Wireless Netw (2010) 16:331–353
DOI 10.1007/s11276-008-0132-3

A new bandwidth allocation mechanism for next generation wireless cellular networks

Polychronis Koutsakis · Moisis Vafiadis · Aggelos Lazaris

Published online: 30 August 2008

Abstract In this paper we design and study the performance of a Medium Access Control (MAC) scheme for the multiplexing and the integrated delivery of voice, mobile messaging, IP, gaming and H.264 videoconference traffic over a high-speed cellular TDMA channel with errors and capture. To the best of our knowledge, this is one of the first papers in the literature investigating the integration of actual H.264 video traces and gaming traffic with other types of traffic over wireless networks. Our results show that the proposed scheme achieves high throughput results while preserving the strict Quality of Service (QoS) requirements of each traffic type, and outperforms two efficient schemes previously proposed in the literature.

Keywords Multiple Access Control · Call admission control · Cellular networks · Multimedia traffic · H.264video · Gaming traffic

1 Introduction

Emerging wireless networks aim to satisfy the goal of incorporating and supporting a large variety of multimedia applications, which demand significantly enhanced transmission rates in comparison to the ones achieved today. Fourth generation mobile data transmission rates are planned to be up to 20 Mbps [1–3]. In this work, we study a wireless channel of this maximum expected transmission rate.

The issues of QoS such as perceived data rate, delay for message delivery, and high system costs are yet to be addressed completely in third generation wireless networks. Interconnection of wireless cellular networks of different standards with the wired networks and with other types of wireless networks such as satellite networks in an efficient and cost-effective way calls for a new generation of wireless networks.

More specifically, fourth generation wireless networks will be able to provide global seamless roaming between heterogeneous wireless and wired networks, interworking units between networks of different standards (2G, 3G, and 4G) and between satellite and terrestrial wireless networks, hence significantly improving the roaming capability of previous generation networks at an affordable cost and increased QoS.

An efficient MAC protocol can play an important role towards the above goals, by exploiting the variations in access and service required by disparate sources and therefore reducing system costs by maximizing system capacity, while integrating different classes of traffic.

In this work, we design a MAC protocol which supports multimedia traffic access to the wireless medium, based on a Time Division Multiple Access (TDMA) scheme. TDMA protocols can be generally categorized according to their duplexing technique, i.e., as TDMA with Time Division Duplex (TDMA-TDD) or TDMA with Frequency Division Duplex (TDMA-FDD) [4].

The literature in both categories of protocols is equally vast (e.g., [2, 3, 5, 6] for TDMA-TDD, [7–10] for

P. Koutsakis (✉)
Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada
e-mail: polk@ece.mcmaster.ca

M. Vafiadis · A. Lazaris
Department of Electronic and Computer Engineering, Technical University of Crete, Chania, Greece
e-mail: moisisv@telecom.tuc.gr

A. Lazaris
e-mail: alazaris@telecom.tuc.gr

TDMA-FDD). TDMA-FDD techniques need two frequency carriers (as opposed to TDMA-TDD techniques which need only one) but they provide a faster method to determine if retransmission is necessary [4]. For this reason, we have adopted the TDMA-FDD technique in our scheme.

More specifically, in this work, we design a MAC scheme which supports multimedia traffic access to a very high-capacity wireless channel with errors and capture. Within the picocell, spatially dispersed source terminals share a radio channel that connects them to a fixed Base Station (BS). The BS allocates channel resources, delivers feedback information and serves as an interface to the Mobile Switching Center(MSC). The MSC provides access to the fixed network infrastructure. Since the BS is the sole transmitter on the downlink channel, it is in complete control of the downstream traffic, using TDMA to relay information to the users. Hence, we focus in this work on the uplink (wireless terminals to BS) channel, where a MAC scheme is required in order to resolve the source terminals' contention for channel access. Our scheme multiplexes voice traffic at the vocal activity (talkspurt) level to efficiently integrate voice (Constant Bit Rate, CBR On/Off Traffic), H.264 videoconference traffic (Variable Bit Rate, VBR), traffic originating from network games (VBR) and bursty data traffic (mobile messaging (SMS), as well as IP traffic) in high capacity picocellular systems. By referring to "data" throughout the paper we refer to both the aforementioned types of data traffic. Our scheme is compared to two other efficient schemes of the relevant literature [7, 9] and is shown to clearly excel in terms of channel throughput and transmission delays for all types of traffic.

The paper is organized as follows. Section 2 presents, in brief, the contribution of this work. Sections 3 and 4 refer to the error and capture models adopted. Section 5 explains the Base Station Scheduling policy proposed in this work and implemented in our scheme, and Sect. 6 lists the system parameters. Section 7 presents: (a) the traffic types and models used in our study, (b) the channel frame structure and the transmission protocols, (c) the two MAC protocols with which we compare our scheme and (d) our simulation results along with a detailed discussion on them.

2 Contribution of this work

In recent work [11] we have designed the MI-MAC (Multimedia Integration Multiple Access Control) protocol. MI-MAC was one of the first protocols in the relevant literature to consider the integration of actual MPEG-4 or H.263 video streams with other types of packet traffic (voice, email, web). The protocol was shown to be a good candidate for next generation cellular networks, as it outperformed (in simulation results and conceptually, respectively) a well-known TDMA-based protocol and three WCDMA-based protocols when evaluated over a wireless channel with burst-error characteristics. However, certain design limitations had been adopted in the protocol's study (this will be further explained in Sects. 5.1 and 7.2). Although the assumptions behind these limitations were reasonable and were used to facilitate the evaluation and comparison of MI-MAC with other protocols in the literature, they still need to be waived in order to design a protocol for a more realistic wireless cellular network scenario. This is done in the protocol which is proposed in the present work, by introducing new ideas regarding the channel frame structure and the Base Station Scheduling. Additionally to these ideas, which help us enrich our former work, two more ideas are proposed in this work. One is a new scheduling idea for voice preemption in favor of video traffic and the second is the use of H.264 video traffic modeling for efficient call admission control at the network entrance. As it will be explained in Sect. 5 and shown by our results in Sect. 7, the use of accurate video traffic modeling based on actual video traces (for any encoding of video traffic) for call admission control is of significant importance for system stability, when video sources are integrated with other traffic sources over a cellular network. All the above-mentioned ideas will be explained in detail in Sects. 5 and 7. Finally, in this work we also consider the capture effect, which was not considered in MI-MAC, and we focus on a 20 Mbps channel, whereas MI-MAC was implemented on a 9.045 Mbps channel. Our results show that the proposed scheme achieves high throughput results while preserving the strict Quality of Service (QoS) requirements of each traffic type; therefore, we believe that our scheme's ideas can be very useful in the MAC protocol design for next generation, multimedia-serving, wireless cellular networks.

3 Channel error model

It has been shown by several authors, for many different types of wireless channels, that the wireless channel can be modeled as a finite state Discrete Time Markov Chain (DTMC) [12]. We adopt again, as in [11], the robust error model for wireless channels presented in [13]; the 3-state (Good (G), Short-Bad (SB) and Long-Bad (LB) states, respectively) DTMC model makes more accurate predictions of the long-term correlation of wireless channel errors than the two-state Gilbert-Elliot model (the Markov chain is shown in Fig. 1). A transmission is successful only if the channel is in the "good state"; otherwise it fails. The difference between the LB and SB states is the time correlation of errors: LB corresponds to long bursts of

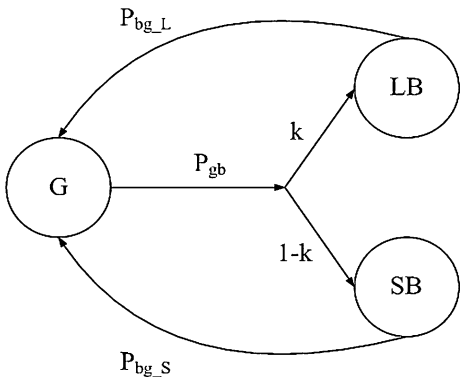

Fig. 1 Channel error model

Table 1 Error model parameters

$P_{good} = 0.99992$
$B_G = 1/p_{gb} = 65160$ slots
$B_{SB} = 1/p_{bg\_S} = 2.38$ slots
$B_{LB} = 1/p_{bg\_L} = 59.53$ slots
$k = 0.05$

errors (e.g., a cellular user is in an area where the signal is weak), SB to short ones (e.g., a user experiences a temporary bad signal). The parameters of the error model are presented in Table 1. The average number of error bursts, in slots, experienced when the states LB and SB are entered, are respectively given by: $B_{LB} = 1/p_{bg\_L}$ and $B_{SB} = 1/p_{bg\_S}$, where p_{bg_S} is the transition probability from state SB to G, and p_{bg_L} is the transition probability from state LB to G. Similarly, the average number of consecutive error-free slots is given by $B_G = 1/p_{gb}$, where p_{gb} is the probability to leave state G. The parameter k is the probability that the Markov chain moves to state LB given that it leaves state G; k also represents the probability that an error burst is long (i.e., the fraction of long bursts over the total number of error bursts). We have chosen in our study the value of the probability P_{bad}, i.e., the steady-state probability that the channel is in bad state, to be equal to $8*10^{-5}$; this value has been chosen in order to test an "almost worst" case scenario for our system, as the video packet dropping probability is set to 10^{-4} and, by choosing a value of bad state probability larger than the upper bound on video packet dropping, the strict QoS requirement of video users would certainly be violated. The values for p_{gb} and for the parameter k have been taken from [13], as well as the ratio between p_{bg_s} and p_{bg_L}. The value for p_{bg_L} is derived from the steady-state behavior of the Markov chain, for the bad state probability chosen. The fact that the total probability of a transmission error is only slightly smaller than the acceptable video packet dropping probability makes the need for very efficient scheduling imperative.

4 Capture

The worst case assumption is that a collision will occur whenever two or more request packets arrive simultaneously at the BS. In practical systems, the mobile terminals are dispersed throughout the picocell and packets may arrive at the BS with significantly different power levels (unless some form of adaptive power control is employed). This difference in received energy may be caused either by terminals transmitting with different power levels, or by the different propagation characteristics caused by terminals transmitting from varying distances and/or over dissimilar transmission paths. It is probable that the request packet with the stronger signal will be successfully received (*captured*) by the BS [14].

In our study, the probability of a successful contention depends upon the total number of contenders simultaneously transmitting their requests and is governed by the capture model presented in [15–17], which specifies the probability of success as a function of the number of contenders (an example is given in Table 2).

In the event that a success occurs when more than one terminal transmit in the same slot, the successful terminal is chosen at random among the contenders. The successful terminal then joins the ready for service queue and awaits its turn for service.

We have used the same capture model in our results for the two protocols [7, 9] with which we compare our work, in Sect. 7.

5 Actions of terminals, base station scheduling and call admission control

5.1 Actions of terminals and base station scheduling

Video terminals have highest priority in acquiring the slots they demand. This choice will be explained later in this section. If a full allocation is possible (after the end of the *request interval*, which we will discuss in Sect. 7.2), the BS then proceeds to allocate any still available information

Table 2 Capture model

Number of contenders	Probability of success
0	0.00
1	1.00
2	0.67
3	0.48
4	0.40
5	0.35
>5	0.00

slots to the requesting voice terminals, then to requesting gaming terminals and finally to data terminals. In the case that a new VF arrives and the number of data and gaming reservations is such that the video terminal can not be fully serviced, the BS *preempts* data and gaming reservations (in this order, as data traffic is more delay-tolerant) in favor of video terminals waiting for transmission. If a full allocation to the video terminals is still not possible, the BS grants to video users as many of the slots they requested as possible (i.e., the BS makes a partial allocation); it should be mentioned here that this policy serves in minimizing the *average* video packet dropping over all video terminals, which is the goal in this work, but it might not be the best choice in a practical system if the video terminal which receives the partial allocation cannot acquire the slots it needs before the arrival of its next VF; in other words, if QoS is considered for each individual terminal (this will be the subject of our future work), it may be preferable in certain cases to avoid a partial allocation when the video terminal's QoS requirements cannot be met for a significant time period, and use the remaining bandwidth to expedite the transmission of packets from other video terminals or from voice terminals, which also have strict QoS requirements. The BS allocates the earliest available information slots to the video terminals, which, if needed, keep these slots in the following channel frames, until the next VF arrives.

In order to provide better service to voice users, the BS also "preempts" data (first SMS, which is more delay-tolerant and then, if needed, IP) and gaming reservations whenever new voice requests are received and every slot within the frame is reserved. When data and gaming reservations are canceled, the BS notifies the affected data/gaming terminal and places an appropriate request at the front of the data or gaming request queue.

