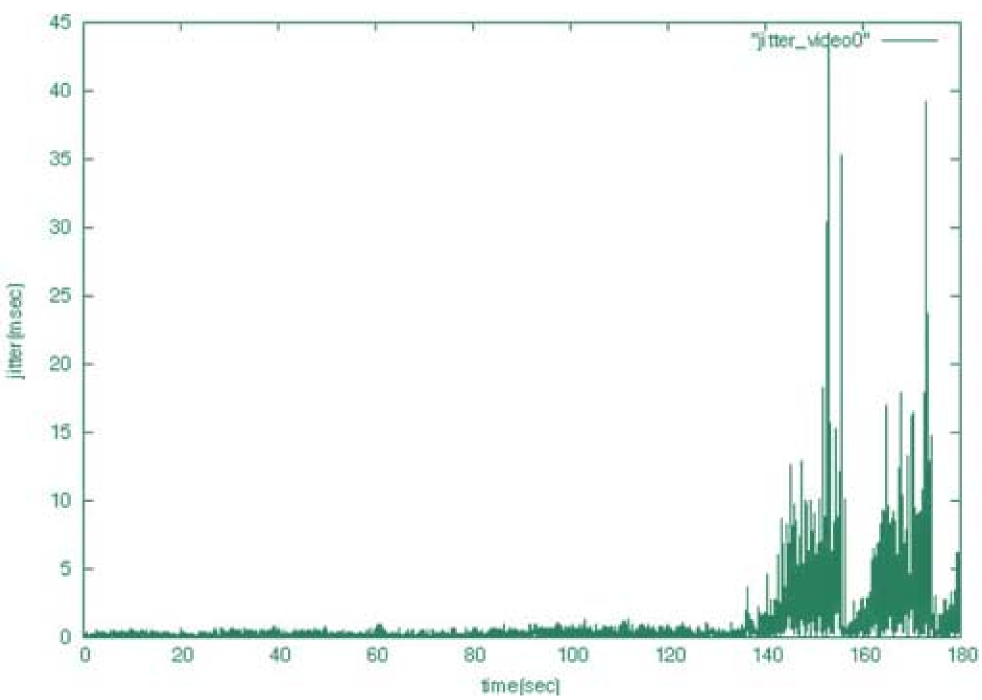

Fig.2 Jitter experienced by a video stream application; from 135s, a FTP/TCP flow is simultaneously run.

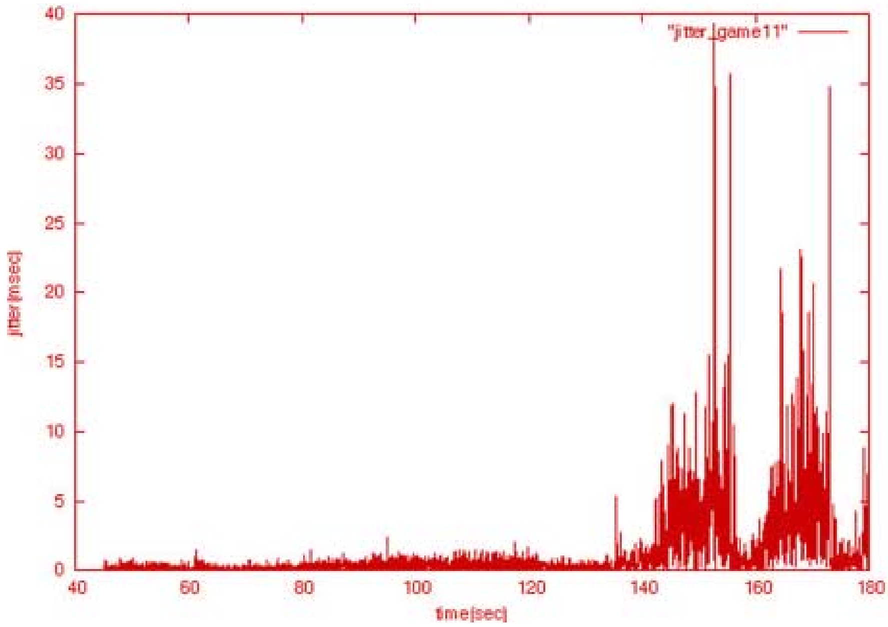

Fig.3 Jitter experienced by an online game application; from 135s, a FTP/TCP flow is simultaneously run.

nel (i.e., RTTxBW), packets start to be queued at the AP, thus increasing the delivery delay of all packets transiting through that AP and affecting real-time applications' performances.

