**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ**
**ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ**
**ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ**

# Πτυχιακή εργασία

## Subject

**Design and implementation of MAC protocol over a high speed wireless channel with noise**

από τον **Βαφειάδη Μωυσή**.

*Εκπονήθηκε υπό την επίβλεψη του Δρ. **ΠολυχρόνηΚουτσάκη***

# Summary of this work

In this paper we design and study the performance of a Multiple Access Control (MAC) scheme for the multiplexing and the integrated delivery of voice, video and bursty data traffic over a high-speed wireless TDMA channel. We compare our results with a previously proposed efficient MAC protocol and we show that, regardless of the much more severe (and realistic) channel conditions examined in our work, our scheme clearly outperforms the previously proposed protocol in terms of channel throughput.

*Αφιερωμένο σε όσους*
*πίστεψαν σε μένα ...*

# *Ευχαριστίες*

***Για την πραγματοποίηση της πτυχιακής αυτής θα ήθελα να ευχαριστήσω αρχικά τον επιβλέποντα καθηγητή μου, Δρ. Πολυχρόνη Κουτσάκη, ο οποίος με την άψογη συνεργασία μας, με βοήθησε στην επιτυχή και άρτια ολοκλήρωση της πτυχιακής μου εργασίας και παράλληλα μου έδειξε δρόμους για μελλοντικές ενέργειες και ασχολίες, δίνοντας μου πέρα από πολύτιμες γνώσεις και εφόδια τα οποία θα με ακολουθούν και θα με συντροφεύουν για την υπόλοιπη ζωή μου.***

***Επίσης, θα ήθελα να ευχαριστήσω όλους τους φίλους μου, οι οποίοι με βοήθησαν, κυρίως ηθικά, για την ολοκλήρωση  αυτής της εργασίας. Ήταν πάντα δίπλα μου σε ότι τους χρειαζόμουν, ακόμα και αν ήταν χιλιόμετρα μακριά.***

***Τέλος, οφείλω ένα μεγάλο ευχαριστώ στους γονείς και στα αδέλφια μου, οι οποίοι με στήριξαν τόσο ηθικά όσο και υλικά, και είναι δίπλα μου σε κάθε επιλογή και απόφαση που παίρνω  για την ζωή μου. Τους ευχαριστώ πολύ.***

***Σας Ευχαριστώ.***

# Contents

# Chapter 1

# Introduction

## 1.1. The field-Background

*I. Wired and Wireless Networks*

The coexistence of wired and wireless communication networks is a hint that none of the two network types is able to service, on its own, all of the users needs. The basic advantage of wireless networks is the user's ability to move, even during the use of the network services. Clearly, this is something that wired networks can not offer. This ability is of great importance to users who, either because of the nature of their work or due to recreation reasons, are constantly in motion and simultaneously, need to communicate. On the other hand, wired networks offer reliable and high-speed communication, thus giving the users the ability to exploit a large variety of services. Especially after the introduction of fiber optics, the transmission speed and reliability have increased considerably, so that a wide variety of services can be offered (e.g., video, high quality image and sound applications). On the contrary, the use of the air as transmission medium in wireless networks leads to limited channel bandwidth, which in turn means limited transmission speed, as well as a high error rate, which is not acceptable by most of the applications. For this reason, the applications supported until today by the wireless cellular communications networks involve mostly voice and low rate data transmission (e.g., Short Message System).

The recent technological developments in wireless communications, especially regarding the speed and quality of the transmission, have made possible the support of many more applications. These developments include:

- the capacity increase of wireless channels, mostly due to the improvement of the transmitter and receiver technologies
- the reduction of transmission power, which leads to greater independence for the wireless terminals
- the development of simpler and more easy-to-use equipment
- the reduction of the total cost for building and maintaining a wireless network [1].

Still, these technological achievements are not by themselves enough to efficiently support the new applications. The existing architectures and protocols have been designed to efficiently support the existing voice and low data rate applications, thus their use in new applications presents problems. For this reason, a new network design is required, both in the architecture and the protocol level. This way, the new applications will be satisfactorily supported, and the network resources will be used more efficiently.

The ATM (Asynchronous Transfer Mode) technique can solve many problems towards this direction. This technique, which is briefly described in the next section, has the ability to adjust to different traffic conditions and QoS (Quality of Service) requirements, therefore it can support a large variety of applications. Still, the fact that the original design of ATM did not include the support of wireless networks, makes certain adjustments necessary.

*II. Asynchronous Transfer Mode*

The ATM has been selected as the official transmission mode for BISDN (Broadband Integrated Services Digital Network). BISDN is a network architecture supporting a large applications’ spectrum (voice, image, multimedia, etc.). The term ″asynchronous″

is not referring to the actual transmission, which in most cases is synchronous, but to the way the available bandwidth is reserved. The channel time is divided in fixed size time slots, which are dynamically reserved by the various network uses, depending on their needs.

The ATM technique is defined with the help of a set of principles [2]:

- The information is transmitted with fixed length data units, which are called cells. The cells consist of a header and a data field. The cell structure is described below.
- ATM uses virtual connections to transmit information.
- The main use of the cell header is the identification of the cells which belong to the same connection.
- The basic unit of an ATM network is the ATM switch. An ATM switch can be characterized as a network device with many ports. Its role is to switch the cells it receives from the input ports to the appropriate output ports.
- The cell identification headers are of local importance only. They are not absolute addresses and they are translated in each ATM switch.

