International Journal of Wireless Information Networks, Vol. 7, No. 2, 2000

On Multiple Traffic Type Integration over Wireless TDMA Channels with Adjustable Request Bandwidth

P. Koutsakis[1] and M. Paterakis[1]

A new medium access control (MAC) protocol for mobile wireless communications is presented and investigated. We explore, via an extensive simulation study, the performance of the protocol when integrating voice, video and data packet traffic over a wireless channel of high capacity (referring to an indoor microcellular environment). Depending on the number of video users admitted into the system, our protocol varies: a) the request bandwidth dedicated to resolving the voice users contention, and b) the probability with which the base station grants information slots to voice users, in order to preserve full priority for video traffic. We evaluate the maximum voice capacity and mean access delay, as well as the aggregate channel throughput, for various voice and video load conditions, and the maximum voice capacity, aggregate channel throughput and average data message delays, for various video, voice and data load conditions. As proven by the comparison with a recently introduced efficient MAC scheme (DPRMA), when integrating voice and video traffic our scheme obtains higher voice capacity and aggregate channel throughput. When integrating all three traffic types, our scheme achieves high aggregate channel throughput in all cases of traffic load.

KEY WORDS: Mobile wireless communications; voice-video-data integration; MAC protocol.

INTRODUCTION

High-speed packet-switched network architectures will soon have the ability to support a wide variety of multimedia services, the traffic streams of which will have widely varying traffic characteristics (bit-rate, performance requirements). The main goal of wireless communication is to allow the user access to the capabilities of the global packet-switched network at any time without regard to location or mobility. Current and future wireless networks are and will be based on the cellular concept. System capacity can be increased by:

a) using a cellular structure with a cell size as small as possible (microcells) to increase frequency reuse. Microcell (picocell) diameters are usually of the order of a few hundred (dozen) meters, therefore the round-trip propagation delay within a microcell is negligible (of the order of 1 μs or less).
b) using efficient medium access control (MAC) protocols to exploit the variations in access and service required by disparate sources.

A well designed multiple access protocol will reduce system costs by maximizing system capacity, integrating different classes of traffic (e.g., voice, data and video, as opposed to today's picture, where wireless networks are optimized for voice communications only), and satisfying the diverse and usually contradictory quality of service (QoS) requirements of each traffic classs (such as voice packet dropping probability, voice packet access delay, video packet dropping probability and data message delay) whilst apportioning the limited radio channel bandwidth among them.

[1] Department of Electronics and Computer Engineering, Telecommunications and Information & Computer Networks Laboratories, Technical University of Crete, Chania, Greece. Tel.: +30 821 37225, +30 821 37258; Fax: +30 821 37202. e-mail: koutsaki@grecian.net, {polk,pateraki}@telecom.tuc.gr

In this work, we design and evaluate a multiple access scheme which multiplexes voice traffic at the vocal activity (talkspurt) level to efficiently integrate voice (Constant Bit Rate, CBR On/Off Traffic), video (Variable Bit Rate, VBR) and bursty data traffic in indoor (high capacity channel) picocellular environments.

Within the picocell, spatially dispersed source terminals share a radio channel that connects them to a fixed based station. The base station 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 base station is the sole transmitter on the downlink channel, it is in complete control of the downstream traffic, using Time Division Multiple Access (TDMA) to relay information to the users. Thus, we focus on the uplink (mobiles to base station) channel, where a MAC scheme is required in order to resolve the source terminals contention for channel access.

We assume that voice, video and data packet traffic is generated by mobile users who access the network with small, lightweight and low-power devices (i.e., in the category of low tier PCS [1]).

Speech alternates between periods of talk (talkspurts) and silence. Thus, voice terminals only require channel access during talkspurt and the time periods corresponding to silence gaps within a conversation can be used to transmit packets from other source terminals (i.e., multiplexing occurs at the talkspurt level). Both voice and video packet delay requirements are strict, because delays in both types of communication are annoying to a listener or viewer. Whenever the delay experienced by a voice or a video packet exceeds the corresponding maximum delay, the packet is dropped. Speech can withstand a small (1–2%) amount of dropped packets without suffering large quality degradation [2], at least one which can be perceived by humans. Video traffic is even less tolerant in the amount of dropped packets that it can withstand (0.01–0.02%) [13, 25]. On the other hand, data applications are more tolerant of delays (delays of up to 200 ms are often acceptable), but 100% delivery of correct packets is often required (e.g. in the case of a file transfer [3].

The paper is organized in two parts. Part 1 includes the work done for integrating voice and video packet traffic. Part 2 presents the work done for integrating all three types of traffic. In Part 1, Section 1 we present the channel model, the voice and video traffic models, the actions of voice and video terminals and the base station scheduling, and the transmission protocol for voice traffic. In Section 2 we introduce the experimental system design parameters. Section 3 is a brief presentation of a protocol named DPRMA [13, 25], with which we compare our voice-video integration results. Section 4 presents our simulation results along with discussion upon them and comparisons with DPRMA. The comparison shows our scheme's clearly better performance. In Part 2, Section 1 we introduce the new system model and design parameters. Section 2 presents our voice, video and data traffic integration results. Finally, Section 3 contains our concluding remarks for all the work presented in this paper.

PART 1: VOICE-VIDEO INTEGRATION

1. System Model

In this section, we present the structure of the channel frame, the actions of the voice and video terminals, and the scheduling algorithm followed by the base station (BS). Additionally, we explain the voice transmission protocol, and we introduce the voice and video traffic models.

1.1. Channel Frame Structure

The uplink channel time is divided into time frames of equal length. The frame duration is selected such that a voice terminal in talkspurt generates exactly one packet per frame. As shown in Fig. 1 (which presents an example of the channel frame structure), each frame consists of two *types* of intervals. These are the *voice request* intervals and the *information* intervals.

Within an information interval, each slot accommodates exactly one, fixed length, packet that contains voice or video information and a header. Voice request intervals are subdivided into mini-slots and each mini-slot accommodates exactly one, fixed length, request packet. The request must include a source identifier. Since we assume that all of the voice source state transitions occur at the frame boundaries,[2] we place all voice request intervals at the beginning of the frame, in order to minimize the vocie packet access delay.

Voice terminals do not exhaust their attempts for a reservation within the request intervals. *Any other free, at the time, information slot of the frame can be temporarily used as an extra request slot (ER slot)* for voice

[2] The explanation for this assumption will be given in section 2.2.