Voice terminals which have successfully transmitted their request packets *do not* acquire all the available (after the servicing of video terminals) information slots in the frame. If this happened, voice terminals would keep their dedicated slots for the whole duration of their talkspurt (1 s on average, i.e., more than 80 channel frames), and thus video terminals would not find enough slots to transmit in; hence, the particularly strict video QoS requirements would be violated. For this reason, we utilize again an idea which we used in MI-MAC, i.e., that *the BS allocates a slot to each requesting voice terminal with a probability p**. In MI-MAC a near-optimal value of p^*, equal to 9% was found through extensive simulations; this value worked well for all video loads examined. The respective near-optimal value for this work was found to be equal to 7.5%, however all values between (7–10%) give similar results. The requests of voice terminals which "fail" to acquire a slot, based on the above BS slot allocation policy, remain queued. The same holds for the case when the resources needed to satisfy a voice request are unavailable.

Within each priority class, the queuing discipline is assumed to be First Come First Served (FCFS).

After implementing the above ideas, we found from our initial results that there were cases where, although it was expected that the system would be able to accommodate the offered traffic, the very high burstiness of video traffic (peak/mean ratio ranges from 8 to 17.6 for the traces under study in our work) still led to a violation of the very strict video QoS requirement of maximum 0.01% packet dropping [7]. For this reason, the following additional idea was introduced to our mechanism, in order to provide video users with the extra bandwidth needed to satisfy their strict QoS requirements.

Our idea is based on the dynamic preemption of *voice* users by *video* users, and operates as follows: for a given number of video users present in the system, the BS starts to preempt voice users in favor of video users *after a specific point (slot) in the channel frame*. Simulations have shown that if voice preemption started at the beginning of the channel frame (immediately after the first request interval, as is the case for *data and gaming users' preemption*), the voice packet dropping probability surpassed the upper bound of 1% very quickly, for a relatively small number of voice terminals. The point where voice preemption needs to start depends on the number of video users in the system, but *does not need to be predefined* by the network administrator. The BS only needs to have a very rough estimation of the voice preemption start point depending on the number of video users (i.e., preemption starts at 70% of the frame when 7 or less video users are present in the system and at 40% of the frame when more than 7 video users are present in the system). The BS, subsequently, receives feedback from the video terminals regarding their packet dropping and, every 100 frames (1.2 s, found to be an adequate amount of time via simulation) computes the average video packet dropping. If it is lower than the upper bound of 10^{-4}, the voice preemption start point "moves" by 1% to the right (i.e., the preemption starts later in the frame, in order to better facilitate voice access). If it is higher than the upper bound, the voice preemption start point "moves" by 5% to the left (i.e., the preemption starts earlier in the frame, in order to better facilitate video access). The reason for the higher "jumps" of the voice preemption start point in the second case is the very strict video QoS requirement regarding packet dropping, which needs to be satisfied.

With the use of this mechanism, the BS is able to find in a few "steps" the ideal preemption start point, and therefore accommodate the bursty video traffic with a close to minimum deterioration on the QoS of voice sources.

As mentioned at the beginning of this section, the rationale of our choice for offering highest priority needs to be explained, especially since the vast majority of relevant papers in the literature offer highest priority to voice traffic.

There are quite a few papers in the literature which use, as an upper bound for video packet dropping, values much larger than the one we used in our work (we used 0.01%, these papers use values between 1 and 3%) while other papers use values that range from 0.001% to 0.1%. The first reason why we chose to give priority to video traffic is that both papers with which we compare our work use very low values as an upper bound for video packet dropping. More specifically, in [7] the upper bound was 0.01% and in [9] it was 0.001%. The authors in [9] were able to offer a higher, on average, transmission probability to voice users than to video users by (a) using a much larger upper bound for the delay that a video packet can endure before it is dropped and (b) considering a non-bursty video (exact information on the peak of the video trace used in that work is not available in [9], however the fact that the average video rate is 1.5 Mbps, that the total channel rate is 5.3 Mbps and that a zero video packet dropping can be satisfied if 2.01 Mbps (57 slots out of the 150 in each frame) is steadily offered to video traffic, leads us to understand that the peak rate does not surpass 2 Mbps). Similarly, the video traffic model used in [7] (not an actual trace) creates video traffic with an average of 1 Mbps and peak of 2.1 Mbps, therefore again the authors were discussing a much less bursty video source than the ones used in our study. Hence, we wanted to make a fair comparison with those schemes, using a low upper bound on video packet dropping. Given that the upper bound on voice packet dropping is 100 times larger (1%, used vastly in the literature), and that video traffic in our study (and in today's networks, in general) is very bursty, we were led to offer full priority to video traffic.

The second, and most important reason for this choice, was the "tenacious" behavior of voice traffic on which we already commented. Voice terminals which pass the contention phase successfully keep their dedicated slots for the whole duration of their talkspurt and therefore video terminals cannot find enough slots to transmit in, resulting in a violation of the strict video QoS requirements. Hence, by offering highest priority to video traffic and using the p^* and voice preemption policies we are taking advantage of the fact that voice traffic can endure a 100 times larger dropping probability than video, and we are "pushing" the QoS of voice users to the acceptable limit, in order to be able to accommodate the bursty video traffic.

Finally, as already mentioned in Sect. 2, certain design limitations had been adopted in [11], in the MI-MAC protocol study. Two of these limitations will be presented in Sect. 7.2, as they are connected to the channel frame structure. The other two limitations, which are connected to the Base Station Scheduling, are the following:

(a) Since MI-MAC was evaluated over one cell of the network, no traffic was considered to be arriving from other cells (handoff traffic); in this work we consider the cases of various volumes of handoff traffic arriving in the cell.

(b) Since a picocellular wireless cellular architecture was assumed (picocell radius 10–50 m), the assumption was made that all users perceived the same uplink channel condition; in reality, however small the picocell radius, the channel fading experienced by each user is different, since users are moving independently of each other; therefore, each user perceives a different wireless channel condition. Also, in [11], errors were considered only in the uplink channel. In the present work fading *per user channel* is considered for both the uplink channel and the part of the downlink channel corresponding to the uplink reservation request interval (i.e., the part of the downlink channel in which users learn about the outcome of their reservation request transmissions, and therefore may need to retransmit their requests); if a user's downlink channel is in bad state when the user is waiting for BS feedback on the outcome of its request transmission, the user considers that the outcome has been a collision and retransmits its request packet. This results in increased contention, which aggravates system performance and makes the need of an efficient scheduling mechanism even more imminent.

In order to waive the above two limitations, we introduce the following ideas, respectively, in our scheme.

(a) It is a common assumption in the literature that the dissatisfaction of a wireless cellular subscriber who experiences forced call termination while moving between picocells is higher than that of a subscriber who attempts to access the network for the first time and experiences call blocking; for this reason we offer *full priority* to handoff traffic. This means that video terminals who have been hand-offed to the cell are the first to attempt to transmit their requests to the BS; when their contention is finished, they are followed by hand-offed voice terminals, then by hand-offed gaming terminals, hand-offed IP terminals and finally by hand-offed SMS terminals. Traffic originating from within the cell follows in priority, in the same order. As it will be explained in Sect. 7.2, the above prioritization by "isolating" each type of traffic and letting it contend only with traffic of the same type is feasible due to the use of the *two-cell stack* reservation random access algorithm (by video and voice terminals) and the *two-cell stack* blocked access collision resolution algorithm (by gaming and data terminals) in order to resolve contention.

Therefore, the impact of taking into account handoff traffic in the system is that system performance is significantly aggravated (in comparison to MI-MAC) as the percentage of traffic which belongs to hand-offed terminals

needs to acquire again the QoS experienced in the previous cell.

(b) We introduce the idea that the system *should take advantage* of the "problem" created when a video user experiences a "bad" channel state (error burst) and is unable to transmit in its allocated uplink slots; this would normally lead to the dropping of the video packets scheduled to be transmitted in these slots, and consequently to higher average video packet dropping probability and the system's failure to satisfy the strict QoS requirements of video traffic. Our new proposed mechanism aims at allocating as many of these slots (defined as "*abandoned slots*" in the rest of the paper) as possible to other video terminals awaiting for packet transmission, in order to decrease their transmission delay. Our choice of not allocating the abandoned slots to voice terminals is based, once again, on the fact that voice users which enter the system will remain in talkspurt for more than 80 channel frames, hence hindering access to these slots for video terminals which have stricter QoS requirements. The abandoned slots can be allocated to gaming or data terminals in order to expedite their information transmission and increase channel throughput, but *only for the current channel frame and only if no other video terminals are awaiting slot allocation*, so that video QoS will not be influenced by the better servicing of gaming and data traffic.

Although conceptually simple, the above approach is not equally simple to implement. The quality of each user's channel can be indicated by the signal-to-noise ratio (SNR) function; as shown in [18], in a FDD system (such as ours, which is a TDMA-FDD one) using pilot symbols that are inserted in the downlink with a certain time-frequency pattern, the mobile terminals can effectively estimate their SNR function and send it to the BS, which can then make its scheduling decisions based on all the collected cross-layer information from the terminals. This process, however, introduces both errors and delays in the estimates. Due to the random nature of the channel, it is impossible for the BS to precisely determine the state of the channel. The best estimate a BS can provide is a probability distribution over the possible channel states [12], which is our assumption in this work, i.e., that the probabilities of the Markov chain model have been derived with the above procedure.

Still, even though the probabilities of our model are known to the BS, the BS cannot know with certainty the type of channel state transition that takes place for a mobile terminal when it leaves the good state, i.e., if the terminal's channel has entered the SB state or the LB state. According to the channel error model, the mean duration of SB state is 2.38 slots, while the mean duration of LB state is 59.53 slots. Therefore, in our scheme, we propose that the *BS estimates each mobile video terminal's precise channel conditions by monitoring the slots allocated to the terminal and checking whether the terminal is transmitting in them or not*. If the total number of a terminal's failed transmissions within its allocated slots surpasses a given threshold, the BS in our scheme deduces that the terminal is in LB state, as the probability that it is in SB is very small given the high number of corrupted transmissions. Based on the channel error model's parameters, it is easy to confirm by both analysis and simulation that the probability that a mobile terminal's channel is in SB when more than 6 slots have been wasted is 6.55%; hence we have set the threshold to be 6 consecutive transmission failures (choosing a higher threshold would result in a more accurate prediction of the channel condition, as the probability of a mistake in the prediction would be significantly lower; however, it would also lead to a higher number of lost slots while the BS is awaiting to make that prediction). When the BS determines that a mobile video terminal is in LB state, if that terminal still has more reserved slots in the current channel frame, the BS deallocates these slots (i.e., these become the terminal's "abandoned slots"). The first of the abandoned slots is not allocated to other video terminals, as its corresponding downlink slot is used by the BS in order to inform video, gaming and data terminals which are queued and awaiting transmission of the new slots which are allocated to them (full priority is given to handoff video terminals, followed by video users originating from within the cell, then by hand-offed gaming users, hand-offed data users, gaming users originating from within the cell and finally by data traffic originating from within the cell; the allocation of the abandoned slots within each priority type is FCFS). When the channel of the mobile terminal to which the abandoned slots were originally allocated returns to the good state, the terminal needs to inform the BS of this change, if it still has packets to transmit. This is done by transmitting a request at the earliest of the following two opportunities: (a) in a minislot of an extra request slot which might exist in the current frame (request slots are divided into minislots and any free information slot of the current channel frame can be temporarily used as an extra request slot, as it will be explained in Sect. 7.2), or (b) in a minislot of a regular request interval, but within the minislots used for contention resolution of *handoff* video terminals, regardless of the call having been a handoff one or not. This means that highest priority is given to video users whose slots had been deallocated due to an LB state estimation, in order to avoid unnecessary further video packet dropping. Hence, these users can update the BS on their bandwidth requirements which may have changed during the LB state due to the arrival, for example, of a new video frame at the terminal's queue. The terminal has to follow the above procedure also in the case of a wrong estimation (i.e., if it was in SB

despite the long error burst). Therefore, in the (unlikely but not improbable) case of a wrong estimation, this does not influence the throughput achieved by MI-MAC in heavy traffic loads (slots are simply allocated to other users) but it results in an unnecessary increase of contention.

In the rare case when the abandoned slots of a video user (user A) are allocated to another video (user B), which in turn enters a bad state after a while, the abandoned slots are not further allocated to other users; they are "returned" to user A, in order not to increase system complexity with consequent "loans" of the same slots to multiple users.

5.2 Call admission control

As we discussed in [11], quite a few efficient call admission control (CAC) algorithms have been proposed in the literature for the transmission of voice, data and multimedia traffic over wireless networks. However, almost all of these mechanisms suffer, as noted by their authors, from a necessary conservatism in their estimation of the channel bandwidth consumed by the multiplexed sources, in order to preserve system stability and the users' QoS requirements. In [19] we reached the conclusion that the adoption of a CAC mechanism based on the very well-known equivalent bandwidth estimation method for the transmission of wireless videoconference traffic from MPEG-4 and H.263 video coders leads to significant throughput deterioration in comparison to the adoption of a strict traffic policing mechanism, i.e., traffic control is implemented much more effectively inside the system than at its entrance. For this reason, we combined in [11] the use of our MAC scheme with the use of a traffic policing scheme which we proposed in that work.