The cell size is 53 bytes. Of these, 5 bytes consist the header, and the remaining 48 bytes are used for information transmission (payload). The cell structure is presented in Figure 1. Each row represents one byte of the cell.

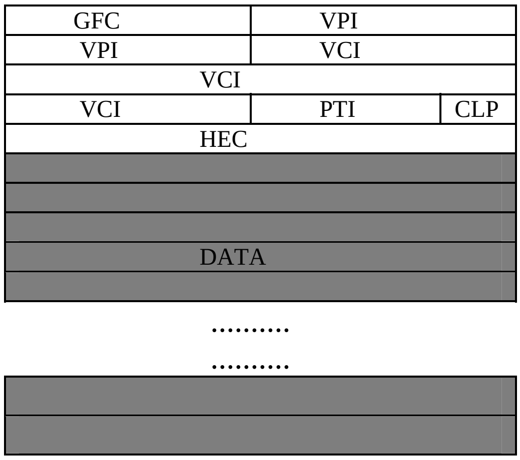


**Figure 1: The structure of an ATM cell.**

The use of the header fields is:

- **GFC (G**eneric **F**low **C**ontrol): (4 bits). It is used to regulate the information rate in an ATM network. Its full use is still an object of research.
- **VPI (V**irtual **P**ath **I**dentifier): (8 bits). It defines a set of virtual circuits, which are routed through the same path. VPI, together with VCI, determines the connection to which the cell belongs.
- **VCI (V**irtual **C**hannel **I**dentifier): (16 bits). It uniquely defines a connection between two end nodes.
- **PTI (P**ayload **T**ype **I**dentifier): (3 bits). It determines whether the data belong to a certain user or consist network administration information.
- **CLP (C**ell **L**oss **P**riority): (1 bit). It determines whether the cell is of low priority and thus can be dropped in case of congestion somewhere in the network.
- **HEC (H**eader **E**rror **C**ontrol): (8 bits). It is used by the end node, for error control of the cell's header.

To achieve high quality of service, the connections are classified in five classes [3]:

1. **C**onstant **B**it **R**ate (**CBR)**. These are connections which demand a constant bandwidth for their entire duration. The applications which use this type of connection are usually voice or circuit emulation applications and have strict requirements on the cell transmission delay (maximum value and jitter).
2. **r**eal-**t**ime **V**ariable **B**it **R**ate (**rt-VBR)**. This type of connections is mainly used for real time applications (i.e., applications which demand very low delays). In these connections, the required transmission rate, and consequently the bandwidth demands, vary over time. Cells which fail to be transmitted within the maximum permissible delay are considered of extremely low value for the corresponding applications, and are usually dropped.
3. **n**on-**r**eal-**t**ime **V**ariable **B**it **R**ate (**nrt-VBR)**. These are connections similar to the ones described above, but their delay requirements are loose (i.e., not strict). On the other hand, these connections have very small tolerance in packet losses, compared to the real-time VBR connections, and for this reason advanced control and error detection/ correction mechanisms are needed for these connections.
4. **A**vailable **B**it **R**ate (**ABR)**. The ABR connections have neither delay nor delay jitter requirements. Their only requirement from the network is that they are provided with bandwidth at least equal to a declared minimum value.
5. **U**nspecified **B**it **R**ate (**UBR)**. These connections are used for non-real-time applications, with neither delay or delay jitter requirements, and with no minimum bandwidth demand. Classic data computer communication applications, such as file transfers and electronic mail, belong to this class. Since there is no minimum bandwidth demand, these connections have the lowest priority and use the bandwidth that is left unused by all the connections of the other classes.

As a multiplexing technique, ATM has significant advantages in comparison to the Synchronous Transfer Mode (STM), which uses static bandwidth reservation. The advantages mainly focus on the efficient use of bandwidth. According to STM, the channel time is organized in frames of equal duration, which in turn are divided into time slots. For each connection, one time slot is reserved in each frame, and the information is transmitted within this slot. An STM channel is defined by the location of its corresponding time slot within the channel frame. Clearly, STM is very efficient for CBR connections. However, its efficiency falls dramatically when it has to support VBR connections, since is has to reserve for each connection bandwidth equal to its maximum transmission rate.

In an ATM network, the support of VBR connections is facilitated via the use of *statistical multiplexing*. With statistical multiplexing, the sum of the maximum transmission rates of all the active connections can surpass the channel bandwidth, while the available bandwidth is dynamically apportioned depending on the (variable) connection bandwidth requirements. With this method, a better usage of the network is achieved, along with the ability to support more concurrent connections. Still, the danger exists, that when the sum of the connections transmission rates surpasses the channel capacity, congestion might occur. In this case, some packets are either dropped, or temporarily kept in storage buffers, which results in increased packet delays. Some of the dropped packets must be retransmitted, which further aggravates the network congestion.

In order to deal with congestion, there is the need of traffic control algorithms, which must be capable to easily adjust to the different QoS requirements of the various connections. For example, voice applications have strict requirements regarding the maximum cell transmission delay and delay jitter. On the other hand, data applications usually demand very low error probabilities and retransmission of dropped cells, while their delay requirements are either loose or non-existent.