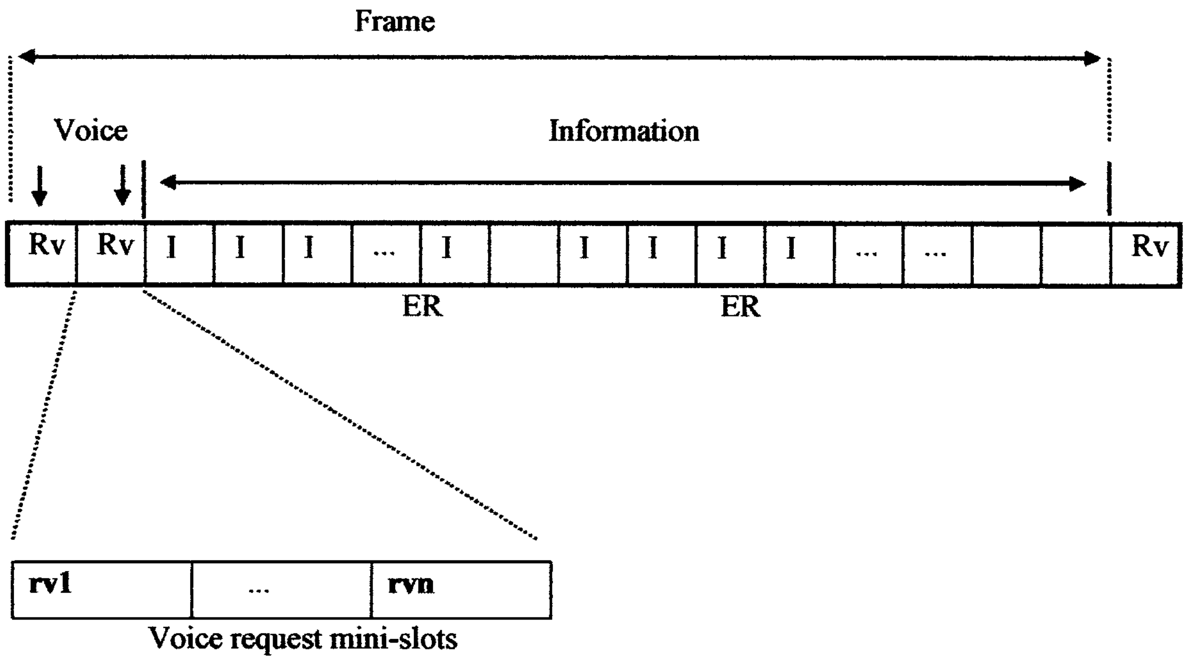

Fig. 1. An example of a channel frame structure showing the voice, data and information intervals within a frame.

terminals. Each one of the ER slots is further divided into mini-slots, just like a standard request (R) slot. This approach has been introduced and implemented in [4, 14].

By using more than one minislot per request slot, a more efficient usage of the available request bandwidth is possible. The concept of reserving a minimum bandwidth for voice terminals to make reservations helps to keep the voice access delay within relatively low limits and gives clearly better performance than the PRMA [7] and quite a few PRMA-like algorithms, such as DPRMA [13, 25], where the absence of request slots leads to a continuously decreasing probability of finding available information slots as traffic load increases, and hence to greater access delays. As shown from our results and also pointed out in [4], a request bandwidth of less than 3% of the total bandwidth is usually sufficient for high system performance (in the sense that is suffices for the requesting terminals to transmit their requests, while at the same time it does not consume a large portion of the bandwidth and leaves enough "space" for terminals with a reservation to transmit their information packets). No request slots are used for the video terminals, because of two reasons, which will be analyzed in paragraph 2.4.

1.2. Voice Traffic Model

Our primary voice traffic model assumptions are the following:

1) Voice terminals are equipped with a voice activity detector [6, 7]. Voice sources follow an alternating pattern of talkspurts and silence periods (on and off), and the output of the voice activity detector is modeled by a two-state discrete time Markov chain.
2) All of the voice source transitions (e.g., talk to silence) occur at the frame boundaries. This assumption is reasonably accurate, taking into consideration that the duration of a frame is equal to 12 ms here, while the average duration of the talksport and silence periods exceeds 1 sec.
3) The number of active voice terminals, N, in the system is assumed to be constant over the period of interest. This is because the changes in the number of calls are usually on the order of tens of seconds, while the frame duration is on the order of tens of milliseconds [8].
4) The voice delay limit is equal to 40 ms.

5) The channel is error-free and without capture.
6) Reserved slots are deallocated immediately. This implies that a voice terminal holding a reservation signals the BS upon the completion of its talkspurt, by setting a special bit in the header of its last talkspurt packet.

1.3. *Video Traffic Model*

We adopt the same video traffic model with the one in [13, 25]. This model is based upon work done by Heyman, et al [21, 22]. In this study of actual videoconferencing traffic, video frames (VFs) were found to be generated periodically and to contain a varying number of cells in each frame. The distribution of the number of cells per VF was found to be described by a gamma (or equivalently negative binomial) distribution. A Markov chain model can be constructed that demonstrates the transition from one state to the next A "state" represents the number of video packets (cells) that a video frame contains. The transition matrix is computed as:

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

where $\boldsymbol{I}$ is the identity matrix, ρ is the autocorrelation coefficient (0.98459 from [23]), and each row of the $\boldsymbol{Q}$ matrix is composed of the probabilities $(f0, \ldots, fK, \boldsymbol{F}K)$. The quantity $f\boldsymbol{K}$ has the negative binomial distribution and represents the probability that a video frame contains k cells. The value of K in equation 1 represents the peak cell rate and $\boldsymbol{F}K = \Sigma k > KfK$. An illustration of this model is displayed in Fig. 2.

The statistics for video conferencing traffic that were obtained in [21], were the result of coding a video sequence with a modified version of the H.261 standard. The results showed a peak cell generation rate of 220 cells/VF, an average generation rate of 104.8 cells/VF, and a standard deviation of 29.7 cells/VF. The cell size was taken equal to 48 bytes, which is equivalent to the ATM cell size. New VFs are assumed to arrive every 40 msecs (i.e., 25 VFs per second).