Based on our conclusions regarding call admission control, we proposed in [20] a new CAC mechanism for the transmission of H.263 videoconference traffic over wireless cellular networks. The basic idea of our mechanism is that, with the use of a model which we have built for H.263 videoconference traffic, a wireless provider should precompute the estimated traffic from various traffic scenarios which can take place in the network. This can be done based on the provider's client database and specifically on the traffic profile declared by each client in his contract with the provider. Therefore, many traffic scenarios which will be encountered in the network will be a priori known to the provider in terms of the actual network resources (bandwidth) which will be needed in order to satisfy the QoS requirements of video users. Certainly, not all traffic scenarios can be precomputed, due to the very large number of all possible traffic loads; however, as explained in [20], with the use of an adequate number of precomputed scenarios and our accurate video model, an online simulation can be quickly conducted when a non-precomputed traffic load occurs in the system, in order to compute the "deviation" between the bandwidth needed currently and the "closest" precomputed traffic scenario.

Similarly to [20], we have implemented a CAC scheme in this work, with the difference that this scheme is based on our H.264 traffic modeling, which will be presented in Sect. 7.1.5. The use of a CAC scheme is especially important, as it does not allow into the network traffic which, added to the already existing network traffic in the cell under study, would cause system instability and the violation of the QoS requirements of users already transmitting in that cell.

We believe that the use of accurate video traffic modeling based on actual video traces (for any encoding of video traffic) for call admission control is of major importance for system stability, when video sources are integrated with other traffic sources over a cellular network.

6 System parameters

We use computer simulations to study the performance of our scheme. The simulator is written in C programming language and simulations were conducted with the use of the system parameters taken from [11]. Each simulation point is the result of an average of 10 independent runs, each simulating 305000 frames (the first 5000 of which are used as warm-up period), i.e., one hour of network activity.

The channel rate is 20Mbps. The frame duration is chosen to be equal to the time a voice terminal needs to generate a new voice packet. For reasons of comparison with DPRMA and with [9], we consider that the information is transmitted in packets with length equal to the size of an ATM cell (53 bytes, 48 of which contain information). However, the nature of our results remains the same, regardless of the packet size, therefore the scheme could be used in any GSM-type network; different sizes have also been proposed in various works of the relevant literature for the transmission of data, both smaller (156 bits, [21]) and larger (200 bytes, [22]) than the ATM size used in our work. Assuming that the speech codec rate is 32 Kbps, yields the frame duration of 12 ms. The 12 ms of frame duration accommodate 566 slots. Consequently our channel's information payload rate is slightly lower than 18Mbps (taking into account the packet headers and the request slots). The high channel rate of 20 Mbps leads to a slot duration of only 0.021 ms. Various handoff traffic loads are considered, ranging from 5% to 15% of the total traffic offered in the network.

7 Simulation results and discussion

7.1 Traffic types and models

We consider multimedia terminals capable of transmitting voice, videoconferencing, gaming and data (SMS and IP) traffic. The traffic models and actual traces used in our study, when each terminal starts transmitting one of these types of traffic, follow.

7.1.1 Voice traffic model

Our primary voice traffic model assumptions are the following:

a. Voice terminals are equipped with a voice activity detector (VAD) [23]. Voice sources follow an alternating pattern of talkspurts and silence periods (on and off), and the output of the voice activity detector is modelled by a two-state discrete time Markov chain.
b. The average duration of the talkspurt period is 1.0 s and of the silence period 1.35 s [7, 9, 23].
c. The upper bound for the voice packet transmission delay is equal to 40 ms and the maximum allowed voice packet dropping probability is set to 0.01 [7, 9].

7.1.2 SMS traffic model

We adopt a Short Message Service (SMS) traffic model. SMS is a store-and-forward service that relies on a Short Message Service Center (SMSC). SMS messages are especially suitable for the transmission of small data bulks and for transmissions repeating in long time intervals (minutes to hours). The SMS payload is 140 bytes (including a header of 13 bytes), i.e, 3 packets of ATM size [24]. The message inter-arrival time distribution is considered exponential.

Estimations of GSM networks' SMS transmission delays refer to delays of 2–30 s, therefore in this work we set a strict upper bound of 2 s on SMS transmission delay.

7.1.3 IP traffic model

We adopt the traffic model presented in [25]. An internet connection realizing access to the World Wide Web (WWW) can be observed on different levels; session, page and packet level. Generally, a WWW page consists of a number of objects (different files, images, etc.), which are simultaneously transmitted during a page download. Thus, a first request for a WWW page, which is manually done by an internet user, causes a download of a main page object [26]. The main object is followed by a number of inline objects, which are automatically requested by a browser or just transmitted by an internet server as a logical succession of the main object.

The transmission of each page object causes establishment of a separate TCP connection. During a TCP connection, there is a data exchange between a transmitter (e.g. internet server) and a receiver (e.g. internet user), including transmission of user data and various control messages, such as TCP acknowledgements. The transmitted data units on the TCP level correspond to the IP packets (a TCP packet contains an IP packet plus TCP overhead). Computers and other communications devices operate with discrete sizes of the IP packets. In general, there are only few possible packet sizes between a minimum and a maximum value.

In [25], it is assumed that in the uplink transmission direction 85% of the packets are control and request IP packets. The remaining 15% are larger IP packets, caused by transmission of short messages, e-mail transfer, etc. The shorter IP packets can have two sizes, 64 and 256 bytes, which are generated with the probabilities 0.45 and 0.40, respectively. The maximum size of the IP packets specified in Ethernet LAN networks is about 1500 bytes and their probability in the model in [25] is set to 0.1. The probability of 1024 bytes packets, representing other larger packet transmissions, is 0.05. Therefore, the average IP packet size is 332.4 bytes (i.e., 7 packets of ATM size).

The IP packet interarrival time is modelled as a geometrically distributed random variable. As in the case of SMS traffic, a strict upper bound of 1 s has been set for the average IP packet transmission delay, in order to test our system's performance for very demanding users, in terms of required QoS.

7.1.4 Gaming traffic model

A quickly emerging application for 3G and 4G wireless systems is games. The characteristics of gaming traffic depend on many factors, such as game design, game style (e.g., fast-paced "shoot'em ups" or slow-paced strategic decision games), player experience and playing style [27]. Studies which have investigated gaming traffic [28, 29] have indicated that games have typically moderate bandwidth requirements. The delay requirements are very tight for fast-paced games (on the order of 100 ms), but looser for strategic games (on the order of 500 ms) [27].

In our work we focus on strategic games, since it is doubtful whether 3G or even 4G wireless cellular networks will be able to satisfy the delay requirements for fast-paced games. We use the model provided in [29] after an extensive analysis of the traffic generated by players of the popular real-time strategy game Starcraft. The authors in [29] model source traffic at the packet level and they provide models for the packet inter-arrival and inter-departure

times to and from the system, respectively, as well as for the packet sizes of inbound and outbound traffic. Since we focus in this work on the uplink channel, we use the models for the packet inter-departure times (*which actually are the traffic inter-arrival times for the transmitting gaming terminals in our system*) and for the packet sizes of outbound traffic. Regarding the packet sizes and in order to obtain analytical distributions approximating the empirical ones, the authors in [29] chose to split the distributions into a few parts to capture the behavior of a main peak and other lower peaks, which were fitted with deterministic distributions. More than 70% of the packets had a payload of 23 bytes, a behavior which is expected from a peer-to-peer game in which each player sends out multiple copies of its update packets to all other peers. The updates need to be small, to keep latency low, and frequent.

Table 3 presents the respective models. Regarding the packet inter-departure times, we observe from the Table that about two-thirds of the packets need a transmission time equal to zero, which in [29] corresponds to values $<$10 ms. Since no exact definition of the times was given in [29], we consider in our work that these packets have a deterministic inter-departure time of 5 ms. The packet sizes in the Table are expressed in bytes, and as already mentioned more than 70% of the packets have a payload of 23 bytes. Since all payload values are less than 48 bytes, all gaming packet sizes in our scheme are equal to 1 ATM packet. We set an upper bound of 500 ms on gaming packet transmission delay, and the allowed gaming packet dropping probability is 1% [27].

7.1.5 H.264 video traffic

7.1.5.1 The H.264 coding standard H.264 is the latest international video coding standard. It was jointly developed by the Video Coding Experts Group (VCEG) of the ITU-T and the Moving Picture Experts Group (MPEG) of ISO/IEC. It uses state-of-the-art coding tools and provides enhanced coding efficiency for a wide range of applications, including video telephony, video conferencing, TV, storage (DVD and/or hard disk based, especially high-definition DVD), streaming video, digital video authoring, digital cinema, and many others [30, 31].

7.1.5.2 H.264 video traffic modeling Similarly to our recent work on modeling H.263 [32] and MPEG-4 [33] videoconference traffic, we focused on the accurate fitting of the marginal (stationary) distribution of video frame sizes of single video traces. We will present in brief our modeling results in this section. These results, as explained in Sect. 5, can be used for efficient call admission control for video users.

Our work follows the steps of the work presented in [34], where Heyman et al. analyzed three videoconference sequences coded with a modified version of the H.261 video coding standard and two other coding schemes, similar to H.261. The authors in [34] found that the marginal distributions for all the sequences could be described by a gamma (or equivalently negative binomial) distribution.

In our work we have studied two different long sequences of H.264 VBR encoded videos in four formats from the publicly available Video Trace Library of [35], in order to derive a statistical model which fits well the real data. The selected videos are of low or moderate motion (i.e., traces with very similar characteristics to the ones of actual videoconference traffic).

The two traces are:

a. A demo from the Sony Digital Video Camera
b. An excerpt of NBC News

The length of the videos is 10 and 30 min, respectively. The data for each trace consists of a sequence of the number of cells per video frame and the type of video frame, i.e., I, P, or B. The interframe period is 33.3 ms. Table 4 presents the statistics for each trace. The statistics change minimally when altering the number of B frames in a GOP, therefore we could have used any version from the

Table 3 Starcraft game traffic model

	Model	Parameters
Packet inter-arrival times		(in seconds)
	Deterministic p = 66.2%	a = 0
	Uniform p = 27.8%	a = 0.05, b = 0.17
	Deterministic p = 6%	a = 0.21
Packet sizes		(in bytes)
	Deterministic p = 6.2%	a = 16
	Deterministic p = 10.9%	a = 17
	Deterministic p = 74.2%	a = 23
	Deterministic p = 8.7%	a = 27

Table 4 Video trace statistics (G: GoP Size, B: number of B frames, F: quantization parameters, where "22" stands for 22-22-24 I–P-B quantization and "28" stands for 28-28-30 I-P-B quantization)

Video name	[Resolution, G, B, F]	Mean (Mbps)	Peak (Mbps)	Standard deviation (Mbps)
NBC news	[CIF, 16, 3, 22]	1.09	8.69	1.16
NBC news	[CIF, 16, 3, 28]	0.44	5.48	0.65
Sony Demo	[CIF, 16, 3, 22]	0.86	10.51	1.43
Sony Demo	[CIF, 16, 3, 28]	0.38	6.69	0.84

online library (possible numbers of B frames = 1, 3, 7, 15). A newly arrived video user in the system is assigned one of the four traces with equal probability.

We have investigated the possibility of modeling the four traces with quite a few well-known distributions (gamma, lognormal, log-logistic, exponential, geometric, Weibull, Pearson V) and our results have shown that the best fit among these distributions is achieved for all the traces studied with the use of the Pearson type V distribution.

Our statistical tests were made with the use of Q-Q plots [36], Kolmogorov-Smirnov (KS) tests [36] and Kullback-Leibler divergence tests [37]. The Q-Q plot is a powerful goodness-of-fit test, which graphically compares two data sets in order to determine whether the data sets come from populations with a common distribution (if they do, the points of the plot should fall approximately along a 45-degree reference line). More specifically, a Q-Q plot is a plot of the quantiles of the data versus the quantiles of the fitted distribution (a z-quantile of X is any value x such that P ($(X \leq x) = z$). The Kolmogorov–Smirnov test (KS-test) tries to determine if two datasets differ significantly and has the advantage of making no assumption about the distribution of data, i.e., it is non-parametric and distribution free. The KS-test uses the maximum vertical deviation between the two curves as its statistic D. The Kullback-Leibler divergence test (KL-test) is a measure of the difference between two probability distributions.