The traffic control algorithm is divided in three parts:

1. Call Admission Control (CAC)
2. Usage Parameter Control (UPC)
3. Congestion Control

With CAC, the network decides whether it will accept a new connection or not. The decision can be based on:

i) the characteristics of the traffic that the connection intends to introduce into the network,
ii) the connection QoS requirements
iii) the current network status

The traffic characteristics and the QoS requirements are defined through a set of parameters which are declared by the user when applying for the establishment of a connection. The most common parameters are the mean and peak cell rate, the burst size, the maximum cell delay tolerance and the maximum cell delay variation tolerance. Based on these parameters, the CAC algorithm decides whether it would accept the new connection or not. The algorithm should not be too strict, which would lead to low network utilization, nor too loose, thus creating congestion. If the connection is accepted, it is assumed that the network and the user agree on a traffic contract. Based on this contract, the network is bound to satisfy the user's QoS requirements, and the user is bound to operate according to its traffic description parameters [4].

However, in ATM networks bandwidth reservation is dynamic, so there is nothing that prevents a connection from violating the traffic contract and transmitting at a rate higher than the one stated. This could happen not only because of the user's bad intention, but also because of a miscalculation of the required bandwidth. Since the ATM cells corresponding to the additional rate might cause network congestion, a mechanism of usage parameter control is necessary. This mechanism controls the traffic which the user introduces into the network, and its aim is to protect the network and the other users from violations of the traffic contract, deliberate or not. A good usage parameter control mechanism should combine [5]:

- simplicity, so that it can be easily applied by the user
- responsiveness to traffic parameter violations
- tolerance, because of system inaccuracies.

Despite the use of call admission control and usage parameter control mechanisms, there still exists the possibility of congestion in some part of the network, because of temporary buffer overload. The aim of congestion control is to recognize the phenomenon and to implement mechanisms that will reduce its consequences. The congestion controls which have been proposed for use in ATM networks are preventive. The use of reactive congestion control methods is forbidden, mainly for two reasons:

i) the response of such methods to congestion is very slow, and this could be destructive to real-time applications, and

ii) the propagation delay is very large, compared to the very small cell transmission time. Thus the number of cells introduced in the network within a few propagation delays time after the congestion begins, can be very big.

Thus, even if the reactive congestion control mechanism reacts fast, congestion will further build up and continue for quite some time. For this reason, preventive methods have been suggested, in order to avoid the congestion problem [6,7,8].

*III. Wireless Networks*

The goal of wireless communication networks is to allow user access to the capabilities of the wired networks. The development of digital coding techniques and the continuous increase in integrated circuits density have made possible the realization of second generation wireless digital systems. The digitization allows the use of Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA) techniques, which present significant advantages in comparison to the Frequency Division Multiple Access (FDMA) technique, which is used in analog systems.

With the TDMA technique, the wireless channel is divided in time slots, which can be dynamically apportioned to each user, depending on the user current needs. This leads to better channel utilization. With the CDMA technique, one frequency can be simultaneously used by many users, and this is achieved because the different signals are separated by using different modulation codes. Other advantages of the TDMA and CDMA techniques are [9]:

- smoother coexistence and communication with the digital wired networks
- capability of voice and data integration
- capability of future channel capacity increase, with the development of voice coders

- lower required transmission power (thus bigger independence duration)
- smaller system complexity.

*IV. Combination of Wireless Networks and ATM Technology*

The combination of wireless networks and ATM technology aims at combining the advantages of wireless communication and the users freedom of movement, with the capability of serving different traffic types and the guarantee of QoS that ATM provides. This combination is not easy, because ATM has been designed for environments very different than those of wireless networks. More specifically, ATM assumes:

- stable users
- ample and constant bandwidth, which can be dynamically apportioned, depending on the current connection needs
- full duplex and point-to-point transmission
- very good transmission quality, and for this reason the incorporated detection and error correction techniques are very limited
- small bandwidth aggravation from the physical layer (physical overhead).

On the contrary, in the wireless environment

- user movement is considered certain
- the wireless channel bandwidth is small and variable, depending on the channel quality
- the transmission is usually half-duplex, because of lack of available frequencies, and it is point-to-multipoint, so there is the need for a more specific routing

- the transmission quality is bad, and for this reason well developed detection and error correction techniques are required
- the aggravation from the physical layer is large, mainly due to the synchronization delay between the transmitter and the receiver [10].

From all the above, it is clear that it is necessary to design specific techniques for the wireless segments of an ATM path, in order to guarantee QoS comparable to that of wired ATM connections. There are three basic requirements which must be fulfilled in a wireless ATM system:

1) efficient support of all ATM classes in the wireless channel (this is achieved with the use of an efficient multiple access control protocol),
2) limitation of the consequences of the relatively high error rate (this is achieved with the use of an error control mechanism),
3) avoidance of large cell dropping during user handover (several proposals exist for an efficient procedure which could achieve this, e.g. [11,12,13]).

### 1.2. Our work

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. This will be achieved via major advances in wireless digital communications technology and due to the continuous proliferation of small, portable and inexpensive computing devices.

Current and future wireless networks are and will be based on the cellular concept [14]. Rather than using one base station (and high transmission power) to cover a large service area (e.g., a city), multiple base stations are used to partition the area into *cells*[1]. Since identical carrier radio frequencies can be used in cells that are far enough apart to eliminate interference, multiple mobile-base pairs can use the same frequency simultaneously (frequency reuse). A procedure known as handover is used to maintain continuous coverage over the service area[2]. It is interesting to observe that the cellular concept is a staple of the broadcast radio and television industries, thus the significance of the work in [15] is that frequency reuse is accomplished over much smaller geographic areas. The ability to maximize system capacity through frequency reuse, along with advances in microelectronics technology, make mobile telecommunication attractive to both the service provider and the consumer.