1.4. *Actions of Voice and Video Terminals and Base Station Scheduling*

Voice terminals with packets, and no reservation, contend for channel resources using a random access

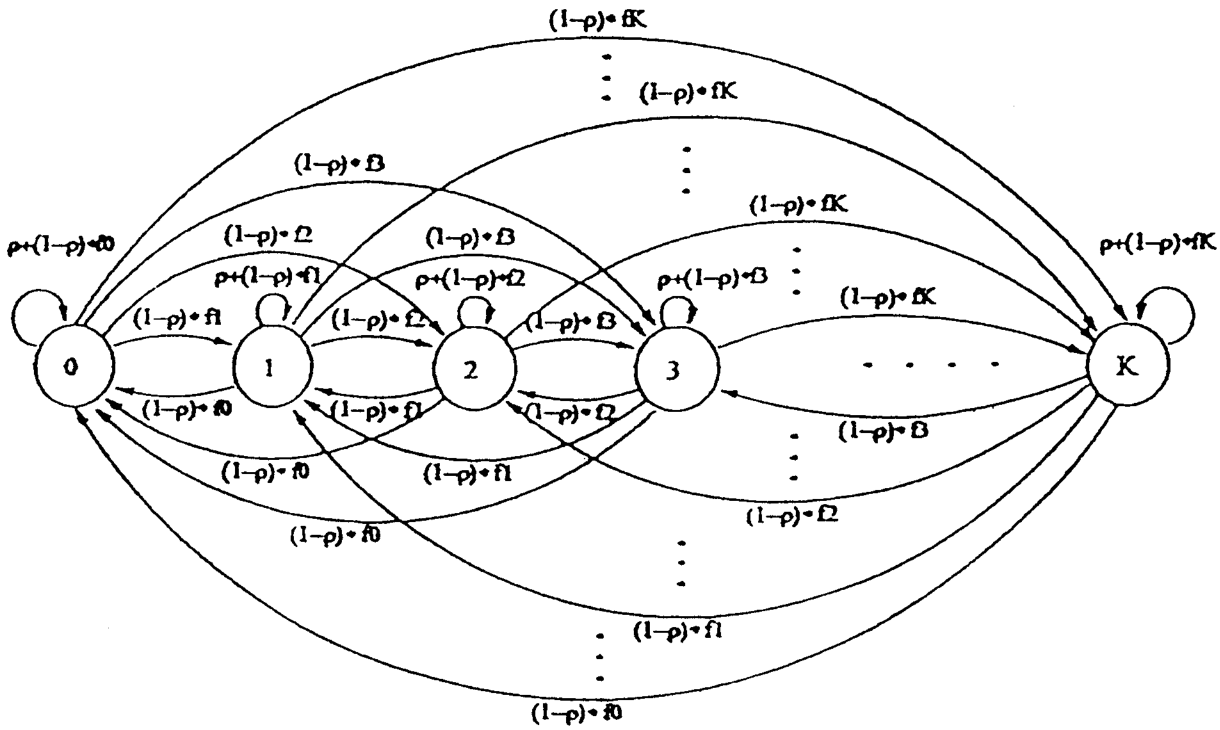

Fig. 2. Markov model for video.

protocol to transmit their request packets only during the voice request intervals. The base station broadcasts a short binary feedback packet at the end of each mini-slot indicating only the presence or absence of a collision within the mini-slot (collision C) versus non-collision (NC)). Since the feedback packet is short (several bits) and the propagation delay within a picocell is negligible, we assume that the feedback information is immediately available to the terminals (i.e., before the next mini-slot). Upon successfully transmitting a request packet the terminal waits *until the end of the corresponding request interval* to learn of its reservation slot (or slots). If unsuccessful within the request intervals of the current frame, the terminal attempts again in the request intervals of the next frame. A terminal with a reservation transmits freely within its reserved slot. Generally, a terminal that fails to transmit a request tries again in successive frames until it succeeds. However, since voice packets that age beyond the voice delay limit are dropped, a voice terminal may stop transmitting requests without ever succeeding, because all of its packets have timed out and it has transitioned into silence.

Video terminals, as already mentioned in paragraph 2.1 and shown in Fig. 1, do not have any request slots dedicated to them. This happens for two reasons:

1) Video sources "live" permanently in the system, they do not follow an ON-OFF state model like voice sources.
2) Video traffic a multi-state Markov model, in which however state transitions do not occur very often.

Thus, there is no need for granting request bandwidth to the video terminals, as it would be wasted in most cases. *Video terminals convey their requirements to the base station by transmitting them within the header of the first packet of their current video stream.*

To locate channel resources, the BS maintains a dynamic table of the active terminals within the picocell. Information within the table might include, for example, the terminal identifier, the channel resources allocated, and quality of service parameters. Upon successful receipt of a voice request packet, the BS provides an acknowledgment and queues the request. The BS allocates channel resources *at the end of the corresponding request interval*, and follows a different allocation policy for video terminals than that for voice terminals.

Video terminals have absolute priority in acquiring the slots they demand. If a full allocation is possible, the BS then proceeds to allocate any still available information slots to the requesting voice terminals. Otherwise, if a full allocation is not possible, the BS grants to the video users as many of the slots they requested as possible (i.e., the BS makes a partial allocation). The BS keeps a record of any partial allocations so that the remaining requests can be accommodated whenever the necessary channel resources become available. In either allocation type case, 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 video frame (VF) arrives.

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. This is a choice we make, in order to prevent a percentage of the voice terminals (the talkspurt of which has a mean duration of 1 sec, i.e. more than 8 channel frames) from entering the system. If this happened, voice terminals would keep their dedicated slots for the whole duration of their talkspurt, and thus video terminals, which need many slots, would not find enough slots to transmit in, and the particularly strict video QoS requirements would be violated. *In order to implement this idea of preventing a percentage of voice terminals from immediately entering the system, the BS in our scheme allocates a slot to each requesting voice terminal with a probability* $\boldsymbol{p}^*$. 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 where the resources needed to satisfy a voice request are unavailable. Voice terminals with queued requests must continuously monitor the base-to-mobile channel. Upon call transmission completion, or when an active voice terminal exits the picocell (handover) the BS will delete the table entry after some prescribed period of time. Within each priority class, the queuing discipline is assumed to be First Come First Served (FCFS).