Although the Pearson V distribution fit was shown to be the best in comparison to the other distributions, the fit was not perfectly accurate due to the gross differences in the number of bits required to represent I, P and B frames. Hence, we proceeded to study the frame size distribution for each of the three different video frame types (I, P, B), in the same way we studied the frame size distribution for the whole trace. This approach was also used in [38]. The mean, peak and variance of the video frame sizes for each video frame type (I, P and B) of each movie were taken again from [35] and the Pearson type V parameters are calculated based on the following formulas for the mean and variance of Pearson V (the parameters for the other fitting distributions are similarly obtained based on their respective formulas).

The Probability Density Function (PDF) of a Pearson V distribution with parameters (α, β) is $f(x) = [x^{-(\alpha+1)} e^{-\beta/x}] / [\beta^{-\alpha} \Gamma(\alpha)]$, for all $x > 0$, and zero otherwise.

The mean and variance are given by the following equations:

$$\text{Mean} = \beta/(\alpha - 1), \ \text{Variance} = \beta^2 \Big/ [(\alpha - 1)^2(\alpha - 2)]$$

The autocorrelation coefficient of lag-1 was also calculated for all types of video frames of the eighteen movies, as it shows the very high degree of correlation between successive frames of the same type. The autocorrelation coefficient of lag-1 will be used in order to build a Discrete Autoregressive Model for each video frame type.

From all the distributions examined the Pearson V distribution once again provided the best fitting results for the 12 cases studied (4 movies, 3 types of frames per movie). We present, indicatively, in Fig. 2 one of the Q-Q plot results. In order to further verify the validity of our results, we performed Kolmogorov-Smirnov and Kullback-Leibler tests for all the 12 fitting attempts. The results of our tests confirm our respective conclusions based on the Q-Q plots (i.e., the Pearson V distribution is the best fit). Indicatively, the K-L test results for the {I, P, B} frames of the Sony Demo ([CIF, G16, B3, F28]) trace are respectively, for the Pearson V distribution {0.35, 0.56, 1.39), for the Lognormal distribution {0.38, 0.64, 1.71}, for the Gamma distribution {0.39, 0.77, 2.71} and for the Weibull distribution {0.49, 0.77, 2.29}.

Although the fitting results when modeling each video frame type separately with the use of the Pearson V distribution were clearly better than the results produced by modeling the whole sequence uniformly, the high autocorrelation between frames of the same type (I, P, B) could not be perfectly "captured" by a distribution generating frame sizes independently, according to a declared mean and standard deviation, and therefore none of the fitting attempts (including the Pearson V), as good as they might be, could achieve perfect accuracy. However, these results led us to extend our work in order to build a DAR model (based on the Pearson V), which inherently uses the autocorrelation coefficient of lag-1 in its estimation. As it will be presented below, the model was shown to accurately capture the behavior of multiplexed H.264 videoconference movies, by generating frame sizes independently for I, P and B frames (*the case of modeling multiplexed video streams is especially significant for our resource allocation study, since numerous sources are multiplexed in the uplink channel*).

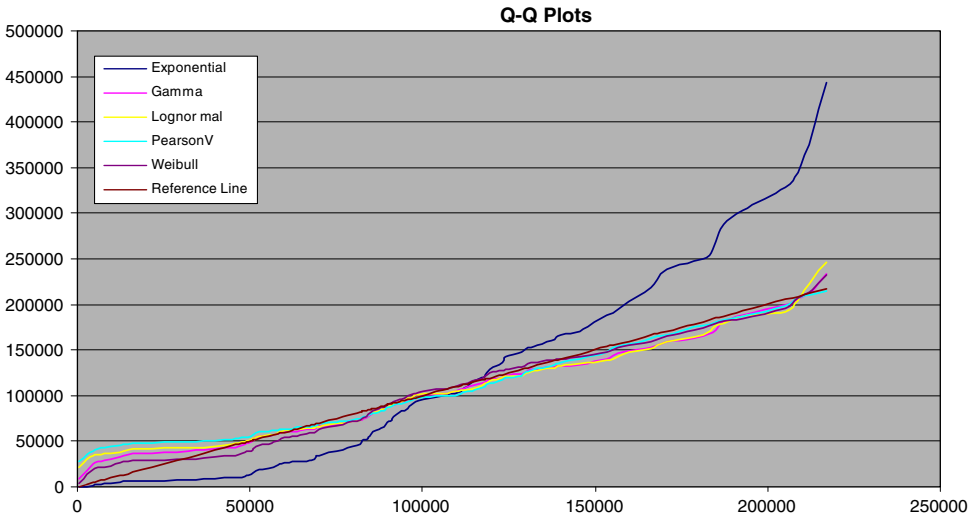

Fig. 2 Q-Q plot of the various distribution fits vs. the actual video for the I frames sequence of the single Sony Demo trace ([CIF, G16, B3, 28]). All values are in bits

Autoregressive models have been used in the past to model the output bit rate of VBR encoders, e.g. [39, 40]. A Discrete Autoregressive model of order p, denoted as DAR(p) [41, 42], generates a stationary sequence of discrete random variables with an arbitrary probability distribution and with an autocorrelation structure similar to that of an Autoregressive model. DAR(1) is a special case of a DAR(p) process and it is defined as follows: let $\{V_n\}$ and $\{Y_n\}$ be two sequences of independent random variables. The random variable V_n can take two values, 0 and 1, with probabilities 1-ρ and ρ, respectively. The random variable Y_n has a discrete state space S and $P\{Y_n = i\} = \pi(i)$. The sequence of random variables $\{X_n\}$ which is formed according to the linear model:

$$X_n = V_n X_{n-1} + (1 - V_n) Y_n$$

is a DAR(1) process.

A DAR(1) process is a Markov chain with discrete state space S and a transition matrix:

$$\mathbf{P} = \rho\mathbf{I} + (1-\rho)\mathbf{Q}$$

where ρ is the autocorrelation coefficient, $\mathbf{I}$ is the identity matrix and $\mathbf{Q}$ is a matrix with $Q_{ij} = \pi(j)$ for i, j $\in S$.

Autocorrelations are usually plotted for a range W of lags. The autocorrelation is calculated by the formula:

$$\rho(W) = E[(X_i - \mu)(X_{i+w} - \mu)]/\sigma^2$$

where μ is the mean and σ^2 the variance of the frame size for a specific video trace.

As in [34], where a DAR(1) model with negative binomial distribution was used to model the number of cells per frame of VBR teleconferencing video, we want to build a model based only on parameters which are either known at call set-up time or can be measured without introducing much complexity in the network. DAR(1) provides an easy and practical method to compute the transition matrix and gives us a model based only on four physically meaningful parameters, i.e., the mean, peak, variance and the lag-1 autocorrelation coefficient ρ of the offered traffic. The DAR(1) model can be used with any marginal distribution [41].

As already explained, the lag-1 autocorrelation coefficient for the I, P and B frames of each trace is very high in all the studied cases. Therefore, we proceeded to build a DAR(1) model for each video frame type for each one of the eighteen traces under study. More specifically, in our model the rows of the Q matrix consist of the Pearson type V probabilities (f_0, f_1, … f_k, F_K), where $F_K = \Sigma_{k>K} f_k$, and K is the peak rate. Each k, for $k < K$, corresponds to possible source rates less than the peak rate of K.

We proceeded again with testing our model statistically in order to study whether it produces a good fit for the I, P, B frames for the trace superposition. For this reason we have used again Q-Q plots, and we present indicatively some of these results in Figs. 3, 4, where we have plotted the 0.01-, 0.02-, 0.03-,… quantiles of the actual P and I video frames' types of the Sony Demo and NBC News traces ([CIF, G16, B3, F28]) versus the respective quantiles of the respective DAR(1) models, for a superposition of 20 traces.

As shown in Fig. 3, the points of the Q-Q plot fall almost completely along the 45-degree reference line, with the exception of the first and last 3% quantiles (left- and right-hand tail), for which the DAR(1) model underestimates the probability of frames with a very small and very large, respectively, number of cells. The very good fit shows that the superposition of the P frames of the actual traces can be modeled very well by a respective superposition of data produced by the DAR(1) model (similar results were derived for the superposition of B frames). Figure 4 presents the comparison of actual I frames with the respective DAR(1) model, for the NBC News trace.

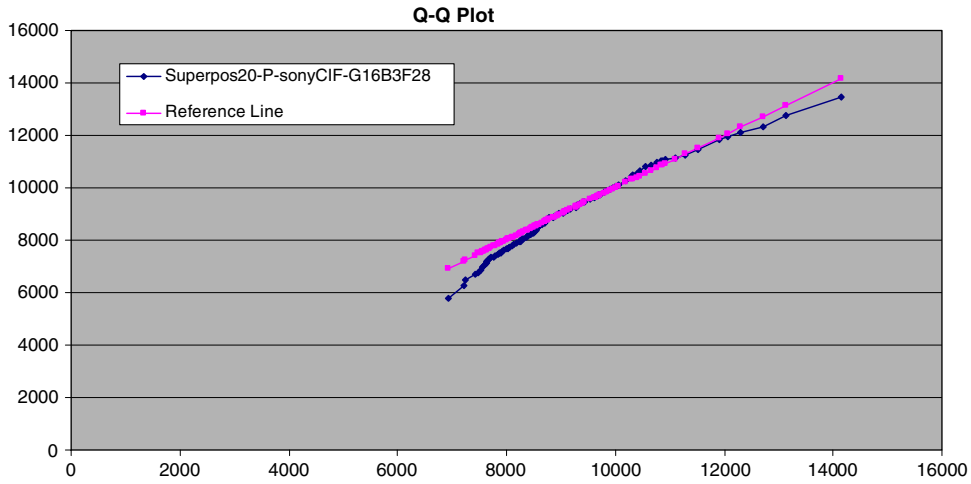

Fig. 3 Q-Q plot of the DAR (1) model versus the actual video for the P frames of Sony Demo ([CIF, G16, B3, 28]), for 20 superposed sources. All values are in packets

The results show that our method for modeling I frames of multiplexed H.264 videoconference streams provides relative accuracy. The results for all the other cases which are not presented in Figs. 3, 4 are similar in nature to the ones shown in the two figures. The very high accuracy of our approach in modeling P and B frames from multiplexed sources and the good accuracy in modeling I frames provide an efficient CAC scheme for H.264 videoconference sources. The use of wavelet modeling for the I -frames' size sequence could improve the accuracy of the I frame sizes modeling even further, but such an approach would unnecessarily increase our scheme's complexity, since the first order model accuracy is adequate for our CAC purposes.

7.1.5.3 H.264 streams For H.264 video coding, quite a few rate-control mechanisms have been recently proposed in the literature (e.g., [43, 44]). These techniques could be used in order to decrease the transmission rate of the traces under study, but the implementation of such mechanisms is beyond the scope of this work. Also, it should be noted that all traces used in this work are much burstier and much more demanding in terms of bandwidth requirements than the trace used in [11] for H.263 traffic (that trace had a mean of 91 Kbps, a peak of 500 Kbps and a standard deviation of 32.7 Kbps). This means that the H.264 video traces used in our study impose a larger "burden" on the system than the one imposed by H.263 videos; the new scheduling ideas proposed in this paper help the system to cope with the larger mean and peak rates and burstiness.