First generation cellular systems have been designed for high power and high mobility (vehicular) voice users. They employ analog FM radio technology and frequency division multiplexing to provide channel access. The first generation North American cellular system, Advanced Mobile Phone Service (AMPS), was deployed in 1983 and is still in use today. Other first generation systems based on different standards were used in European countries (e.g., UK, France and Germany each had their own) and Japan has at least one.

Second generation cellular systems exploit high power (0.5-5 W) digital radio technologies to increase system capacity, while continuing to focus primarily on vehicular voice users. The second generation Pan-European system, Global System for

---

[1] The information unit named ″cell″ in Part I of the introduction is from now on going to be named ″packet″,to avoid confusion between the two terms.

[2] Whenever the mobile crosses cell boundaries, it must change its frequency to correspond with that of the new base station. A recent survey of handover research can be found in [30].

Mobile Communications (GSM), is based on TDMA technology [16]. GSM was deployed in 1993 and it has been adopted by over 50 countries, about half of which are non-European. Additionally, one will see reference to Digital Cellular System 1800 (DCS 1800) which is GSM operating at 1.8 GHz [17]. Several standards exist for second generation systems in North America: IS-54/IS-36 are based on TDMA [18]; and, IS-95 is based on CDMA, a direct sequenced spread spectrum technology [19,20]. IS-54 is currently being introduced into the largest cellular markets and IS-95 has been deployed in several parts of the USA. Finally, we note that Japan has deployed its own second generation cellular systems, Personal Digital Cellular (PDC), based on TDMA [21].

Ideally, using digital technology, third generation wireless networks will integrate the needs of pedestrian users (low power and low mobility) with the needs of the vehicular users, and provide ″seamless″ integration of various types of service (e.g., cordless phones, paging systems and wireless data networks) as well. At lest two major efforts are underway to develop mobile telecommunication standards for the emerging third generation systems. The universal mobile telecommunication system (UMTS) is being developed as a Pan-European standard, and the future public land mobile telecommunications (FPLMTS) is being developed as a global standard by the International Telecommunications Union (ITU) [22,23].

In a third generation cellular system, the system capacity can be increased by:

a) using a cellular structure with a cell size as small as possible (microcells and picocells) 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), and satisfying the diverse and usually contradictory QoS requirements of each traffic class (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.

Most multiple access protocols are based on one of the known multiple access techniques, i.e., FDMA, CDMA or TDMA. In wireless ATM networks, the lack of available frequencies and the demand for dynamic bandwidth reservation, especially for VBR connections, make the use of FDMA inefficient. On the other hand, CDMA limits the maximum transmission rate of a connection to a relatively low value, which is a problem for broadband applications (requiring transmission rates > 2 Mbps). Thus, most protocols in this field use a dynamic TDMA scheme [24], due to its capability to adjust to the needs of each connection, reserving more or less time slots in each case.

In this work, we design and evaluate multiple access based on TDMA, schemes which multiplex 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 outdoor and indoor microcellular (picocellular) environments.

Within the microcell (picocell), spatially dispersed source terminals share a radio channel that connects them to a fixed base 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 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.

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. Thus, packets must be delivered within specified maximum delays. 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 [25], 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%) [26]. On the other hand, data applications are more tolerant of delays, but 100% delivery of correct packets is often required (e.g. in the case of a file transfer) [27].

This work is organized in three chapters. Chapter 2 is divided in two parts. The first part presents a scheduling mechanism for integrating video, voice and data packet traffic without video contention (this mechanism is compared to another efficient scheme recently proposed), and the second presents our MAC scheme for the integration of video, voice and data packet traffic under a completely realistic scenario. Chapter 3 presents our final conclusions, a summary of the major contributions of this work, and suggestions for possible future work on the subject.

# Chapter 2
# Scheduling in Next Generation Wireless Networks: Preemption vs. Contention and Noise in a MAC Scheme for Multimedia Integration

## 2.1. Introduction

Emerging wireless networks aim to satisfy the goal of incorporating and supporting a large variety of multimedia applications. An efficient MAC protocol can exploit the variations in access and service required by disparate sources and therefore can reduce system costs by maximizing system capacity, while integrating different classes of traffic.

In this work, we design a MAC scheme which supports multimedia traffic access to the wireless medium and compare it with the MAC protocol proposed in [28], which showed high throughput results. The basic differences of our scheme with [28] concern:

1. The base station scheduling policy.
2. The wireless channel conditions (we study the performance of our scheme under two different channel error models, while [28] considered an ideal channel without errors).
3. The way video users acquire access to the channel.

Our MAC scheme is designed for the uplink wireless channel, where such a scheme is required in order to resolve the source terminals contention for channel access. The scheme multiplexes voice traffic at the vocal activity (talkspurt) level to efficiently integrate voice (Constant Bit Rate, CBR On/Off Traffic), MPEG-4 videoconference traffic (Variable Bit Rate, VBR) and bursty e-mail data traffic in high capacity picocellular systems.