B. Mobile-AP distance impact on FTP goodput

It goes without saying that the quality of wireless transmissions decreases with the distance between the AP and the mobile device. However, while discussing the more appropriate configuration for the AP in a house, it is important to analyze how different parameter settings at the MAC layer would couple with different distances between the AP and the mobile device. To this aim, Figure 5 shows that, positioning the mobile device at a distance of 5m from the AP, the maximum goodput achievable by the FTP application is already obtained when utilizing only two retransmissions at the MAC layer. With a distance of 10m between the AP and the mobile node, the goodput of the FTP application decreases when utilizing lower values for the maximum number of MAC layer retransmissions. However, the reduction in terms of goodput is limited when setting the maximum number of retransmissions to 3 and could be considered an acceptable sacrifice when i) it is limited to a case where the AP and the mobile node are far from each other and ii) it improves the jitter experienced by real-time applications.

C. Limiting buffer size and retransmissions at the MAC layer

Here we analyze the impact that different buffer sizes and maximum number of retransmissions at the MAC layer have on performances of the different types of traffic. Focusing on the first parameter, Figure 6 and Figure 7 confirm that having a larger buffer size helps TCP in achieving a higher goodput. Obviously, there is no goodput difference when wireless losses, not recovered by MAC retransmissions, are frequent enough to keep the TCP transmission rates low and hence never have the possibility to utilize more than 50 buffer slots.

On the other hand, having large buffers along the path may augment the total delay time experienced by packets. In fact, each packet waits in queue for a time which proportionally grows with the number of packets already present in the same queue at its arrival. In case of intense traffic, buffers tend to be congested and hence queuing delays may become a significant component of the global delays experienced by each packet. At the same time, having larger buffer size on a link also spreads the range of possible queuing delays that traveling packets may experience on that link (depending on the filling level of the buffer). The resulting jitter strongly impact on the performance achieved by real-time applications and, in particular, by highly interactive applications such as videochat and online games. Statistics of the aforementioned game flow jitters permit a clearer understanding of the performance disparity generated by diverse buffer sizes. Specifically, Table IV shows the average, the variance, and the maximum jitter value experienced by online game packets while the TCP-based FTP flow was active (i.e., from second 135 to second 180).

Online gaming applications can bear a maximum delay of 150ms between the generation of an event and its propagation to all the players in the Internet. A scalable architecture exploiting mirrored game servers and an efficient synchronization scheme to uplift the interactivity level has been proposed in [12]. However, the efforts in having a fast synchronization among mirrored game servers may be wasted if the client-server part of the connection introduces exag-