Since new video frames (VFs) arrive every 33.3 m, we have set the maximum transmission delay for video packets to 33.3 m, with packets being dropped when this deadline is reached. That is, all video packets of a VF must be delivered before the next VF arrives. The allowed video packet dropping probability is much stricter than that of voice, and is set to 0.0001 [7]. Video sources have exponentially distributed sessions with a mean duration of

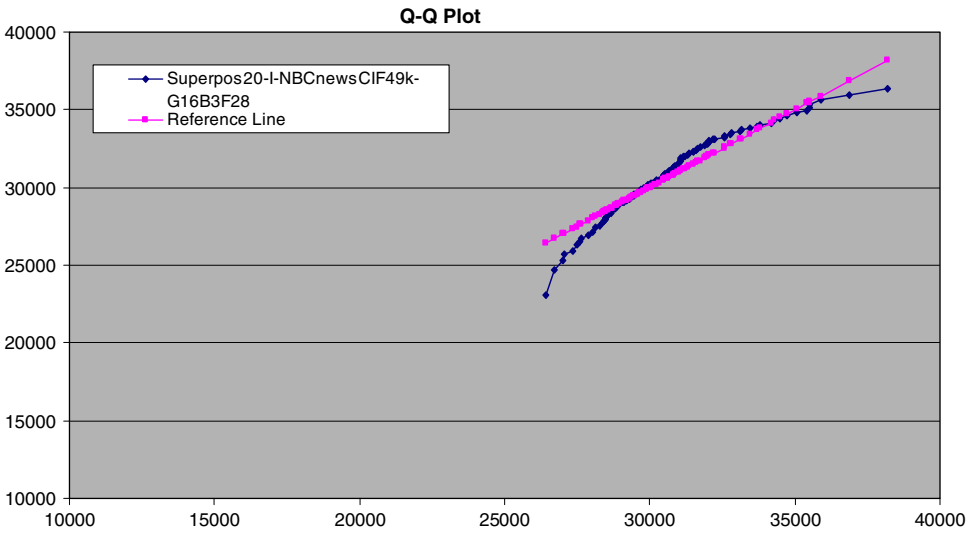

Fig. 4 Q-Q plot of the DAR (1) model versus the actual video for the I frames of NBC News ([CIF, G16, B3, F28]), for 20 superposed sources. All values are in packets

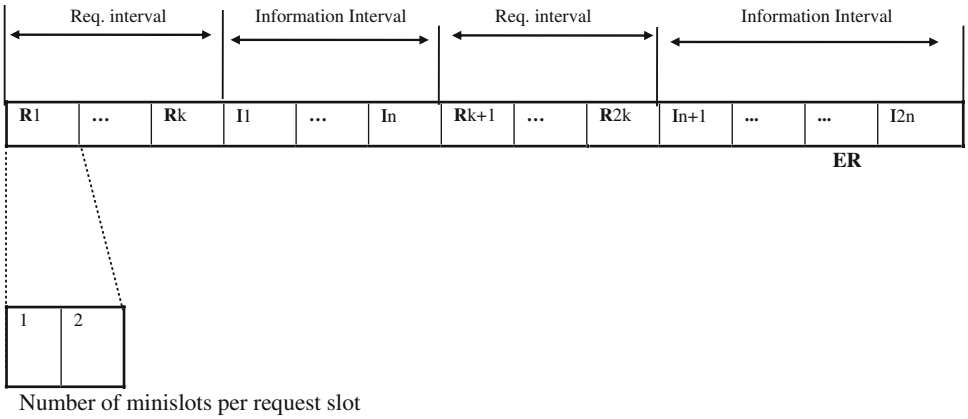

Fig. 5 Frame structure for the 20 Mbps channel, frame duration 12 ms

5 min (this duration has been denoted by global trials as an expected one for users of another wireless cellular video application [45]).

7.2 Channel structure and transmission protocols

The uplink channel time is divided into time frames of equal length. The frame duration is selected such that an active voice terminal (i.e., a terminal in talkspurt) generates exactly one packet per frame. Each frame consists of two *types* of intervals. These are the *request* intervals and the *information* intervals (shown in Fig. 5). The BS allocates channel resources at the end of each request interval. Within an information interval, each slot accommodates exactly one, fixed length, packet that contains voice, video, gaming or data information and a header.

As already mentioned in Sects. 2 and 5, certain design limitations had been adopted in [11], in the MI-MAC protocol study. Two of these limitations regard the channel structure and the actions of the mobile terminals (another two limitations in MI-MAC regard the Base Station Scheduling, and were discussed in Sect. 5):

(a) It was assumed that all of the voice source state transitions occur at the frame boundaries, therefore in [11] we placed all request intervals at the beginning of the frame, in order to minimize the voice packet access delay. This assumption is conceptually accurate, taking into consideration that the duration of a frame is equal to 12 ms, while the average duration of the talkspurt and silence periods are much larger (1 s and 1.35 s, respectively); still, compared to the realistic scenario of voice source state transitions occurring anywhere within the frame, it is a simplification. Hence, in this study *we consider that voice source state transitions can occur at any time within the frame*. The same assumption stands also for video, gaming and data message arrivals, therefore, in order to minimize packet access delay for all traffic types, we divide the request and information intervals of the frame in two parts, one placed at the beginning and the other in the middle of the frame, as shown in Fig. 5.

(b) Video sources were assumed in [11] to "live" permanently in the system (as in [7], they did not follow an ON-OFF state model like voice sources). Because of this assumption, and also because of the fact that the duration of the simulation study was long, video sources were assumed in MI-MAC to have already entered the system at the beginning of the simulation runs; for this reason, no request slots were used for video terminals' access to the channel. The above assumption guaranteed that the performance metrics results acquired would correspond to "worst-case" results for the system since the most bandwidth-demanding type of users (video), who also have the strictest QoS requirements, are constantly active; in this work video sources do not "live" permanently in the system, but have exponentially distributed sessions with a mean duration of 5 min, as explained in Sect. 7.1.5.3. This "relieves" a burden from the information interval of the channel in comparison to [11], as video terminals occasionally leave the information interval, but adds a significant burden to the request interval, which has to compensate for the increase in contention as video users attempt to gain channel access. When a video terminal has been successful in gaining channel access, it uses the header of the first packet of its current video frame to envoy its new slot requests (in comparison to the previous video frame) to the BS, both in the case of an increase and in the case of a decrease in bandwidth request.

In our work request slots can be shared by *all types of terminals* (video, voice, gaming and data) to transmit their requests to the BS. Due to the much stricter QoS requirements of video traffic, video terminals are given highest priority to transmit their requests to the BS; they are followed in priority by voice terminals and gaming terminals, respectively. Data request transmission is last, with priority given to IP-traffic, which is considered less delay-tolerant in our system than SMS traffic. This separation of each traffic type requests is possible due to the use of the two-cell stack random access algorithm, as it will be explained below.

The request intervals consist of slots, which are subdivided into mini-slots, and each mini-slot accommodates exactly one, fixed length, request packet. By using more than one minislot per request slot, a more efficient usage of the available request bandwidth is possible. We chose the number of minislots per request slot to be equal to 2, to allow for guard time and synchronization overheads, for the transmission of a generic request packet and for the propagation delay within the picocell (the number of minislots in a request slot cannot be increased at will, since a large number of minislots would mean that their duration would be very short and would not suffice for the request packet to be transmitted to the BS and for the BS to send an acknowledgement to the requesting terminals).

One conceptual similarity of our scheme with MI-MAC is that the number of request slots is variable per frame, depending on the total channel load in each frame. The choice of the frame structure parameters has been made as follows.

For all the examined scenarios of system load, we tried to find a maximum request bandwidth which would suffice for voice, gaming and data terminals. The reason that video traffic is not used in this calculation is that, in the presence of video terminals, which are bursty and very needy in terms of bandwidth, the number of voice, gaming and data terminals (all of which transmit at low rates) which can be accommodated by the system decreases drastically. Therefore, in the absence of video traffic we can calculate the maximum request bandwidth which will be needed by the system in order to resolve contention among all other types of terminals. This maximum number has been found, after extensive simulations, to be equal to 40 request slots (i.e., 20 request slots in each request interval of the frame, and 263 information slots in each information interval). We enforce a fully dynamic mechanism for the use of the request bandwidth: The number of request slots varies between 10 and 40 (10 request slots (5 in each request interval) is the minimum number of request slots needed for the end of contention because of the two-cell stack algorithm used for contention resolution; this algorithm is presented in the following paragraph). In the cases when less than 40 request slots are needed for the end of the contention (which, again, is known to the BS due to the use of the two-cell stack random access algorithm), the BS signals all user terminals for the existence of the additional information slots in the current frame. This dynamic design of the frame structure leads to a possible exploitation of as many as 30 slots of the frame as information slots, when the number of requesting voice, gaming and data terminals in a frame is very low and the contention among them ends quickly.

In our study, we adopt the *two-cell stack* reservation random access algorithm [46] for use by both video and voice terminals, due to its operational simplicity, stability and relatively high throughput when compared to PRMA-like algorithms. Another important reason for the choice of this algorithm is that it offers a clear indication of when contention has ended; this happens when two consecutive "non-collision" signals are transmitted by the BS in the downlink (this is the reason why at least 10 request slots, each containing 2 minislots, will be needed as a minimum for the system to understand that there is no new requesting traffic in a channel frame; at least 1 request slot is needed for each type of traffic originating from handoffs and 1 request slot for each traffic type originating from within the cell). Therefore, this algorithm supports the prioritization mechanism used for video, voice, gaming and data access to the requested minislots. The concept of reserving a minimum bandwidth for terminals to send requests and make channel reservations helps to keep the user access delay within relatively low limits and gives clearly better performance than the PRMA [23] and PRMA-like algorithms, such as [7, 9], where the absence of request slots leads to a continuously decreasing probability of finding available information slots as traffic increases, and hence to greater access delays. This will also be shown in our results' comparison with [7, 9] in Sect. 7.7. More specifically, given that the total number of slots in a channel frame is 566, the minimum number of 10 request slots, which need to be "sacrificed" in the case of no new requesting traffic corresponds to $10/566 = 1.77\%$ of the total bandwidth and is a very small price to pay compared to the advantages offered by the use of request slots and the use of the two-cell stack algorithm. Even the maximum bandwidth that can be dedicated to requests in the presence of a very high traffic load (i.e., 40 request slots, which correspond to 7.07% of the total bandwidth) is not significant when compared with the fact that for very high traffic loads the PRMA and PRMA-type algorithms become unstable.

The *two-cell stack* blocked access collision resolution algorithm [47] is adopted for use by the gaming terminals and by both types of data terminals in order to transmit their request packets. This algorithm is of window type, with First Come First Served (FCFS)-like service.

Any free information slot of the current channel frame (i.e., any slot which has not been allocated to a terminal after the end of the most recent request interval) can be temporarily used as an extra request (ER) slot [48]. ER slots can be used by all types of terminals, with the same priority set for regular request slots (video, voice, gaming, data). ER slots are again subdivided to two minislots.

7.3 DPRMA

The DPRMA protocol [7] was inspired by PRMA [23] and proposed for accommodating multimedia traffic, as PRMA

was optimized for systems for voice traffic only. The basic differences of DPRMA with our protocol are the following.

The first difference exists in the scheduling mechanism for video sources. The BS in DPRMA does not grant the earliest available information slots to requesting users. The BS first identifies which slots are currently unallocated and determines how many such slots exist. Next, it examines each of these slots in sequential order to determine if the slot will be assigned. Throughout the process, the BS maintains a record of how many slots *Sn* the user *n* (the user currently serviced by the BS) still needs. Every time a slot is successfully assigned, *Sn* is decremented. In addition, the BS keeps track of the number of available slots *Sc* that have not yet been considered for assignment. Each time a new slot is considered, *Sc* is decremented. As the BS sequentially considers each available slot, it assigns each one with probability *Pa*, where $Pa = Sn/Sc$. Thus, the probability that a slot is assigned is dependent upon how many slots are still needed to satisfy a user's request. This process tends to *spread the allocation of slots randomly throughout the frame*.

The second difference concerns the video and data traffic considered in each scheme, and the QoS requirements for data traffic. DPRMA does not use actual traces from video sources, but instead uses a video traffic model from [34], which is based on H.261 videoconference traffic (i.e., a model for video traffic from past technology encoders). Also, DPRMA considers an abstract model for data traffic (not referring to a specific type of data traffic), with which data packets (i.e., not messages) are generated independently from each other according to a Poisson process. DPRMA does not set any upper bound on data packet delay. The only QoS metric regarding data packets is that each data user has a large buffer of 10 KB (roughly equal to 190 packets) which should not overflow. This is however a very loose requirement, which does not help the system ensure that the data users QoS requirements in terms of transmission delays are satisfied.

The third difference is the use of certain transmission rates in DPRMA for all types of users. In DPRMA, a user continuously determines the appropriate reservation request that ensures timely delivery of its traffic. Newly generated packets are queued in a buffer as they await transmission. As the size of the queue grows, the user increases its reservation request to avoid excessive transmission delay. If the queue length subsequently decreases, the user then requests a lower reservation rate to avoid running out of packets. The buffer size that corresponds to an increase or decrease in the reservation request is defined as a threshold. DPRMA uses seven threshold levels, and, respectively, seven transmission rates for video users (the lowest rate is 70.667 kb/s and the highest 4.523 Mb/s). One pair of up- and down-threshold levels is implemented for data users, and one pair for voice users. The use of a number of transmission rates in DPRMA increases system complexity without ensuring that the video terminal will be allocated the maximum possible number of slots in each frame, based on its needs; the use of the transmission rates does guarantee that, in the long run, the terminal will be well serviced, but this policy is inadequate, as the very strict video packet dropping requirement asks for the best possible *short-term* (i.e., for every video frame) allocation.

The fourth difference is that DPRMA uses neither request slots nor our idea of p^*, but adopts a PRMA-like approach for voice and data users, by allowing them to compete for the available information slots by transmitting their packets according to a probability ($Ptv = 0.05$ for voice, $Ptd = 0.007$ for data traffic). No transmission probability is needed for video users because in DPRMA video users are considered to "live" permanently in the system and to have obtained reservations prior to the beginning of the simulation. This is not the case in our protocol, as we attempt to simulate a fully realistic wireless cellular network scenario, as explained in Sects. 5 and 7.2.