A presentation of the traffic models used for our study follows, and then the paper is divided in two parts: Part I presents a modification of the protocol presented in [28], by introducing errors in the wireless channel, as well as a data preemption policy. It is shown that the data preemption policy, regardless of the bursty channel errors, provides impressively better channel throughput results than those in [28]. Part II presents our fully developed MAC scheme, which has three more significant differences (additionally to the presence of errors and the data preemption policy) compared to [28], in order to design a protocol that is both efficient and realistic.

## 2.2. Traffic Models

*A. Voice and Data Traffic Models*

Our primary voice traffic model assumptions are the following:

a. Voice terminals are equipped with a voice activity detector [29]. 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. 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 (as it will be explained in Section I.B), while the average duration of the talkspurt period is 1.41 seconds and of the silence period 1.78 seconds.

c. The voice delay limit is equal to 40 ms and the allowed voice packet dropping probability is set to 0.01. [30]

We adopt the data traffic model based on statistics collected on email usage from the Finish University and Research Network (FUNET) [32]. The probability distribution function f(x) for the length of the data messages of this model was found to be well approximated by the *Cauchy* (0.8, 1) distribution. The message inter-arrival time distribution is exponential, and the average data message length is 80 packets.

*B. MPEG-4 streams*

In our study, we use the trace statistics of actual MPEG-4 streams from [33, 36]. The video streams have been extracted and analyzed from a camera showing the events happening within an office. We have used the high quality version of the movie, which has a mean bit rate of 400 Kbps, a peak rate of 2 Mbps, and a standard deviation of the bit rate equal to 434 Kbps.

All the streams are encoded at 25 video frames per second, which means that a new video frame is generated every 40 msecs. In our study we have made the assumption that the maximum transmission delay for video packets is 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 allowed video packet dropping probability is set to 0.0001 [30].

# Part I

## A. Channel Structure, Actions of Terminals and Base Station Scheduling

As mentioned in the Introduction, this Part of our work presents the necessary modifications in the MAC scheme presented in [28], in order to study a more realistic scenario (presence of channel errors) together with the use of a data preemption policy.

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 Figure 2).

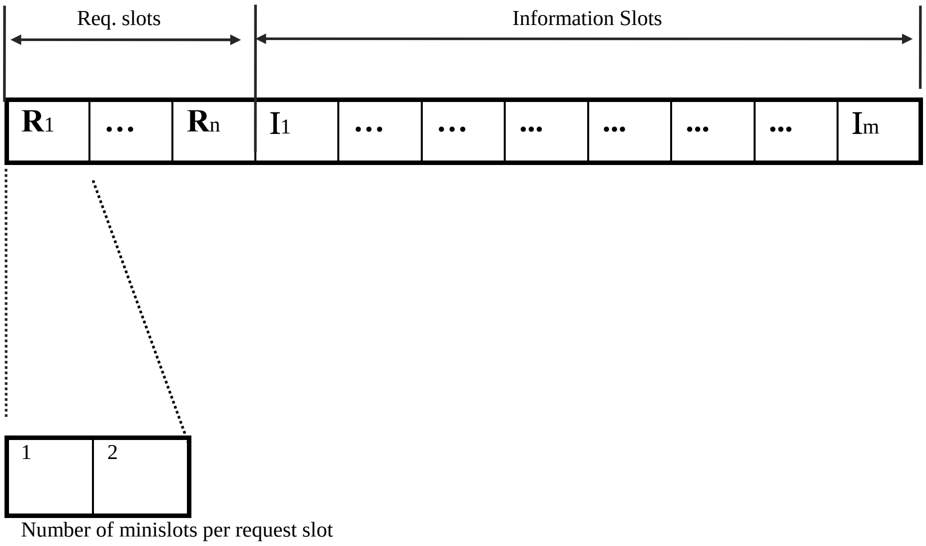


**Figure 2. Frame structure for the 20 Mbps channel, frame duration 12 ms.**

Within an information interval, each slot accommodates exactly one, fixed length, packet that contains voice, video or data information and a header.

In [28], no request slots were used for video terminals. They were considered to "live" permanently in the system and envoy their slot requests to the Base Station (BS) by transmitting them within the header of the first packet of their current video frame, and the reason for that choice was that videoconference users were assumed to transmit low activity video content with a few small changes in bit rate over time; therefore, request bandwidth dedicated to video users would be wasted, as they would seldom use it to notify the BS for a change in their bit rate transmissions. In this Part of our work, we use the same assumption.

Furthermore, we assume, as in [28], that the number of request slots per channel frame is variable, depending on the number of the video terminals.

Voice terminals are given highest priority to transmit their requests to the BS. When all contending voice terminals have transmitted their requests, the data request transmission follows. The concept of reserving a minimum bandwidth for terminals to make channel reservations helps to keep the user access delay within relatively low limits and gives clearly better performance than the PRMA [29] and quite a few PRMA-like algorithms, such as DPRMA [30], 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.

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. Since we assume that all of the voice source state transitions occur at the frame boundaries, we place all request intervals at the beginning of the frame, in order to minimize the voice packet access delay.

In our study, we adopt the *two-cell stack* reservation random access algorithm [31] for use by both video and voice terminals, due to its operational simplicity, stability and relatively high throughput when compared to the PRMA (Aloha-based) [29] and PRMA-like algorithms [30].

The *two-cell stack* blocked access collision resolution algorithm [31] 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.