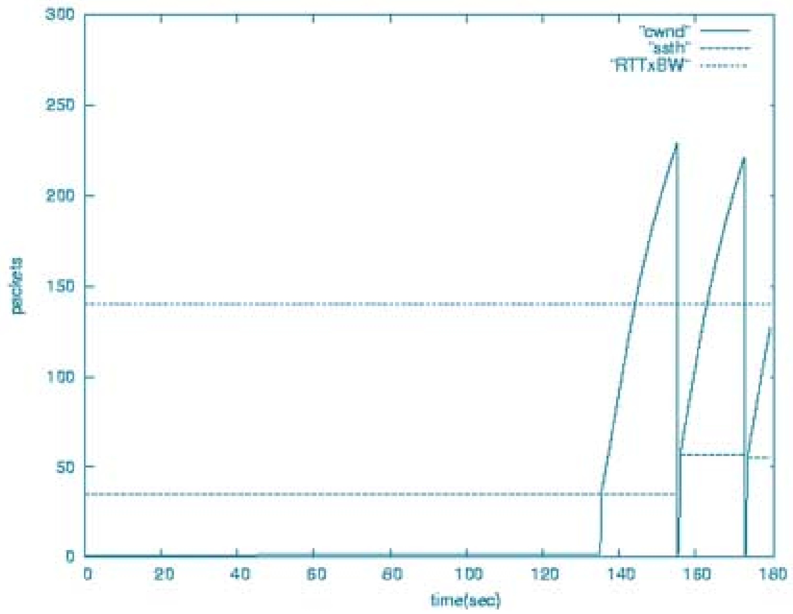

Fig.4 Regular TCP's congestion window; transmissions starting at 135s

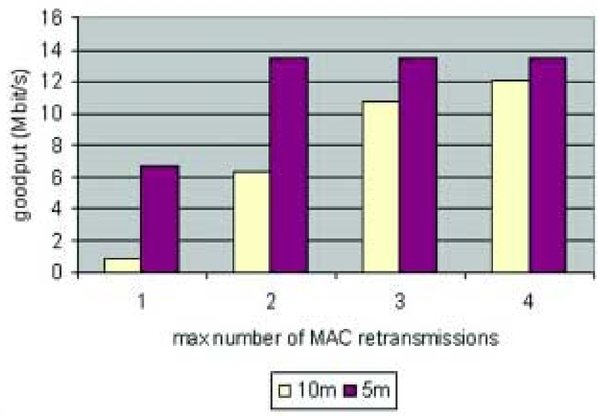

Fig.5 FTP average goodput with different user-AP distances

gerate delays, as may happen with larger MAC layer buffer size.

In any case, the worst jitter is experienced when buffers are steadily filled up by the FTP flow. One possible solution to limit this problem is that of bartering part of the FTP goodput with lower queuing delays. This tradeoff can be found by simply reducing the maximum number of retransmission at the MAC layer. Renouncing to the forth possible retransmission could bring the delays benefits evidenced in Table V; while the TCP goodput will remain almost unaltered as shown in Figure 5.

Even better jitter average and variance can be gained further diminishing the maximum number of MAC retransmission to 2. However, we advice against this choice, unless placing the device hosting the FTP application closer to the AP or in a house with a better shadowing deviation of the transmitted signal. Otherwise, the FTP throughput descends significantly as can be observed in Figure 5.

Summarizing, we can say that a more appropriate configuration of the IEEE802.11g than the traditional one would probably make use of a maximum number of 3 retransmissions, thus guaranteeing a high FTP throughput while maintaining a low per-packet delay and jitter. Moreover, when a unique queue is maintained for all the traffic flows, a small size (50 packets) should be preferred.

D. Summarizing results

We summarize statistical results obtained by: i) utilizing a standard IEEE 802.11g MAC configuration (*Regular*) and ii) appropriately setting the MAC layer parameters (*MAC-Setting*). Specifically, the MAC

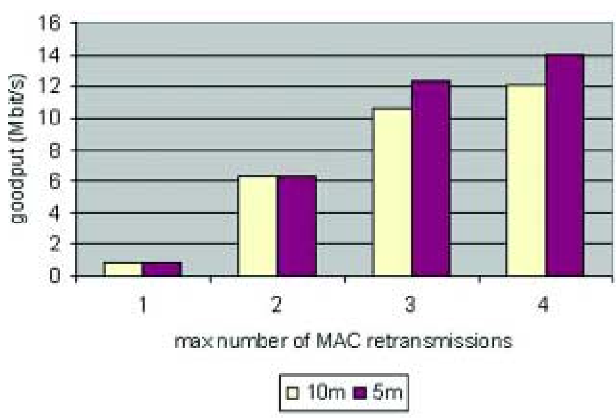

Fig.6 FTP average goodput with different MAC buffer sizes; distance from the AP = 10m

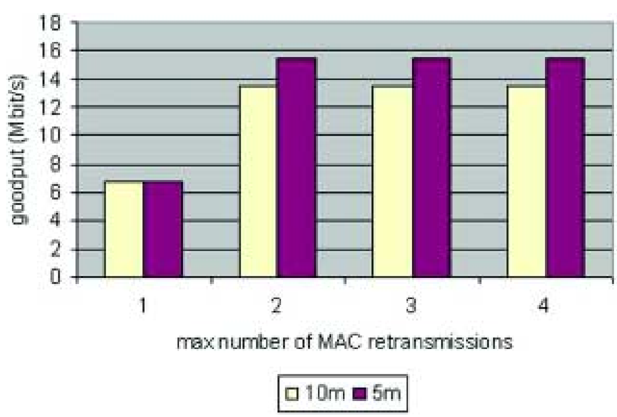

Fig.7 FTP average goodput with different MAC buffer sizes; distance from the AP = 5m.

layer parameters for MAC-Setting were set with a maximum number of retransmissions equal to 3 and