The fifth difference is that in DPRMA all users waste one slot when giving up their reservations. This does not happen in our protocol because of the VAD used for voice terminals and because the BS knows exactly when a video user has transmitted all the packets of its VF (since video users convey this information to the BS, whereas in DPRMA they convey only at times a reservation request rate in order to keep the content of their video packet buffers below certain thresholds).

The sixth difference is that DPRMA does not consider channel errors, and therefore does not have a mechanism in order to "exploit" the bad channel state of one video user in favor of another video user with good channel state, as is done in our protocol.

The seventh difference is the use of video traffic modeling for CAC in our scheme, as explained in Sect. 5.

7.4 DPRMA*

Since DPRMA was evaluated for different types of multimedia traffic than the ones considered in MI-MAC, we have modified DPRMA very slightly in order to be able to make comparisons between DPRMA and our protocol. We will refer to this modified protocol as DPRMA*. DPRMA* has five differences in comparison to DPRMA:

First, it is implemented on the same types of multimedia traffic as our protocol.

Second, its performance is evaluated under the same channel error model as our protocol.

Third, in DPRMA, it was found that in a data-only system (for the data model used in [7]) nearly identical performance was achieved when $0.006 < Ptd < 0.1$. Since

data users are given lowest priority, the lowest possible value of Ptd = 0.007 was chosen in the scheme. In DPRMA*, where two types of data traffic (SMS and IP) are considered and SMS traffic is more delay-tolerant, the SMS data transmission probability is kept to the lowest possible value of Ptdsms = 0.007, whereas for IP data traffic the transmission probability is chosen equal to Ptdip = 0.014, (i.e., double than Ptdsms). Our results have shown that this choice for Ptdip provides much smaller average access delays for IP traffic than the delays provided by lower values and does not severely influence the QoS of voice and gaming traffic, as was the case with much higher values of Ptdip. For gaming traffic, which was not present in DPRMA, we used Ptgam = 0.02 in DPRMA*, as this value, once again, was shown via simulation to provide steadily good QoS metrics results for all the traffic loads used in our study.

Fourth, for "fairness" reasons, i.e., for being able to compare the "best" possible version of DPRMA* with our scheme, we have slightly changed the number of threshold levels' transmission rates for video users. We have used nine transmission rates (and, respectively, nine threshold levels), whereas in DPRMA seven transmission rates were used. The rates were defined by the same equation used in [7]; the lowest rate was equal to 35.33 Kbps and the highest rate was equal to 9.045 Mbps.

Fifth, in DPRMA video users were assumed to "live" permanently in the system (as in our former work in [11]). For this reason, we had to find a fair way of accommodating in DPRMA* the exponentially distributed sessions of video users which are considered in our scheme, as well as the video handoff traffic (if we considered that this is done in a "magical" way obviously the results for DPRMA* would be greatly improved, but the protocol would be totally impractical). Hence, our chosen approach in DPRMA* is that when new or handoff video users arrive at the cell during channel frame X they do not enter contention (since this was the approach used in the original DPRMA) but they wait until the beginning of frame X + 1 before they start transmitting (i.e., they need to wait on average half a channel frame in order to inform the BS of their bandwidth requirements).

No other differences exist between DPRMA and DPRMA*.

7.5 The MAC scheme in [9]

The authors in [9] design a MAC protocol for wireless cellular networks which is based on PRMA but is focused on the transmission of multimedia traffic (video, voice and data, as in DPRMA).

Due to the large bandwidth requirements, the authors consider the presence of only 1 video user in their 5.3 Mbps wireless channel. The video is a real trace encoded with a past technology encoder (MPEG-1).

The first important difference of our scheme with [9] is that in [9] a fixed number of slots are dedicated in each frame to video traffic (the authors' work focuses in part on finding an "optimal" number of slots for the single video user). It is clear that, due to the burstiness of video traffic, even with a very good choice for the number of slots dedicated to the video user, in many frames this choice will lead to the loss of valuable bandwidth, when the video user happens to be transmitting at a lower rate (as mentioned in Sect. 5.1, in [9] the authors dedicate 38% of the total slots in each channel frame to the single video user, which on average was found to generate traffic equal to 31.24% of the channel capacity; therefore, even for a video user of very limited burstiness, like the one used in [9], 6.76% of the channel capacity is lost in the attempt to offer the required QoS to video traffic). Also, it should be emphasized that the authors' results in terms of the optimum value of voice permission probability and video permission probability, as well as for the best value for the dedicated slots to the video user, are valid *only* for the specific scenario that is studied in [9]. Therefore, in the presence of more than 1 video user these probabilities and the respective number of dedicated slots would have to change. The same is true for the case when video traffic encoded with different encoding schemes than MPEG-1 would enter the network, and for the case when data traffic following a different data model would enter the network (the authors use a data traffic model where both the data message length and the data message interarrival times are exponentially distributed). Hence, system complexity is extremely high in [9] if for each different traffic type and number of sources present in the system new system parameters need to be calculated. On the contrary, in our scheme the specific traffic models and the number of users of each traffic type are of minor importance, as our mechanism prioritizes the terminals based on the type of traffic and is fully dynamic in terms of the request slots which will be dedicated to newly requesting terminals. The only dependence of our scheme on the traffic type has to do with the use of our traffic model for H.264 traffic prediction. However, this is a loose dependence, as quite a few competent traffic models exist in the literature for previous technology encoding schemes (MPEG-1, MPEG-2, H.263, MPEG-4); therefore, in the case of a video terminal transmitting traffic encoded with any of these schemes, our MAC protocol can be easily implemented simply by changing our H.264 traffic model with any of the respective models for other video encodings.

The second significant difference between our protocol and [9] is that, in [9], video users needing more slots have to contend with voice and data users in order to acquire

them, in a PRMA-manner. This means that video users are "exposed" to the possibility of collisions with lower priority terminals. As explained in Sect. 7.2, this is not the case in our scheme, where all types of traffic are prioritized.

The third difference concerns the fact that in [9] no upper bound is set on data packet delay, therefore the system cannot ensure that the data users QoS requirements in terms of transmission delays are satisfied.

The fourth difference is that in [9], data terminals must contend for each packet transmission. This choice increases contention (in which all types of terminals are exposed) and, additionally, is an exhausting procedure for data terminals, given their very low permission probability (ranging from 0.001 to 0.005) and the fact that a data message carries on average 10 packets. In both our scheme and DPRMA data terminals can make reservations for more than one slot, therefore they only need to contend for the transmission of the first packet of their message.

The fifth difference is that [9] does not have a mechanism in order to "exploit" the bad channel state of one video user in favor of another video user with good channel state (as is done in our protocol), since [9] does not consider channel errors.

The sixth difference is the use of video traffic modeling for CAC in our scheme, as explained in Sect. 5.

The seventh difference is that in [9] no preemption takes place for lower priority users.

The eighth difference is that in [9], as in DPRMA, all users waste one slot when giving up their reservations. As already explained in Sect. 7.3, this does not happen in our scheme.

7.6 Modified [9] MAC protocol

The MAC protocol presented in [9] was evaluated for different types of multimedia traffic than the ones considered in our protocol. Therefore, similarly to our approach for DPRMA, we have modified [9] very slightly in order to be able to make comparisons between [9] and our protocol. We will refer to this modified protocol as [9]*.

[9]* Has four differences in comparison to [9]:

First, it is implemented on the same types of multimedia traffic as our protocol.

Second, its performance is evaluated under the same channel error model as our protocol.

Third, in [9] the permission probabilities for each traffic type were calculated based on the fact that only one video user was present in the system (even in this case, as the authors pointed out in [9] the task of computing the best values for the permission probabilities and for the number of dedicated slots to video was complex). In the case of the 20 Mbps channel under study, the number of video users that can be present in the system is significantly larger. Since the permission probabilities for video affect the permission probabilities for voice, gaming and data terminals, the complexity of finding the optimal values for [9]* is extremely high—as already mentioned, this is one significant disadvantage of [9]. Actually, we found that the optimal values are different for each different video load. In order to provide a scheme, in [9]*, which would resemble the logic of [9], we have used for the permission probabilities the following values (given that video traffic has first priority in our scheme): Pvideo = 0.1, Pvoice = 0.015, Psms = 0.001, Pip = 0.003, Pgam = 0.005; these values produced steadily good QoS metrics results for all the traffic loads used in our study.

Fourth, the number of slots dedicated to video traffic in [9]* was calculated as follows. In [9], as explained in Sect. 7.5, 38% of the total slots in each channel frame were dedicated to the single video user, which on average was found to generate traffic equal to 31.24% of the channel capacity. Therefore, since in [9] the ratio of the slots offered divided by the traffic generated was 38%/31.24% = 1.216, we have computed the number of dedicated slots in [9]* as 1.216*N*566*R/20, where N is the number of video terminals present in the system, 566 is the number of slots per channel frame, R (in Mbps) is the mean bit rate of the video source (which can be any of the four used in our study) and 20 Mbps is the channel rate.

No other differences exist between [9] and [9]*.

7.7 Results

The significant differences among our scheme, DPRMA and [9], which were outlined in Sects. 7.3 and 7.5 will be shown to help our scheme significantly outperform DPRMA* (although DPRMA* uses the impractical assumption of no contention among video terminals) and [9]*, based on our QoS metrics. DPRMA* will be shown to be second in performance; its results are better than those when [9]* is used, due to the aforementioned assumption of no video contention, but also due to the scheduling choice in [9]* of forcing data terminals to contend for the transmission of each packet in their message, and due to the fact that [9]* does not use preemption in favor of higher priority traffic, as both DPRMA* and our scheme do.

As already mentioned earlier in the paper, we have implemented DPRMA* and [9]* on the same types of multimedia traffic as our protocol, and we have evaluated their performance in a 20 Mbps wireless channel, under the same channel error model as our protocol.

Table 5 presents the maximum voice capacity and channel throughput achieved by each of the three schemes, for a mean IP packet interarrival time equal to 8 ms (corresponding to an average traffic load of 370 Kbps), 40

Table 5 Maximum voice capacity and channel throughput results for the three schemes, with various traffic loads, and 10% handoff traffic (IP packet interarrival time = 8 ms, Ngaming = 40)

Start point for voice preemption in our scheme	λ (messages/frame)	Video terminals	Voice terminals			Channel throughput (%)		
			Our scheme	DPRMA*	[9]*	Our scheme	DPRMA*	[9]*
0.14	2	10	12	×	×	51.5	×	×
	4	10	7	×	×	52.1	×	×
	6	10	4	×	×	52.9	×	×
	8	10	1	×	×	53.4	×	×
0.22	2	8	398	319	293	63.0	57.6	55.7
	4	8	387	311	278	63.4	57.0	54.8
	6	8	384	299	260	63.9	56.1	52.9
	8	8	370	286	235	64.5	55.6	51.8
0.35	2	6	587	506	461	69.3	63.8	61.5
	4	6	574	502	450	69.9	63.4	60.9
	6	6	567	488	440	70.7	62.9	59.9
	8	6	562	468	422	71.2	62.2	59.3
0.52	2	4	895	834	780	77.8	72.7	70.5
	4	4	864	796	735	78.7	73.2	69.7
	6	4	815	739	673	77.1	69.8	65.9
	8	4	767	671	608	75.6	65.9	62.3
0.7	2	2	1112	1048	1002	88.5	84.1	81.3
	4	2	1065	991	941	87.4	81.6	77.9
	6	2	986	914	840	86.0	78.5	73.6
	8	2	908	801	717	85.6	74.9	69.4

gaming users present in the system (corresponding to an average traffic load of 850 Kbps) and for various SMS traffic loads; by "the maximum voice capacity and channel throughput" we refer to the respective maximum numbers that the system can reach with each protocol while satisfying the QoS requirements of each traffic type, as these were defined in Sect. 7.1. The column titled "start point for preevo" denotes the point in the frame where voice preemption (preevo) has to start in our scheme, for the specific traffic loads. Again, we emphasize that this is just a rough estimation that the BS needs in order to know where to start voice preemption; then, by using the algorithm outlined in Sect. 5, the BS is able to find in a few "steps" the ideal preemption start point, and therefore accommodate the bursty video traffic with a close to minimum deterioration on the QoS of voice sources.

The SMS message arrival rates of 2, 4, 6 and 8 messages/frame correspond to 212 Kbps, 424 Kbps, 636 Kbps and 848 Kbps of SMS data load respectively. All these results have been produced for the case when 10% of each type of traffic enters the network at some point as handoff traffic.