The BS allocates channel resources <u>at the end of the corresponding request interval</u>, and follows a different allocation policy for each of the three types of 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, the BS grants to the video users as many of the slots they requested as possible. The BS allocates <u>the earliest available</u> 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 80 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, we introduce the following mechanism: the BS in our scheme allocates a slot to each requesting voice terminal with a probability* ***p**** *(the value of which is determined via simulation, for various video traffic loads, and has been set to 10% in our work).* The requests of voice terminals which ″fail ″ to acquire a slot, based on the above BS slot allocation policy, remain queued. Voice terminals with queued requests must continuously monitor the base-to-mobile channel. Reserved slots are deallocated immediately, i.e., 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. Within each priority class, the queuing discipline is assumed to be First Come First Served (FCFS).

The above comments on why we need to use **p***, stand for the explanation of our “defensive” slot allocation policy to data users, as well. As video traffic is quite bursty, an effort of allocating to a data terminal as many slots as it needs in one frame, in order to transmit its whole message, would again result in the deterioration of the video QoS.

The above described policy was used in [28]. The BS scheduling policy in this first Part of our work has one significant difference from the respective one in [28]: in order to provide better service to video and voice requests, the BS “preempts” data reservations; whenever new video and voice requests are received and every slot within the frame is reserved, the BS attempts to service the video and voice (in this priority order) requests by canceling the appropriate number of reservations belonging to data terminals (if any). When data reservations are canceled, the BS notifies the affected data terminal and places an appropriate request at the front of the data request queue.

## B. Error Models

In the design of the MAC scheme in [28], the channel was assumed to be without errors. This, in practice, is impossible for a wireless channel; therefore we introduce two different error models in our system and study the system performance in the presence of each model.

The first error model adopted is a simple model of uniformly distributed errors. 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 $9*10^{-5}$; this value has been chosen in order to test the “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 video packet dropping upper limit, the strict Quality of Service (QoS) requirement of video users would certainly be violated.

The second error model is the widely studied Gilbert-Elliot [34, 35] model, where the channel switches between a “good state”, where no errors occur, and a “bad state”, where errors occur with probability equal to 1. Errors in the wireless channel typically occur over relatively short bursts and are highly correlated in successive slots, but uncorrelated over long time windows. Therefore, errors can be well represented by the 2-state Markov model from [34, 35]. In order to make comparisons between the two error models (uniform and Gilbert-Eliot), we kept the probability that the channel is in a bad state equal to $9*10^{-5}$. The probability $P_r$ (bad-good), i.e., the probability that the channel switches from the bad to the good state, was set to $P_r$ (bad-good) = 0.33 (i.e., the average duration of error bursts is 3 slots).

## C. System Parameters

The system parameters are taken from [28]. 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. Assuming that the speech codec rate is 32 Kbps and that the packet length is equal to the size of an ATM cell, 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 higher than 18 Mbps since we encapsulate the ATM headers and the reservation slots. The high channel rate of 20 Mbps leads to a slot duration of only 0.021 msecs.

The number of the request slots in the system is kept in this Part of our work equal to the number of request slots in [28], i.e., it varies from 5 to 30 (10 to 60 request minislots), as shown in Table 1, depending on the number of video sources admitted into the system.

| **Number of Video Sources** | 0 | 1-6 | 7-12 | 13-15 | 16-20 |
|---|---|---|---|---|---|
| **R slots** | 30 | 25 | 20 | 12 | 5 |

**Table 1. Number of request slots depending on the video load.**

## D. Simulation Results and Discussion

An extensive simulation study was carried out. Each run simulated one hour of actual network activity. All the results shown correspond to averages of 10 independent runs (Monte Carlo method).

Tables 2 and 3 present the simulation results for the protocol designed in [28] (referred to as "original" in our Tables and Figures), as well as for the modified protocol with preemption but without errors (referred to as "pree" in our Tables and Figures) and for the modified protocol with preemption and channel errors (referred to as "U" when the uniform error model is adopted and "G" when the Gilbert-Elliot error model is adopted, in our Tables and Figures).

It is evident from both Tables that, for all cases of traffic loads studied, the modified protocol *clearly outperforms the protocol of [28] in terms of channel throughput, although [28] did not consider channel errors*. It should be noted that our protocol achieves the worst throughput results when the uniform error model is adopted, and not the burstier Gilbert-Elliot model as one could expect. The reason for this result is that error bursts in the Gilbert-Elliot model are followed by long intervals of errorless transmissions, which are very much needed for the system to cope with the high error rate and the burstiness of the model.

Another comment that needs to be made regarding the results of the modified MAC scheme is that the very significant increase in the number of voice terminals (and hence, of the channel throughput) in comparison to [28] is owed to the use of data preemption; therefore, the average data message delay is larger in the modified MAC scheme, for all cases of traffic loads examined. Still, our results have shown that this increase in data message delay is in most cases of the order of 1 second (the data message delay is close to 1 second in most cases of traffic load in [28], and close to 2 seconds in most cases of traffic load in the modified MAC scheme) and it seldom reaches 4-5 seconds. Given that the data traffic examined is email, this increase in data message delay is completely tolerable.