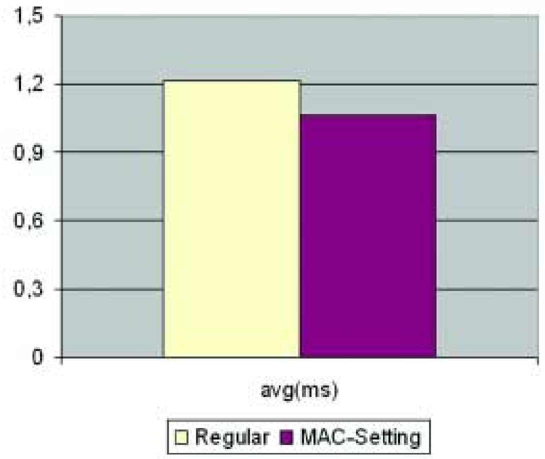

Fig.8 Average online game jitter

a buffer size of 50 packets at the AP. The distance between the AP and the mobile devices was 10m and

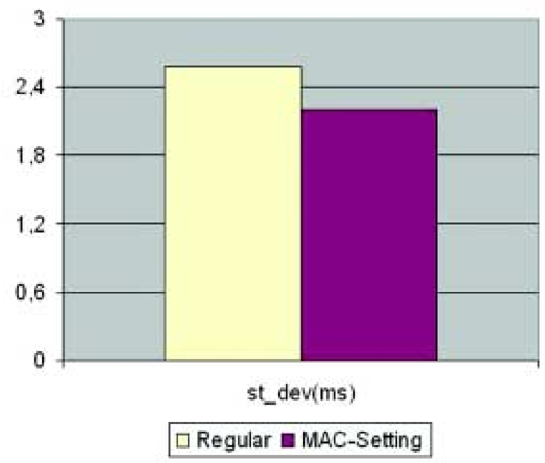

Fig.9 Standard deviation of the online game jittera

the shadowing deviation value was set to 9.

The compared statistical parameters are the average (Figure 8), the standard deviation (Figure 9), and the maximum value (Figure 10) of jitter experienced by online game packets going from the server to the client via the AP. Results obtained from the other real-time applications running in the simulated scenario (i.e. video-stream and video-chat) are coherent with the showed ones; we hence skip to present their charts. Rather, we also show the average goodput achieved by

Table.4 Gaming Flow Jitter: Statistics For Various Mac Layer Buffer Sizes; Considered Period = [135-180S], Max Mac Retr = 4

Jitter	50 pkts	100 pkts
maximum (ms)	33.740	108.36
average (ms)	3.056	5.229
variance	16.665	49.470

Jitter	50 pkts	100 pkts
maximum (ms)	31.091	44.632
average (ms)	2.292	3.835
variance	11.502	24.431

Table.5 Gaming Flow Jitter: Statistics For Various Mac Layer Buffer Sizes; Considered Period = [135-180S], Max Mac Retr = 3

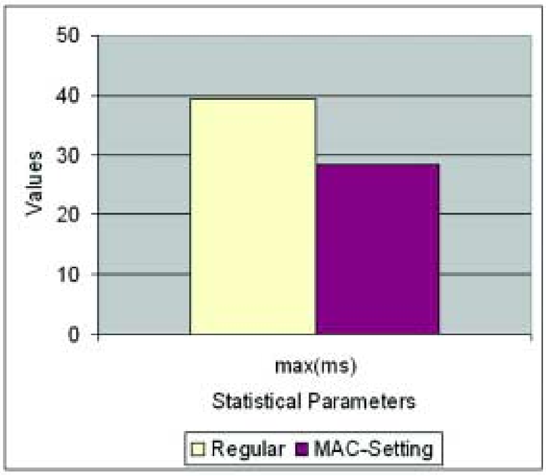

Fig.10 Maximum online game jitter

the concurrent TCP connection (Figure 11).

As it is evident, in a wireless home scenario, a better configuration of MAC layer parameters is desirable in order to support real-time applications' performances even if at the cost of reducing the goodput achieved by downloading applications. MAC layer parameters could hence be set by AP's manufacturers and sold to customers as a product specifically designed to sup-

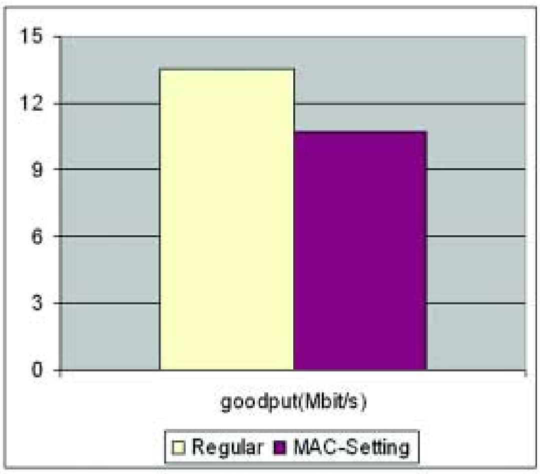

Fig.11 FTP average goodput

port multimedia entertainment in wireless homes.