Finally, in order to preserve the video QoS within the desired strict limits, we enforce the following scheduling policy for the video terminals:

As already mentioned, video terminals acquire from the BS the earliest available information slots within a frame and they can keep transmitting in these slots in the subsequent frames, until the arrival of the next VF. Taking into account the fact that new VFs arrive every 40 msecs and that the channel frame duration is 12 ms (i.e., a VF arrives every 3.33 channel frames), we understood that it is quite possible that certain slots dedicated to a video terminal may happen to be placed after the period of arrival (PA) of the next VF (see an example

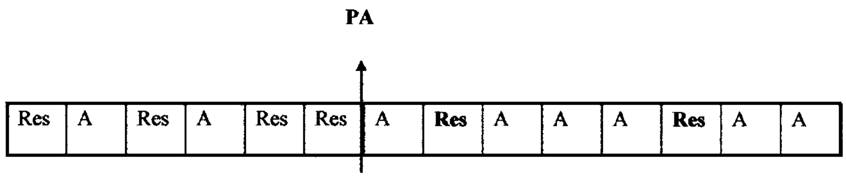

Fig. 3a. Status of information slots at the beginning of a frame in which a new VF will arrive.

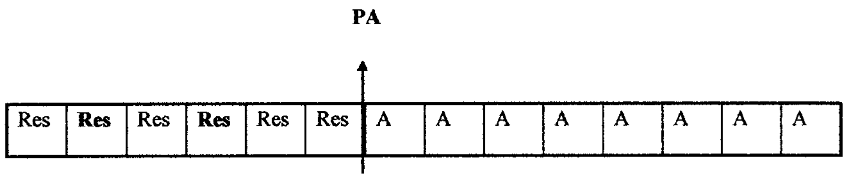

Fig. 3b. "Reshuffling" done by the BS.

in Fig. 3a, where two of the slots allocated to a video terminal are placed after PA and they are marked with bold characters). Thus, in order to avoid an unnecessary dropping of video packets, *the BS in our scheme performs a "reshuffling."* In each frame where a new VF is expected, the BS first finds all the availble slots in the frame which precede PA, and then replaces as many as possible of the slots dedicated to the video terminal after PA with an equal number of available slots before PA (see Fig. 3b). This way, video packets are lost only if the available slots before PA are less than the dedicated video slots after PA, or if there aren't enough available slots to accommodate the VF of a video terminal in the first place.

1.5. Transmission Protocol

Quite a few reservation random access algorithms have been proposed in the literature, for use by contending voice terminals to access a wireless TDMA channel (e.g., PRMA [7], Two-Cell Stack [3, 18], Controlled Aloha [3, 17], Three-Cell Stack [9]). In our study, we adopt the *two-cell stack* reservation random access algorithm, due to its operational simplicity, stability and relatively high throughput when compared to the PRMA (Aloha-based) ([10]) and PRMA-like algorithms, such as [4, 13, 15]. In this protocol, the contending set of voice terminals is split probabilistically into two equiprobable subsets *at the beginning of a frame.*[3] Only one of these subsets is transmitted in the first voice request minislot.

[3] Notice that the voice terminals split *before* the beginning of the voice contention. Our results for the algorithm without this initial split showed the voice capacity to be lower by several terminals.

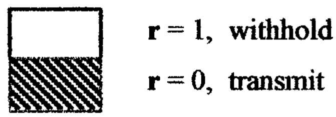

Fig. 4. Visualizing the two-cell stack algorithm.

The operation of the collision resolution mechanism of the protocol can be visualized by a two-cell stack (as shown in Fig. 4), where in a given voice request minislot the bottom cell contains the transmitting terminals, and the two cell contains the withholding terminals. If the transmitting set is found to contain more than one terminal (i.e., a collision occurs within the corresponding minislot), it is split probabilistically into two subsets (with equal probability), one of which remains in the transmitting cell while the other one joins the withholding cell. The end of the terminal contention for the request minislots is uniquely identified by the occurrence of two consecutive non-collisions.

2. System Parameters

We use computer simulations to study the performance of our MAC scheme. The simulations were conducted with the use of the C programming language in UNIX SUN workstations, and with the parameters contained in Table I. Each simulation point is the result of an average of 10 independent runs, each simulating 305,000 frames (the first 5,000 of which are used as warmup period).

The channel rate is 9.045 Mbps (from [13, 25]). The 12 ms of frame duration accommodate 256 slots. *As shown in Fig. 5, the number of voice request slots is not*

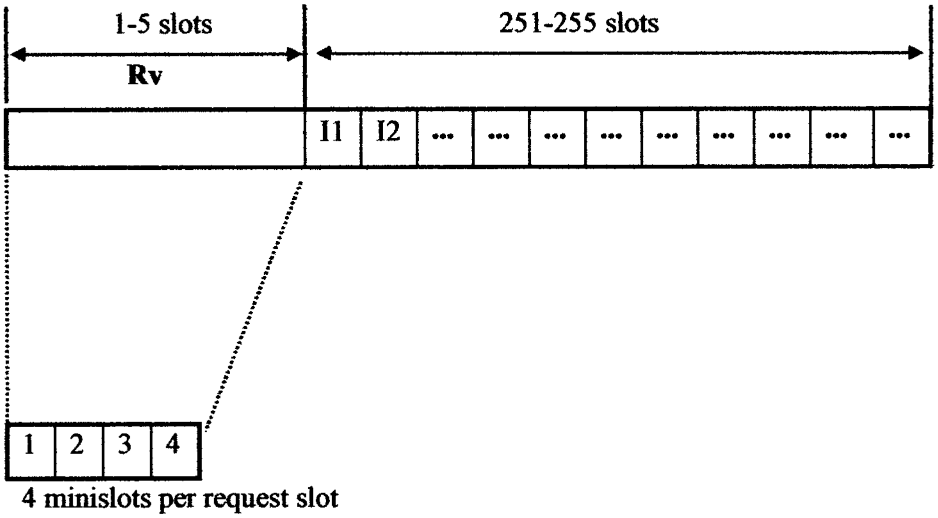

Fig. 5. Frame structure for the 9.045 Mbps channel.