| R-Slots | λ (mess./frame) | Video Terminals | Voice Terminals (Original) | Channel Throughput (%) (Original) | Voice Terminals (Pree) | Channel Throughput (%) (Pree) |
|---|---|---|---|---|---|---|
| 5 | 0.5 | 20 | 0 | 51.24 | 98 | 57.82 |
| 12 | 0.5 | 15 | 288 | 61.85 | 386 | 68.62 |
| 12 | 1.5 | 15 | 94 | 61.4 | 385 | 82.84 |
| 20 | 0.5 | 10 | 496 | 66.44 | 593 | 73.7 |
| 20 | 1.5 | 10 | 298 | 65.69 | 591 | 87.38 |
| 25 | 0.5 | 5 | 780 | 76.75 | 880 | 84.15 |
| 25 | 1.5 | 5 | 586 | 76.3 | 763 | 88.94 |
| 25 | 0.5 | 1 | 971 | 82.27 | 1039 | 86.9 |
| 25 | 1.5 | 1 | 771 | 81.37 | 870 | 88.19 |
| 30 | 1.5 | 0 | 938 | 91.72 | 982 | 94.34 |

**Table 2. Voice Capacity and Channel Throughput results for [28] and for the modified protocol with preemption.**

| R-Slots | λ (mess./frame) | Video Terminals | Voice Terminals (U) | Channel Throughput (%) (U) | Voice Terminals (G) | Channel Throughput (%) (G) |
|---|---|---|---|---|---|---|
| 5 | 0.5 | 20 | 7 | 51.33 | 33 | 53.32 |
| 12 | 0.5 | 15 | 304 | 62.96 | 311 | 63.36 |
| 12 | 1.5 | 15 | 299 | 76.54 | 311 | 77.53 |
| 20 | 0.5 | 10 | 557 | 70.99 | 561 | 71.32 |
| 20 | 1.5 | 10 | 557 | 84.9 | 561 | 85.55 |
| 25 | 0.5 | 5 | 853 | 82.31 | 857 | 82.43 |
| 25 | 1.5 | 5 | 762 | 88.84 | 763 | 88.95 |
| 25 | 0.5 | 1 | 1037 | 86.79 | 1038 | 86.92 |
| 25 | 1.5 | 1 | 869 | 88.05 | 870 | 88.19 |
| 30 | 1.5 | 0 | 982 | 94.34 | 982 | 94.34 |

**Table 3. Voice Capacity and Channel Throughput results for the modified protocol with preemption and channel errors.**

Figure 3 presents the video packet dropping probability versus the number of voice users, when 10 video terminals are present in the system and the email message arrival rate is equal to 0.5 messages/frame (this data message arrival rate, for the email data traffic model studied, is equivalent to a traffic load of 1.28 Mbps). The most important observations from the plot are the following:

a. The MAC scheme presented in [28] exhibits worse results than all three modified versions presented in this first Part of the paper, as the video packet dropping probability reaches the upper bound of 0.01% in [28] much earlier than in the other three versions.

b. As expected, the best voice capacity results are exhibited by the “preemption” version of the protocol (without errors), which achieves a voice capacity of 590 terminals before the video packet dropping probability surpasses its upper bound. Still, in the more realistic scenarios when errors are present in the wireless channel, both the U and G versions of the protocol exhibit voice capacity results better than [28] by about 70 voice terminals. As shown in Figure 3, video packet dropping in the U and G versions of the protocol start from 0.009% (which is expected, as this is the steady-state probability that the channel is in bad state) and very slowly increase towards 0.01%.

c. Again, as in the results presented in Tables 2-3, the adoption of the uniform error model leads to somewhat worse results than the Gilbert-Elliot model.

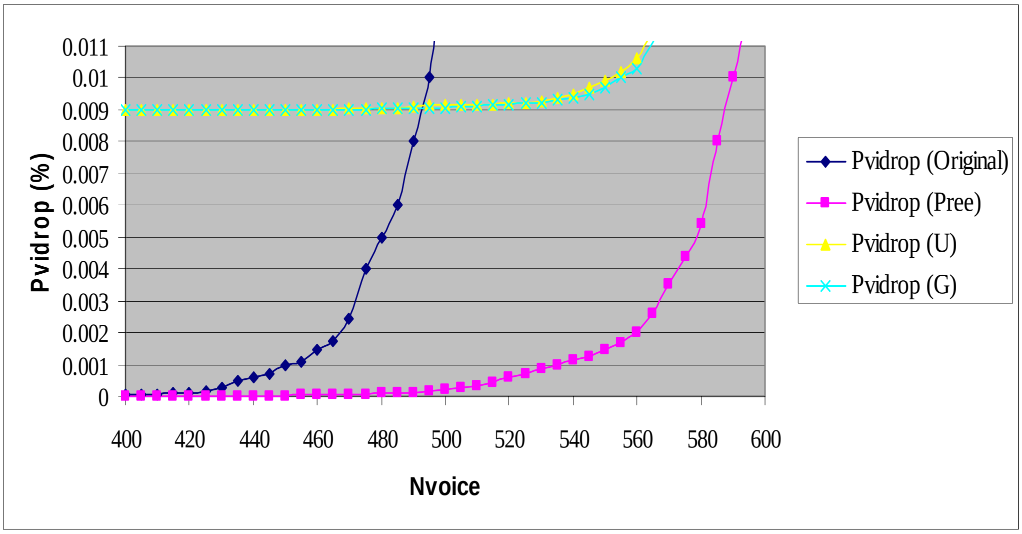


**Figure 3. Video packet dropping probability versus number of voice users (Nvid=10, λ=0.5 messages/frame).**

# Part II

## A. Channel Structure, Actions of Terminals and Base Station Scheduling

As mentioned in the Introduction, this Part of our work presents our fully developed MAC scheme, which has three significant differences with the one presented in [28], additionally to the channel errors and the data preemption policy which were introduced in Part I.