As shown in the Table, our scheme is able to accommodate a significantly larger number of voice sources for all traffic loads, in comparison to DPRMA* and [9]*. More specifically:

1. Our scheme's maximum channel throughput is higher than the channel throughput achieved by the other two schemes in all the studied cases, averaging a 7.1% higher throughput in comparison to DPRMA* and a 10.3% higher throughput over [9]*. The cases of traffic loads which can be accommodated by our scheme but cannot be accommodated by the other two schemes (we comment on this below) are not included in this calculation.
2. Our scheme achieves higher voice capacity than the other two schemes in all the studied cases. In comparison to DPRMA*, the voice capacity increase in our scheme ranges from 6.1% to 29.4%, averaging at 15.6% over all the studied cases. In comparison to [9]*, the voice capacity increase in our scheme ranges from 11% to 57.4%, averaging at 27.8% over all the studied cases. Again, the cases of traffic loads which can be accommodated by our scheme but cannot be accommodated by DPRMA* and [9]* are not included in the above calculations.

Some additional comments that need to be made on the results in Table 5 are:

a. As expected from the bursty nature of video traffic, all schemes achieve much higher channel throughput when the number of video terminals in the system decreases

and, subsequently, the number of voice traffic (which is Constant Bit Rate and therefore not bursty) increases.

b. We observe from the Table that, for higher video traffic loads, voice capacity remains almost constant in our scheme as the SMS message arrival rate λ increases. The reason for this result is our data preemption policy. Still, in the cases of lower video traffic loads, we observe that voice capacity decreases with the increase in the data message arrival rate. The reason for this result is that the system can not sustain, in these cases, a higher voice traffic load, as the high number of voice users impedes video users from finding enough idle slots to transmit in, hence the data preemption mechanism in favor of video traffic causes the average SMS message, IP packet and gaming packet delays to exceed their set upper bounds.
c. The symbol "x" in the Table denotes that DPRMA* and [9]* are unable to accommodate the respective traffic load, due to the violation of one or more QoS requirements of the various traffic types.
d. The results shown in the Table for the case of 10 video users are a good example for the usefulness of the CAC scheme which we propose. In DPRMA* and [9]*, when 10 video users enter the network, the lack of a CAC scheme will allow them to attempt to transmit, in which case their QoS requirements will be violated. On the contrary, in our scheme, when 10 video users are already transmitting and an 11th video user attempts to enter the network (this capacity cannot be accommodated by our scheme), the call admission control module knows a priori that the addition of this user will lead to the violation of the existing users' QoS and denies the user access to the network.

Figure 6 presents the video (Pvidrop) and voice (Pdrop) packet dropping probability in our MAC scheme versus the number of voice users, when 5 video terminals are present in the system, the mean IP packet interarrival time is 3 ms (corresponding to an average traffic load of 990 Kbps), 60 gaming users are present in the system (corresponding to an average traffic load of 1.28 Mbps) and the SMS message arrival rate is equal to 6 messages/frame (corresponding to an average traffic load of 638 Kbps). The effectiveness of our BS scheduling policy is clear from the figure, which shows that with the voice preemption policy in favor of video traffic the average voice packet dropping probability is steadily close to the upper bound of 0.01, so that the system can incorporate more video users.

Also, the figure shows that the BS scheduling policy of offering highest priority to video traffic helps to keep the average video packet dropping probability just slightly higher than the probability $p = 8.2*10^{-5}$ of a transmission error, for a large region of voice users' capacities.

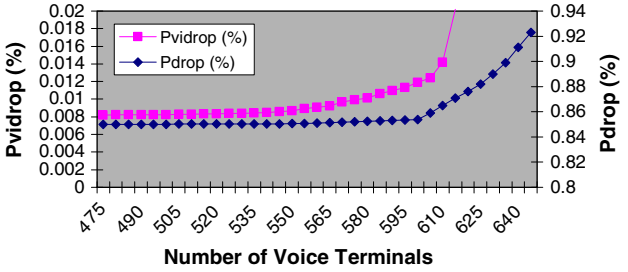

Fig. 6 Video and voice packet dropping probability versus number of voice users in our scheme. (Nvid = 5, Ngaming = 60, IP packet interarrival time = 3 ms, λ = 6 SMS messages/frame), 5% handoff traffic

Therefore, Fig. 6 shows that our BS scheduling policy "pushes" both types of traffic to their limits regarding their QoS requirements, and hence succeeds in achieving high channel throughput for most cases of traffic loads.

Figures 7 and 8 present the average SMS message delay and IP packet delay, respectively, versus the number of voice terminals for the three schemes. The results presented in these figures have been obtained for an average traffic load of 15% of the total channel capacity emanating from video, gaming, IP and SMS traffic. Our scheme is shown from the two figures to be able to accommodate 1074 voice users while satisfying both the IP and SMS traffic QoS requirements in terms of maximum delays. DPRMA* and [9]* are able to accommodate 959 and 814 voice users, respectively, i.e., our scheme achieves a 12% increase in voice capacity in comparison to DPRMA* and a 31.9% increase in voice capacity in comparison to [9]*.

Figure 9 presents the average gaming packet delay versus the number of voice terminals for the three schemes. The results presented in the figure have been obtained for a much higher average traffic load (35% of the total channel capacity) emanating from video, gaming, IP and SMS traffic, in comparison to the load used in Figs. 7 and 8. Our scheme is shown once again to clearly excel when compared with the other two protocols, as it is able to accommodate 395 voice users while satisfying the gaming traffic QoS requirements (in the figure, only the gaming packet delay is shown; the other QoS requirement for gaming traffic, i.e., gaming packet dropping probability $<1\%$ follows a similar curve to that shown in Fig. 9 for all schemes). DPRMA* and [9]* are able to accommodate 326 and 288 voice users, respectively, i.e., our scheme achieves a 21.2% increase in voice capacity in comparison to DPRMA* and a 37.2% increase in voice capacity in comparison to [9]*.

Finally, we would like to emphasize that although this work focuses on the case study of H.264 videoconference traffic, its implementation is not limited to this case. The protocol can be easily implemented on any kind of

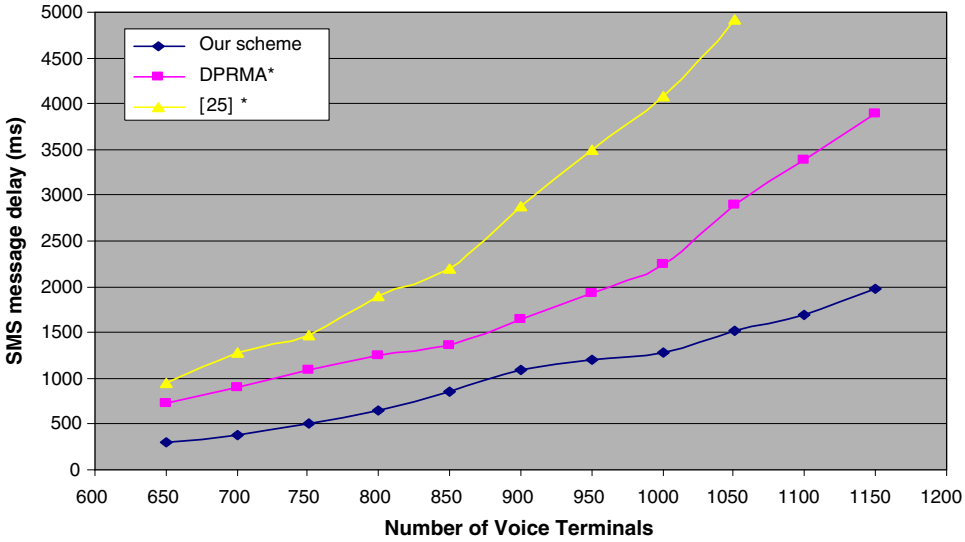

Fig. 7 Average SMS message delay versus number of voice users (Nvid = 2, $\lambda = 2$ SMS messages/frame, Ngaming = 14, IP packet interarrival time = 15 ms), 15% handoff traffic

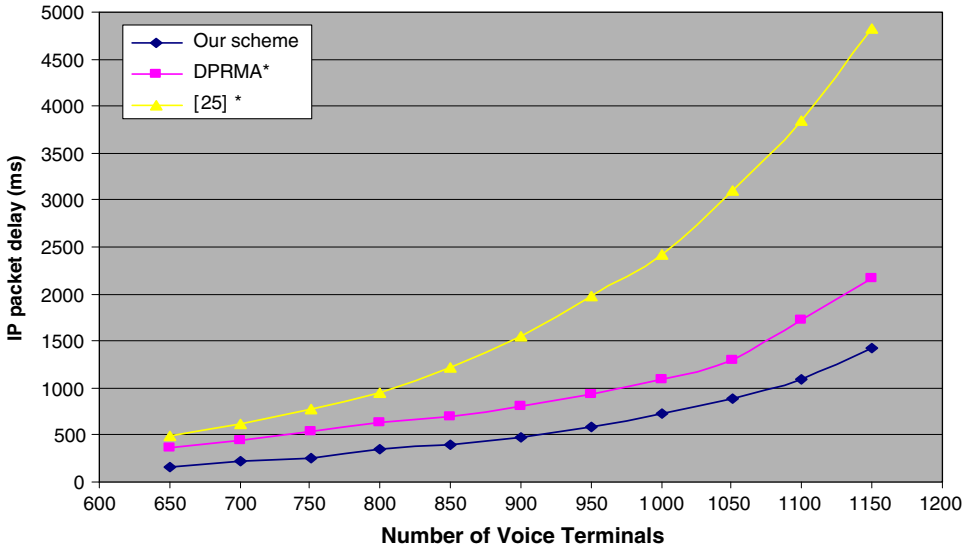

Fig. 8 Average IP packet delay versus number of voice users (Nvid = 2, $\lambda = 2$ SMS messages/frame, Ngaming = 14, IP packet interarrival time = 15 ms), 15% handoff traffic

multimedia traffic, as long as the priorities for each traffic type are clearly set by the provider. For video traffic of different encoding, the efficiency of the call admission control mechanism can be ensured with the use of an accurate model (e.g., the work proposed by our group in [32, 33] for H.263 and MPEG-4 traffic).

8 Conclusions

In this paper we have proposed and investigated the performance of a Multiple Access Control (MAC) protocol for transmitting H.264 videoconference streams and voice with gaming, SMS and IP data traffic over a high-speed wireless TDMA channel with errors and capture. To the best of our knowledge, this is one of the first papers in the literature investigating the integration of H.264 video traffic and network games traffic with other types of traffic over wireless networks. We evaluated the protocol's performance through an extensive simulation study, and our results show that, regardless of the severe channel conditions examined in our work (very high error probability compared to the video QoS requirements), our scheme steadily achieves high channel throughput and outperforms two other efficient MAC protocols for multimedia traffic, while satisfying the QoS requirements of all three traffic types. This is achieved with the use of our new Base Station Scheduling ideas, and their combination with an efficient idea for call admission control at the network entrance. Furthermore, due to the use of the dynamic channel frame structure, the two-cell stack random access algorithm and the CAC mechanism based on video traffic modeling, our scheme is more stable than the PRMA-like protocols and can easily prioritize and accommodate all

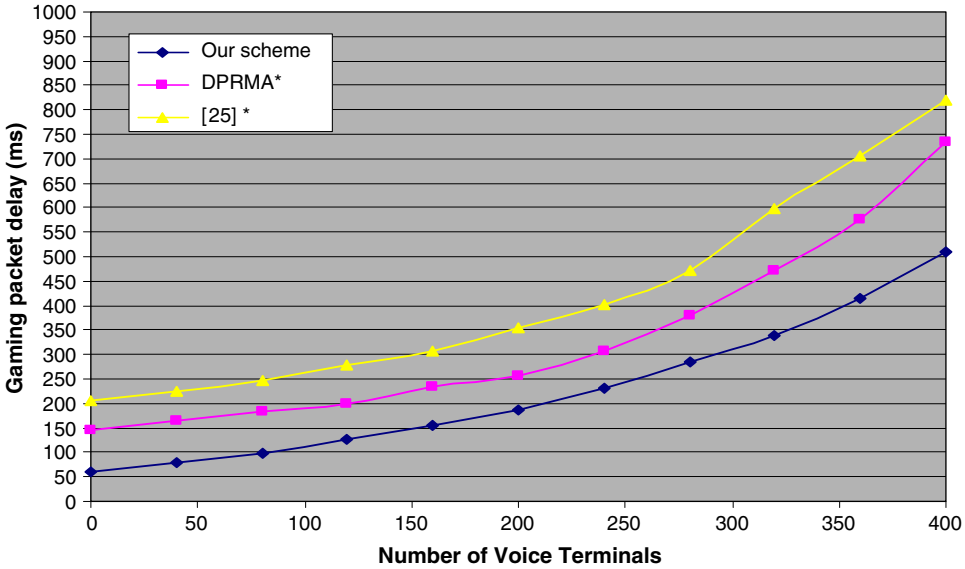

Fig. 9 Average gaming packet delay versus number of voice users (Nvid = 7, $\lambda = 4$ SMS messages/frame, Ngaming = 20, IP packet interarrival time = 2 ms), 10% handoff traffic

traffic types and any incoming handoff traffic. Also, in terms of our scheme's complexity, the use of two separate modules for CAC and scheduling, as well as the use of a simple, first-order video traffic model, help towards the low complexity of the proposed protocol; the only limitation which can be attributed to our scheme is that, in terms of the CAC module, there needs to be enough system memory in order to store all new, previously non-precomputed traffic scenarios. On the contrary, PRMA-like protocols increase significantly in complexity as they continuously need to adjust the permission probabilities of existing traffic types to ensure that the QoS requirements of new and handoff traffic are satisfied.