Table I. Experimental System Parameters

Design parameters	
Channel rate (Mbps)	9.045
Speech codec rate (Kbps)	32
Frame duration (ms)	12
Slots per Frame	256
Slot duration (μs)	46.875
Request slots per frame	1–5
Minislots per request slot	4
Packet size (bytes)	53 (5 header)
Voice delay limit (ms)	40
Mean talkspurt duration (s)	1.0
Mean silence duration (s)	1.35
Maximum voice dropping probability	0.01
Peak video bit generation rate (Mbps)	2.112
Average video bit generation rate (Mbps)	1.006
Standard deviation of video bit rate (Mbps)	0.285
Video delay limit (ms)	40
Maximum video dropping probability	0.0001

fixed in the scheme. It depends on the number of video sources admitted into the system,[4] *and it varies accordingly between 1 and 5 slots (see Table II).*

Even for the case where 5 request slots are needed,

Table II. Adjustable Voice Request Bandwidth Depending on the Number of Video Users

Number of video users	Number of request slots
6	1
5	1
4	2
3	2
2	3
1	4
0	5

this corresponds to a 1.95% request bandwidth only, which is well within the desired range (<3%). We should note that:

1) In our design, we chose the number of minislots per request interval (4), to allow for guard time and synchronization overheads, for the transmission of a generic request packet (e.g., 40 bits long) that contains the source identifier, along with some data (e.g., priority, slots required, etc.), and for the propagation delay within the picocell (0.1 μs for a picocell with radius 20 m).
2) Because of assumption 3 of our voice traffic model, all voice request intervals are located at the beginning of each frame.
3) The maximum transmission delay for video packets is set to 40 msecs, 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 maximum trans-

[4] The channel bandwidth consumed by each video source is large, and thus, when we examine cases with a small number of video sources, the system can accommodate a significantly larger number of voice sources (because each one of them consumes much less information bandwidth than a video source). In this case, more voice request slots are needed in order to allow voice sources to enter the system without significant dropping of voice packets.

mission delay for voice packets is also set to 40 msecs.

4) The allowed voice packet dropping probability is set to 0.01, whereas the allowed video packet dropping probability is set to 0.0001.

3. DPRMA

In order to justify the better performance of our scheme, we will explain the four differences of DPRMA [13, 25] from our scheme.

The first difference exists in the scheduling mechanism for video sources. The BS does not grant the earliest available information slots. 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 is the use, in DPRMA, of certain transmission rates for the video 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 7 threshold levels, and, respectively, 7 transmission rates.

The third difference is that DPRMA does not use either voice request slots or our idea of ***p****, but adopts a PRMA-like approach for voice users, by allowing them to compete for the available information slots by transmitting their packets according to a probablity, PTv.

The fourth difference is that, in DPRMA, both voice and video users waste one slot when giving up their reservations. This does not happen in VVI, because of the VAD used for device 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 only convey at times a reservation request rate in order to keep the content of their video packet buffers below certain thresholds).

Table III. Comparison of Results Between VVI and DPRMA

Number of video users	Maximum voice capacity (voice terminals)		Channel throughput (%)		
	VVI	DPRMA	VVI	DPRMA	Probability p^*
6	3	0	74.2	73.7	0.0072
5	110	99	79.6	77.7	0.03
4	205	187	82.9	80.0	0.06
3	295	291	85.5	84.8	0.085
2	386	385	88.2	88.0	0.128
1	475	475	90.6	90.6	0.18
0	587	563	96.7	92.7	1

4. Results and Discussion

Table III presents the results of both our scheme (VVI, Voice-Video-Integration) and DPRMA, for the maximum voice capacity achieved by each scheme, given the number of video users in the system. The last column of the Table shows the allocation probability ***p**** for which VVI achieves this capacity.

The reasons for which VVI excels in comparison with DPRMA are (in respect to the four differences between the two schemes, presented in paragraph 4):

1) The mechanism proposed for the video slot scheduling in DPRMA is less efficient than that of VVI, because the random spreading of the allocated slots leads to the loss of video packets scheduled to be transmitted in slots following their corresponding PA, whilst slots preceding PA are available. In other words, *due to our reshuffling mechanism* VVI achieves higher bandwidth utilization than DPRMA.
2) The combination of differences two and four in section 3, makes clear that our scheme again leaves less unused slots than DPRMA.
3) By using the two-cell stack random access algorithm, VVI allows the voice users to make their requests to the BS more effectively than

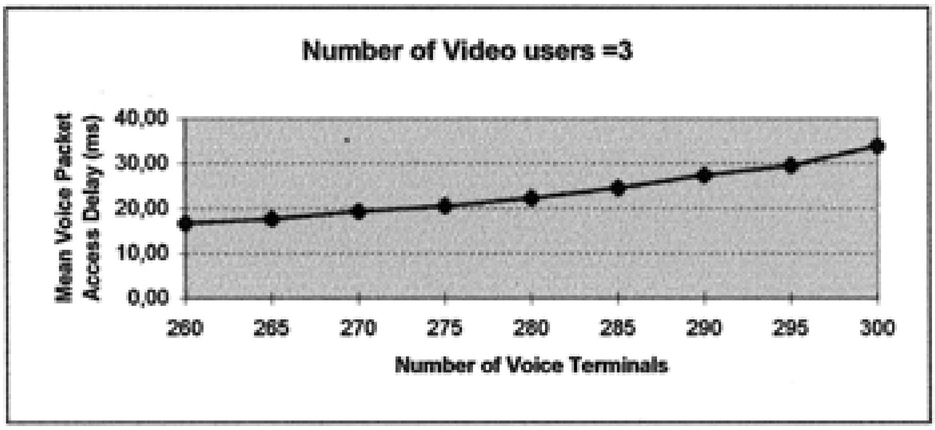

Fig. 6. Steady state mean voice access delay versus the number of active voice terminals, when 3 video users have been admitted in the system.

DPRMA, which uses the PRMA algorithm for that purpose. The "obstacle" put to the voice users in acquiring a slot (***p****) is set in VVI after they have sent their request to the BS, they will wait in the queue at the BS for a possible slot allocation *without having to further contend*. On the contrary, in DPRMA the "obstacle" is set by using a small transmission probability when implementing the PRMA protocol for voice slot contention. The latter approach is less effective, because voice users must *repeatedly enter contention* in order to reserve a slot, thus leaving more slots unused. Additionally, the use of ER slots helps VVI "exploit" certain available slots that DPRMA leaves unused.

All these factors explain the better performance of our scheme. In addition, our mechanism is much less complicated than that of DPRMA. DPRMA imposes a "burden" both on the BS (for calculating Pa each time it needs to allocate a slot), and on the terminals (for making the necessary adjustments to their transmission rate each time a threshold is surpassed). The mechanism in VVI, on the other hand, does not adopt either of these two approaches, and the use of the ***p**** probability for allocating slots to voice terminals which we introduce, is a simple task for the BS to perform. The calculation of ***p**** for each case presented in Table III is done via simulation and is performed only once. Then, the BS uses this value of ***p**** for allocating voice slots.