Finally, the proposed setting does not compromise the possibility of applying other techniques designed to support real-time traffic in a wireless environment. In particular, the proposed configuration for the MAC layer can perfectly coexist with other solutions, such as priority queues [13]. The resulting combination would produce even better performances than those achievable by singularly employing these techniques.

VI. CONCLUSIONS

In this work, we presented networking issues when digital entertainment is delivered to wireless devices in a house Some popular networking entertainment applications are considered and the impact of the underlying wireless technology has been extensively investigated. In particular, we have shown how even a single persistent TCP-based flow can deteriorate real-time applications' performances. This constitutes the reverse of the well known argument by which UDP's lack of congestion control would harm TCP, whereas we have demonstrated how the TCP's lack of buffering control is harmful as well toward UDP-based applications.

This problem has been investigated by analyzing outcomes with different scenario's parameters. We have concluded presenting a more appropriate tuning of the IEEE802.11 parameters that would allow a more equilibrated coexistence between downloading and real-time applications.

REFERENCES

[1] S. Moyer, D. Marples, S. Tsang, and A. Ghosh, “Service portability of networked appliances,” 2002. [Online]. Available: citeseer.ist.psu.edu/ moyer00service.html

[2] A. Mani, H. Sundaram, D. Birchfield, and G. Qian, “The networked home as a user-centric multimedia system,” in NRBC '04: Proceedings of the 2004 ACM workshop on Next-generation residential broadband challenges. New York, NY, USA: ACM Press, 2004, pp. 19 0.

[3] G. Bell and J. Gemmell, “A call for the home media network,” Commun. ACM, vol. 45, no. 7, pp. 71 5, 2002.

[4] Y.-M. Wang, W. Russell, A. Arora, R. K., Jagannathan, and J. Xu, “Towards dependable home networking: An experience report,” in Proc. IEEE nternational Conference on Dependable Systems and Networks (DSN 2000), New Yor, USA, June 2000, pp. 43-49.

[5] IEEE, “Standard for wireless lan medium ccess control (mac) and physical layer (phy) specifications,” specifications, ISO/IEC 8802-11:1999(E), 1999.

[6] G. Huston, “Tcp in a wireless world,” IEEE Internet Computing, vol. 5, no. 2, pp. 82-84, 2001.

[7] C. H. Nam, S. C. Liew, and C. P. Fu, “An experimental study of ARQ protocol in 802.11b wireless LAN,” in In Proc. of Wireless Personal Multimedia Communications (WPMC'02), 2002.

[8] M. Gerla, K. Tang, and R. Bagrodia, “Tcp performance in wireless multi-hop networks,” in WMCSA '99: Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications. Washington, DC, USA: IEEE Computer Society, 1999, p. 41.

[9] G. Xylomenos and G. C. Polyzos, “ TCP and UDP performance over a wireless LAN,” in INFOCOM (2), 1999, pp. 439-46. [Online]. Available: citeseer.ist.psu.edu/xylomenos99tcp.html

[10] (2005) Movie trace files. [Online]. Available: http://www-tkn.ee. tu-berlin.de/research/trace/ltvt.html

[11] J. Farber, “ Traffic modelling for fast action network games,” Multimedia Tools Appl., vol. 23, no. 1, pp. 31-46, 2004.

[12] C. E. Palazzi, S. Cacciaguerra, S. Ferretti, and M. Roccetti, “On maintaining interactivity in event delivery synchronization for mirrored game architectures,” in 1st IEEE International Workshop on Networking Issues in Multimedia Entertainment (NIME'04), GLOBECOM 2004, Dallas, TX,, 2004.

[13] IEEE, “ Unapproved draft amendment standard for information technology-telecommunications and information exchange between systems-lan/man specific requirements-part 11 wireless medium access control (mac) and physical layer (phy) specifications: Medium access control (mac) quality of service (qos) enhancements,” specifications, ISO/IEC 8802-11:2005 (E), 2005.