Acknowledgment This work has been funded, in part, by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. [Online] http://www.eurotechnology.com/4G/.
2. Passas, N., Skyrianoglou, D., & Merakos, L. (1997). Traffic scheduling in wireless ATM networks. In *Proceedings of the IEEE ATM'97 Workshop*. Lisbon, Portugal, May 1997.
3. Colombo, G., Lenzini, L., Mingozzi, E., Cornaglia, B., & Santaniello, R. (2002). Extended performance evaluation of PRADOS: A scheduling algorithm for traffic integration in a wireless ATM network. *Wireless Networks, 8*(2–3), 265–274.
4. Akyildiz, I. F., McNair, J., Martorell, L. C., Puigjaner, R., & Yesha, Y. (1999). Medium access control protocols for multimedia traffic in wireless networks. *IEEE Network, 13*(4), 39–47.
5. Raychaudhuri, D., French, L. J., Siracusa, R. J., Biswas, S. K., Ruixi, Y., Narasimhan, P., et al. (1997). WATMnet: A prototype wireless ATM system for multimedia personal communication. *IEEE Journal on Selected Areas in Communications, 15*(1), 83–95.
6. Musumeci, L., Giacomazzi, P., & Fratta, L. (2000). Polling - and contention-based schemes for TDMA- TDD access to wireless ATM networks. *IEEE Journal on Selected Areas in Communications, 18*(9), 1597–1607.
7. Dyson, D. A., & Haas, Z. J. (1999). A dynamic packet reservation multiple access scheme for wireless ATM. *ACM/Baltzer MONET Journal, 4*(2), 87–99.
8. Wang, X., & Chen,Y. (2001). A TDMA/FDD MAC protocol based on a new resource updating scheme. In *Proceedings of the IEEE GLOBECOM 2001*. San Antonio, USA.
9. Elnoubi, S., & Alsayh, A. M. (2004). A packet reservation multiple access (PRMA)-based algorithm for multimedia wireless system. *IEEE Transactions on Vehicular Technology, 53*(1), 215–222.
10. Bianchi, G., Borgonovo, F., Fratta, L., Musumeci, L., & Zorzi, M. (1997). C-PRMA: A centralized packet reservation multiple access for local wireless communications. *IEEE Transactions on Vehicular Technology, 46*(2), 422–436.
11. Koutsakis, P., Psychis, S., & Paterakis, M. (2005). Integrated wireless access for videoconference from MPEG-4 and H.263 video coders with voice e-mail and web traffic over wireless networks. *IEEE Transactions on Vehicular Technology, 54*(5), 1863–1874.
12. Holliday, T., Goldsmith, A., & Glynn, P. (2002). Wireless link adaptation policies: QoS for deadline constrained traffic with imperfect channel estimates. In *Proceedings of the IEEE ICC 2002* (Vol. 5, pp. 3366–3371). New York, USA.
13. Bottigliengo, M., Casetti, C., Chiasserini, C.-F., & Meo, M. (2004). Short-term fairness for TCP flows in 802.11b WLANs. In *Proceedings of the IEEE Infocom 2004*, Hong Kong, China.
14. Cidon, I., Kodesh, H., & Sidi, M. (1988). Erasure capture and random power level selection in multiple-access systems. *IEEE Transactions on Communications COM-36, 3*, 263–271.
15. Daigle, J. N., Giacomazzi, P., & Petorovic, R. (1995). Contention protocol performance in wireless personal communication systems. In *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communication (PIMRC)* (pp. 1112–1117). Toronto, Canada.
16. Brasche, G., & Walke, B. (1997). Concepts, services and protocols of the new GSM Phase 2+ General Packet Radio Service. *IEEE Communications Magazine, 35*(8), 94–104.
17. Daigle, J. N., & Magalhaes, M. N. (2003). Analysis of packet networks having contention-based reservation with application to GPRS. *IEEE/ACM Transactions on Networking, 11*(4), 602–615.
18. Li, Y. (2000). Pilot-symbol aided channel estimation for OFDM in wireless systems. *IEEE Transactions on Vehicular Technology, 49*(4), 1207–1215.
19. Koutsakis, P., & Paterakis, M. (2004). Call admission control and traffic policing mechanisms for the transmission of

videoconference traffic from MPEG–4 and H.263 video coders in wireless ATM networks. *IEEE Transactions on Vehicular Technology, 53*(5), 1525–1530.
20. Chatziperis, S., Koutsakis, P., & Paterakis, M. (2008). A new call admission control mechanism for multimedia traffic over next generation wireless cellular networks. *IEEE Transactions on Mobile Computing, 7*(1), 95–112.
21. Vuran, M. C., & Akyildiz, I. F. (2007). A-MAC: Adaptive medium access control for next generation wireless terminals. *IEEE/ACM Transactions on Networking, 15*(3), 574–587.
22. Salem, N. B., Buttyan, L., Hubaux, J.-P., & Jacobsson, M. (2003). A charging and rewarding scheme for packet forwarding in multi-hop cellular networks. In *Proceedings of the ACM MobiHoc* 2003. Maryland, USA.
23. Nanda, S., Goodman, D. J., & Timor, U. (1991). Performance of PRMA: A packet voice protocol for cellular systems. *IEEE Transactions on Vehicular Technology, 40*, 584–598.
24. ETSI. Digital Cellular Telecommunications System (Phase 2 +); Technical Realization of the Short Message Service (SMS); Point-to-Point (PP). (GSM 03.40)
25. Hrasnica, H., & Lehnert, R. (2003). Investigation of MAC protocols for broadband PLC networks under realistic traffic conditions. In *Proceedings of the ITCom 2003*. Orlando, USA, September
26. Tran-Gia, P., Staehle, D., & Leibnitz, K. (2001). Source traffic modeling of wireless applications. *International Journal of Electronics and Communications, 55*(1), 27–36.
27. Fitzek, F., Köpsel, A., Wolisz, A., Krishnam, M., & Reisslein, M. (2002). Providing application-level QoS in 3G/4G wireless systems: A comprehensive framework based on multirate CDMA. *IEEE Wireless Communications, 9*(2), 42–47.
28. Borella, M. S. (2000). Source models of network game traffic. *Computer Communications, 23*(4), 403–410.
29. Dainoti, A., Pescape, A., & Ventre, G. (2005). A packet-level traffic model of starcraft. In *Proceedings of the Second IEEE International Workshop on Hot Topics in Peer-to-Peer Systems (HOT-P2P'05)*. San Diego, USA, July 2005.
30. Wiegand, T., Sullivan, G., & Luthra, A. (2003). Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496 -10 AVC). May 2003.
31. Sullivan, G. J., Topiwala, P. & Luthra, A. (2004). The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions. In *Proceedings of the SPIE Conference on Applications of Digital Image Processing XXVII, Special Session on Advances in the New Emerging Standard:H.264/AVC*. Denver, USA, August 2004.
32. Koutsakis, P. (2006). A new model for multiplexed VBR H.263 videoconference traffic. In *Proceedings of the IEEE GLOBECOM 2006*. San Francisco, USA.
33. Lazaris, A., Koutsakis, P., & Paterakis, M. (2008). A new model for video traffic originating from multiplexed MPEG–4 videoconference streams. *Performance Evaluation, 65*(1), 51–70.
34. Heyman, D. P., Tabatabai, A., & Lakshman, T. V. (1992). Statistical analysis and simulation study of video teleconference traffic in ATM networks. *IEEE Transactions on Circuits and Systems for Video Technology, 2*(1), 49–59.
35. [Online] http://trace.eas.asu.edu/h264/index.html.
36. Law, A. M. & Kelton, W. D. (1991). *Simulation modeling & analysis*. 2nd ed. McGraw Hill Inc.
37. Burnham, K. P., & Anderson, D. R. (2002). *Model selection and multi-model inference*. New York: Springer-Verlag.
38. Frey, M., & Ngyuyen-Quang, S. (2000). A gamma-based framework for modeling variable-rate video sources: The GOP GBAR model. *IEEE/ACM Transactions on Networking, 8*(6), 710–719.
39. Maglaris, B., Anastassiou, D., Sen, P., Karlsson, G., & Robbins, J. D. (1988). Performance models of statistical multiplexing in packet video communications. *IEEE Transactions on Communications, 36*(7), 834–844.
40. Shim, C., Ryoo, I., Lee, J., & Lee, S. (1994). Modeling and call admission control algorithm of variable bit rate video in ATM networks. *IEEE Journal on Selected Areas in Communications, 12*(2), 332–344.
41. Adas, A. (1997). Traffic models in broadband networks. *IEEE Communications Magazine, 35*(7), 82–89.
42. Jacobs, P. A., & Lewis, P. A. W. (1983). Time series generated by mixtures. *Journal of Time Series Analysis, 4*(1), 19–36.
43. Yu, H., Lin, Z., & Pan, F. (2005). An improved rate control algorithm for H.264. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)* 2005 (Vol. 1, pp. 312–315) Kobe, Japan.
44. Jiang, M., & Ling, N. (2006). On Lagrange multiplier and quantizer adjustment for H.264 frame-layer video rate control. *IEEE Transactions on Circuits and Systems for Video Technology, 16*(5), 663–669.
45. Baig, E. C. (2006). Will consumers tune in to a tiny TV in their hand? July 2006, [Online] http://www.usatoday.com/tech/wireless/2006-08-17-mobile-tv_x.htm.
46. Cleary, A., & Paterakis, M. (1995). An investigation of stack based algorithms for voice packet transmission in microcellular wireless environments. In *Proceedings of the IEEE International Conference on Communications (ICC)* (pp. 1375–1381). Seattle, USA, June 1995.
47. Cleary, A., & Paterakis, M. (1997). Design and performance evaluation of an RRA scheme for voice-data channel access in outdoor microcellular wireless environments. *Mobile Networks and Applications (MONET) Journal, 21*(1), 31–43.
48. Mitrou, N. M., Lyberopoulos, G. L., & Panagopoulou, A. D. (1993). Voice and data integration in the air-interface of a microcellular mobile communication system. *IEEE Transactions on Vehicular Technology, 42*(1), 1–13.

Author Biographies

Polychronis Koutsakis was born in Hania, Greece, in 1974. He received his 5-year Diploma in Electrical Engineering in 1997 from the University of Patras, Greece and his M.Sc. and Ph.D. degrees in Electronic and Computer Engineering in 1999 and 2002, respectively, from the Technical University of Crete, Greece. He was a Visiting Lecturer at the Electronic and Computer Engineering Department of the same University for three years (2003–2006). Since July 2006 he is an Assistant Professor at the Electrical and Computer Engineering Department of McMaster University, Canada. His research is funded by the National Sciences and Engineering Research Council of Canada (NSERC) and by the Ontario Research Fund (ORF). His research interests focus on the design, modeling and performance evaluation of computer communication networks, and especially on the design and evaluation of multiple access schemes for multimedia integration over wireless networks, on call admission control and traffic policing schemes for both wireless and wired networks, on multiple access control protocols for mobile satellite networks, wireless sensor networks and powerline networks, and on traffic modeling.

Dr. Koutsakis has authored more than 75 peer-reviewed papers in the above mentioned areas, has served as a Guest Editor for an issue of the ACM Mobile Computing and Communications Review, as TPC Chair for the 4th ACM WMUNEP 2008, and serves as a TPC member annually for conferences such as IEEE GLOBECOM, IEEE ICC, IEEE LCN, and as a reviewer for most of the major journal publications focused on his research field. He is a member of the IEEE.

Moisis Vafiadis was born in Elefsina, Greece, in 1980. He received his 4-year Diploma in Electronics at the Technological Educational Institute of Crete, Greece. He is currently working towards the M.Sc. degree in the Department of Electronic and Computer Engineering of the Technical University of Crete, Greece. His current research interests focus on the design, modeling and performance evaluation of wireless personal communication networks. He has authored five papers in the above mentioned areas.

Aggelos Lazaris was born in Athens, Greece, in 1982. He received his 5-year Diploma in Electronic and Computer Engineering from the Technical University of Crete, in 2006. He is currently working towards the M.Sc. degree in the same department. His research interests include video encoding and transmission; traffic-modeling; resource allocation for wireless networks. He has authored eight papers in the above-mentioned areas.