Finally, Figs. 6 and 7 present the cases of the number of video terminals being equal to 3 and 5, respectively, and show the mean voice access delay, as a function of the number of voice terminals (VTs) in the system. We observe in both figures that the mean voice packet access delay is much smaller than the 40 msecs limit (around 30 msecs), for the maximum voice capacity in each case. This means that the system would be capable of supporting a higher number of voice terminals in the case where the maximum allowed video dropping probability limit was higher than 0.01%. It is our voice slot allocation policy (with ***p****) which "forces" the voice dropping probability to reach the 1% limit, for the numbers of video terminals presented in Table III, and this approach is used in order to keep the video dropping probability less than 0.01%.

It should be noted that a comparison with DPRMA's respective mean voice access delay results is not possible, as they are not presented by the authors, in neither [13], or [25].

PART 2: VOICE-VIDEO-DATA INTEGRATION

1. New System Model

1.1. Channel Frame Structure

In the case of the integration of all three types of traffic, *we introduce the idea* [24] *that the request slots* (1–5, depending on the case, as explained in Part 1) *can be shared by voice and data terminals (first by voice terminals and, after the end of voice contention, by data terminals)*, in order to optimize the use of the request bandwidth. As expected from the bursty nature of data traffic and shown also from the simulation results in sec-

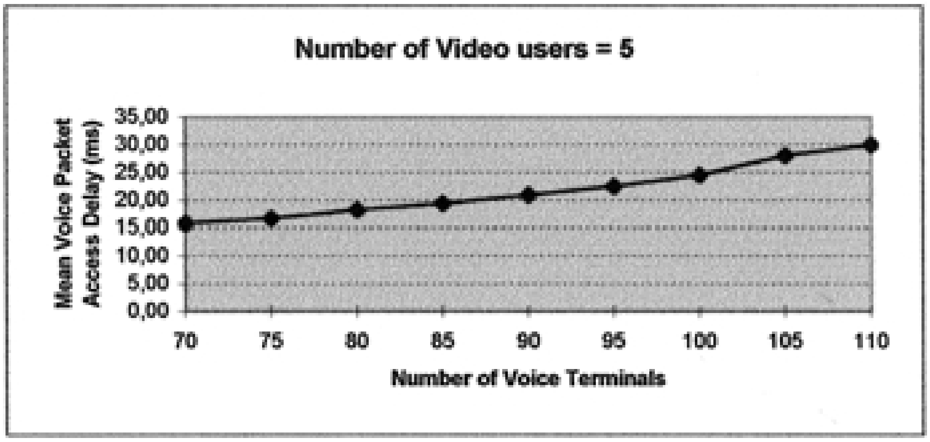

Fig. 7. Steady state mean voice access delay versus the number of active voice terminals, when 5 video users have been admitted in the system.

tion 2, when integrating all three traffic types (given the bursty nature of two of them), it is necessary to significantly decrease the number of voice users in the system (compared to the voice capacity achieved in Part 1), in order to accommodate data traffic. Thus, the number of request slots selected in Part 1 to accommodate the voice requests in each case suffices for accommodating now both voice and data requests. In addition, the ER slots can be used by both voice and data terminals, with priority given to voice terminals.

1.2. Data Traffic Model

We adopt the following data traffic model:

Data traffic has low priority compared to voice traffic, and data messages are generated by a large unknown number of data terminals (theoretically infinite). The aggregate message arrivals are Poisson distributed with mean λ messages per frame. Additionally, we assume that the messages vary in length according to a geometric distribution with parameter q and mean $B = 1/q$. B is expressed in packets per message, and the steady state data rate packet arrival is equal to λB packets/frame (B is equal to 8 in our study).

An upper limit on the mean data message delay, equal to 200 ms, is assumed.

1.3. Data Packets Transmission Protocol

The actions of the data terminals, in order to acquire access to the channel resources, are similar to those of voice terminals. The *two-cell stack* blocked access collision resolution algorithm [5, 19] is adopted for use by the data terminals in order to transmit their data request packets. This algorithm is of window type, with FCFS-like service.

1.4. Base Station Scheduling

The base station scheduling for video and voice packet traffic is identical to the one described in Part 1. Data terminals have the lowest priority in acquiring the slots they demand. Each data user is allowed to reserve *just one slot per frame*. This choice is explained in the next section, where we discuss our results.

2. Voice-Video-Data Integration Results and Discussion

Table IV presents the results for the maximum voice capacity, obtained for various values of the number of video users (0, 1, 2, 4, 5, 6) and the data message arrival rate λ. It is clear, from the results presented in the table, that the smoothest transitions (decreases) for the maximum voice capacity take place when no video users exist in the system. This is easily explained by the fact that video traffic is quite bursty. Thus, the number of voice terminals has to be drastically decreased as the message arrival rate increases, in order to cope with the burstiness of the video traffic mainly but also with the burstiness of the data traffic, and still be able to preserve the QoS requirements for each traffic type. This explains our "defensive" choice of not granting, in any case, more than one information slot per frame to each data user, as this would lead to a further decrease of the

Table IV. Maximum Voice Capacity for a Set Number of Video Users and Set Data Message Arrival Rate

λ (mes./frame)	Second row: Number of video users Next rows: Maximum voice capacity and throughput (%)											
	0		1		2		4		5		6	
0.1	566	96.3	464	91.1	377	88.6	201	83.2	104	79.3	0	74.3
0.5	557	96.0	433	87.1	350	85.3	185	82.1	90	78.2	x	x
1.0	549	96.3	410	84.9	330	83.5	164	80.1	70	76.5	x	x
1.5	539	96.2	400	84.8	314	82.4	150	79.4	57	75.9	x	x
2.0	530	96.2	385	83.8	300	81.6	136	78.6	44	75.3	x	x
2.5	521	96.3	357	82.4	287	81.3	124	78.2	34	75.2	x	x
3.0	510	96.0	362	83.1	277	80.9	115	78.2	27	75.6	x	x
3.5	503	96.4	345	81.8	265	80.5	104	78.0	16	75.3	x	x
4.0	494	96.5	338	82.2	258	80.9	94	77.9	6	75.2	x	x

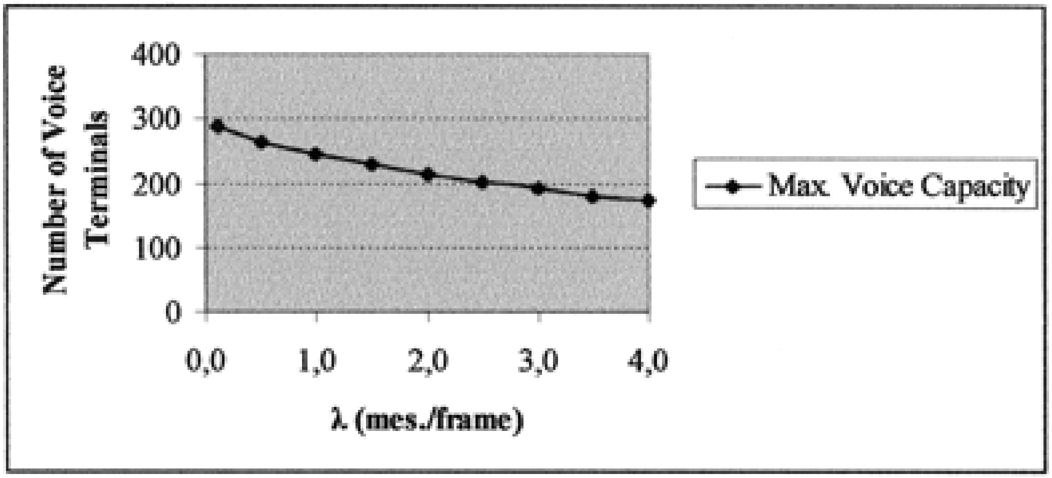

Fig. 8. Maximum voice capacity vs. data message arrival rate, when 3 video users are admitted in the system.

maximum voice capacity in order to preserve the video QoS requirements.[5]

The throughput achieved by our scheme is decreasing when the number of video terminals increases, as expected because of the bursty nature of the video traffic. Still, as shown in Table IV, it remains quite high for all cases, thus pointing out the good performance of the scheme.[6] The corresponding results for the case of 3 video users being accommodated in the system, are shown in Fig. 8, and all the above comments stand for them as well.

Both Figs. 9 and 10 show the mean data message delay[7] as a function of the number of voice terminals (VTs) in the system, and the average channel load (in packets/frame). Figure 9 presents the case of the number of video terminals being equal to 3 and the data message arrival rate equal to 2 messages/frame. Figure 10 presents the case of the number of video terminals being equal to 5 and the data message arrival rate equal to 0.5 messages/frame.

We observe in both figures that the average data message delay is much smaller than the 200 msecs limit, for the maximum voice capacity in each case. This means that the system would be capable of supporting a higher data message arrival rate for the same voice capacity, in the case where the maximum allowed video

[5] This choice was made after quite a few attempts to grant more than one information slot/frame to data users. The simulation results have shown this choice to bring the best results.

[6] The symbol "x," used in the last two columns of the Table, stands for the system's inability to accommodate the corresponding traffic.

[7] The data message delay is defined as the time period from the instant a data terminal generates a message, until it completes the transmission of the last packet of its message in a reserved slot.

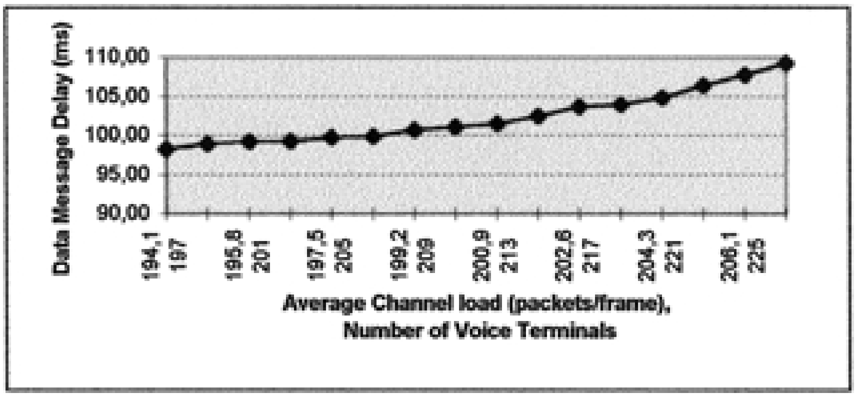

Fig. 9. Average data message delay vs. number of voice, terminals for λ = 2 messages/frame and number of video users admitted in the system = 3.

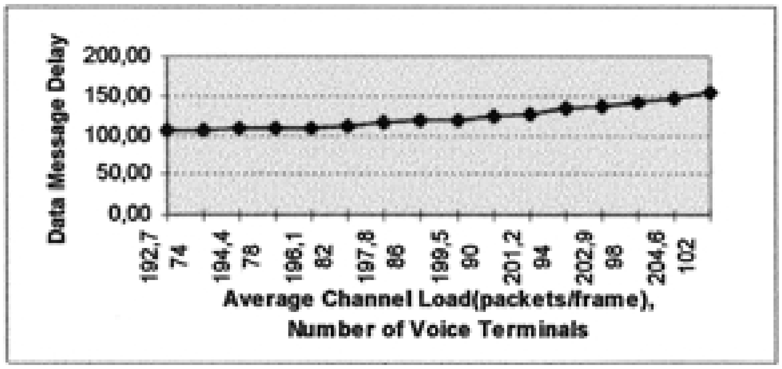

Fig. 10. Average data message delay vs. number of voice terminals, for λ = 0.5 messages/frame and number of video users admitted in the system = 5.

dropping probability limit would be higher than 0.01%, which is again the most restraining parameter for the scheme.

FINAL CONCLUSIONS

In this paper we have proposed and evaluated a new multiple channel access control scheme for integrating voice, video and data packet traffic in a high capacity picocellular environment. Video traffic is offered absolute priority over voice and data traffic, due to its more stringent quality of service requirements.

Via an extensive simulation study we demonstrate that our scheme evidently excels when compared in voice-video integration to a recently introduced MAC scheme, called DPRMA. Furthermore, the proposed scheme achieves high throughput when integrating all three traffic types, despite the very restraining video dropping probability limit.

The results achieved by our scheme are a consequence of the combination of three factors: a) our voice slots allocation policy; b) our video slots scheduling policy; and c) the use of the unused information slots as extra request slots. These three factors help us increase (in comparison with DPRMA) the achieved voice capacity and aggregate channel throughput by "exploiting" all the available information slots within each frame in the best possible way.

REFERENCES

1. D. C. Cox, Wireless personal communications: What is it? *IEEE Pers. Commun.*, Vol. 2, No. 2, pp. 20–35, 1995.
2. B. Mukherjee, Integrated voice-data communication over high speed fiber optic networks, *IEEE Computer Magazine*, Vol. 24, No. 2, pp. 49–58, 1991.

3. A. C. Cleary, Voice-data channel access integration in third generation microcellular wireless networks: Design and performance evaluation, Ph.D. Dissertation, CIS Dept., University of Delaware, 1997.
4. N. M. Mitrou, Th. D. Orinos, and E. N. Protonotarios, A reservation multiple access protocol for microcellular mobile-communication systems, *IEEE Transactions on Vehicular Technology*, Vol. 39, No. 4, pp. 340–351, 1990.
5. M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, A full sensing window RA algorithm for networks with strict delay constraints, *Algorithmica*, Vol. 4, pp. 313–328, 1989.
6. P. T. Brady, A technique for investigating on-off patterns of speech, *Bell Sys. Tech. Journal*, 1965.
7. S. Nanda, D. J. Goodman, and U. Timor, Performance of PRMA: A packet voice protocol for cellular systems, *IEEE Transactions on Vehicular Technology*, Vol. 40, pp. 584–598, 1991.
8. T. Suda and T. Bradley, Packetized voice/data integrated transmission on a token passing ring local area network, *IEEE Transactions on Communications*, Vol. COM-37, No. 3, pp. 238–244, 1989.
9. A. C. Cleary and M. Paterakis, An investigation of reservation random access algorithms for voice-data integration on microcellular wireless environments, *International Journal Wireless Information Networks*, Vol. 2, No. 1, Jan. 1995, pp. 1–16.
10. A. C. Cleary and M. Paterakis, Design and performance evaluation of an RRA scheme for voice-data channel access in outdoor microcellular wireless environments, *Mobile Networks and Applications (MONET) Journal*, Vol. 2, No. 1, 1997, pp. 31–43.
11. J. M. De Vile, A reservation based multiple access scheme for a future universal mobile telecommunications system, *Proc. of the 7th IEE Conf. on Mobile and Personal Commun.*, Brighton, UK, pp. 210–215, 1993.
12. J. Dunlop *et al.*, Performance of a statistically multiplexed access mechanism for a TDMA radio interface, *IEEE Pers. Commun.*, Vol. 2, No. 3, pp. 56–64, 1995.
13. D. A. Dyson and Z. J. Haas, A dynamic packet reservation multiple access scheme for wireless ATM, *Proceedings of the IEEE MILCOM'97*.
14. N. M. Mitrou, G. L. Lyberopoulos, and A. D. Panagopoulou, Voice and data integration in the air-interface of a microcellular mobile communication system, *IEEE Transactions on Vehicular Technology*, Vol. 42, No. 1, pp. 1–13, 1993.
15. W. C. Wong and D. J. Goodman, A packet reservation multiple access protocol for integrated speech and data transmission, *IEEE Proceedings-1*, Vol. 139, pp. 607–612, 1992.
16. A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis, 2nd Ed.*, McGraw Hill Inc., 1991.
17. D. Bertsekas and R. Gallager, *Data Networks, 2nd Ed.*, Prentice Hall Inc., 1992.
18. A. C. Cleary and M. Paterakis, On the voice-data integration in third generation wireless access communication networks, *European Transactions on Telecommunications and Related Technologies*, Vol. 5, No. 1, pp. 11–18, 1994.
19. M. Paterakis and P. Papantoni-Kazakos, A simple window random access algorithm with advantageous properties, *IEEE Transactions on Information Theory*, Vol. IT-35, No. 5, pp. 1124–1130, 1989.
20. R. Rom and M. Sidi, *Multiple Access Protocols: Performance and Analysis*, Springer Verlag, 1990.
21. D. Heyman, T. V. Lakshman, A. Tabatabai, and H. Heeke, Modeling teleconference traffic from VBR video coders, *International Conference on Communications '94*, Vol. 3, pp. 1744–1748, 1994.
22. D. Heyman, A. Tabatabai, and T. V. Lakshman, Statistical analysis and simulation study of video teleconference traffic in ATM networks, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 2, No. 1, 1992.
23. H. Heeke, Statistical multiplexing gain for variable bit rate video codecs in ATM networks, *Proc. of the 4th Int. Packet Video Workshop*, Tokyo, Japan, March 1991.
24. P. Koutsakis and M. Paterakis, Near-optimal voice-data integration over third generation medium and high capacity wireless TDMA Channels, *Proc. of the IEEE Wireless Communications and Networking Conference 1999 (WCNC '99)*, New Orleans, USA.
25. D. A. Dyson and Z. J. Haas, A dynamic packet reservation multiple access scheme for wireless ATM, *ACM/Baltzer MONET Journal*, accepted 1999.

Polychronis Koutsakis was born in Hania, in 1974. He received his Diploma of Electrical Engineering from the University of Patras, in 1997, and his M.Sc. from the Department of Electronic and Computer Engineering of the Technical University of Crete, in 1999. He is currently working towards acquiring his PhD from the same Department. His research interests are focused on protocol design for wireless communication networks, on the effect of burst channel errors on protocol performance, as well as on queueing theory. He is a student member of the IEEE, a member of the Greek Chamber of Professional Engineers, and of the Greek Association of Electrical and Mechanical Engineers.

Michael Paterakis received his Diploma degree from the National Technical University of Athens, Greece, his M.Sc. degree from the University of Connecticut, and his Ph.D. degree from the University of Virginia, in 1984, 1986, and 1988, respectively, all in Electrical Engineering. Since 1995, he is a faculty member in the Department of Electronic and Computer Engineering at the Technical University of Crete, Greece, where he is currently a Professor and Department Chairman. He was an Associate Professor in the Department of Computer and Information Sciences at the University of Delaware, where he has been since September 1988. His research interests include computer communication networks with emphasis on protocol design, modeling and performance evaluation of broadband high speed networks, of multiple access wireless microcellular communication systems, and of packet radio networks; centralized and distributed multimedia information delivery systems; queuing and applied probability theory and their application to computer communication networks and information systems. He has published extensively in archival scientific journals, refereed conference proceedings and edited books, in the abovementioned technical areas. He served on the Technical Program Committees of the 1991 Intl. Conference on Distributed Computing Systems, the 1992 and 1994 IEEE INFOCOM Conferences, the 1997 IEEE Workshop on the Architecture and Implementation of High Performance Communication Subsystems (HPCS'97), and the 6th IFIP Workshop on Performance Modeling and Evaluation of ATM Networks (IFIP ATM'98). Professor Paterakis is a Senior Member of the IEEE. He is also a member of the IEEE Technical Committee on Computer Communications, the Greek Chamber of Professional Engineers, and the Greek Association of Electrical and Mechanical Engineers.