The first difference concerns the channel frame structure: the choice of having a variable number of request slots depending on the video load increases system complexity, even for a small number of "states" (as the one presented in Table 1), as the BS needs to change the channel frame structure and notify all terminals of the picocell of the change. Therefore, we chose to keep the number of request slots constant for all traffic loads in our MAC scheme.

The second difference concerns the way in which video users acquire access to the channel. As explained in Part I, no request slots were used for video terminals as the bit rate of videoconference users is assumed to experience few small changes over time. The above choice, although valid conceptually, is still a simplification, as even in videoconference movies fluctuations in the bit rate are not so rare that they can be considered negligible. Also, in [28], video users were considered to "live" permanently in the system and to have entered the system prior to the beginning of the simulation, therefore no mechanism was provided even for their first access to the wireless channel. Hence, in our MAC scheme video terminals also take part in the contention for channel access through the request slots. More specifically, video terminals transmit a request to the BS when the new Video Frame (VF) which they wish to transmit is larger than the previous one, i.e., when they need to acquire a larger percentage of channel resources. When the new VF is smaller in size than the previous one, the video terminal signals the BS for the deallocation of the slots it no longer needs. Therefore, in this work request slots can be shared by *all types of terminals* (video, voice and data) to transmit their requests to the BS.

Video terminals are given highest priority to transmit their requests to the BS and are followed in priority by voice terminals. When all contending video terminals have transmitted their requests, the voice request transmission follows, and the data request transmission is last.

The third difference concerns the BS scheduling policy. The choice of having a constant number of request slots in order to decrease the system complexity creates a severe scheduling problem for servicing both video and voice terminals: for a given data message arrival rate, as the number of video terminals in the system increases, the respective number of voice terminals which the system can support decreases significantly (and therefore, the total number of video and voice users decreases significantly); this decrease in voice capacity was combined in [28] (and in Part I of our work) with a subsequent decrease in the number of request slots, which in turn provided more information slots for the highly demanding (in bandwidth) video terminals to transmit their packets. When the number of request slots remains constant, as in this Part of our work, the increase in the number of video users leads again to a decrease in the number of total users in the system, but in this case some of the request slots remain unused, thus creating a problem of lack of information slots for servicing video users. Therefore, the BS scheduling policy needs to be enhanced with a mechanism which will provide video users with the extra bandwidth needed to satisfy their strict QoS requirements. We have implemented and studied such a mechanism, which dynamically solves the scheduling problem created by the constant number of request slots.

Our mechanism is based on the 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 (as is the case for data 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 and is not predefined (as were the numbers of request slots in [28]), therefore system complexity is kept low. The BS has a very rough estimation of the voice preemption start point depending on the number of video users (i.e., preemption starts at 60% of the frame when less than (or exactly) 10 video users are present in the system and at 40% of the frame when more than 10 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 seconds, 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 in packet dropping, which doesn't allow for time to be lost before video packet dropping is lower than its set upper bound.

## B. Error Models

The uniform and Gilbert-Elliot error models were adopted again, as in Part I. The differences with the models used in Part I concern the probability that the channel is in a bad state, and they will be explained in Section II.D.

## C. System Parameters

The number of the request slots in our system is constant, as mentioned above, and equal to 25 (after extensive simulations with various numbers of request slots, this number of request slots provided the best throughput results when our new BS scheduling policy was used). The other system parameters are the same with the ones in Part I.

## D. Simulation Results and Discussion

Once more, an extensive simulation study was carried out. Each run simulated one hour of actual network activity. All the results shown correspond to averages of 10 independent runs (Monte Carlo method).

Our choices of keeping the number of request slots constant and letting video terminals contend for request slots create a "burden" for the system in servicing video traffic, as explained earlier. The implemented voice preemption mechanism helps the system to cope with this "burden", but as shown from our simulations, the system can not support any video load for a channel error probability larger than $6*10^{-5}$, for neither of the two channel error models. Therefore the channel error probability is chosen equal to $6*10^{-5}$, and for reasons of comparison with the results in Part I, we have kept the probability of transition from bad to good channel state, in the Gilbert-Elliot model, equal to $P_r$ (bad-good)= 0.33.

Tables 4 and 5 present once more the simulation results for the protocol designed in [28], along with the respective results for our MAC scheme with voice preemption but without errors (referred to as "preevo" in our Tables and Figures), and with both preemption and channel errors (referred to as "Upreevo" when the uniform error model is adopted and "Gpreevo" when the Gilbert-Elliot error model is adopted, in our Tables and Figures). The column titled "start point for preevo" denotes the point in the frame where voice preemption has to start, for the specific traffic loads. The points shown in Tables 4-5 have been found through simulation, based on the mechanism described in Section II.A. For the case of one video terminal present in the system, our simulations have shown that no voice preemption is necessary, as of course is the case when no video terminals are present in the system (therefore, no values for the "start point for preevo" are given in the last three rows of Tables 4-5) .

As expected from the much more realistic channel frame structure and mechanism for channel access for video terminals, our scheme achieves worse voice capacity and throughput results than the modified MAC presented in Part I (which did not consider video contention and had a variable number of request slots). Still, it is clear from both Tables that, for almost all of the cases of traffic loads studied, our MAC scheme *outperforms the scheme of [28] in terms of channel throughput, although [28] did not consider neither channel errors nor video contention, and proposed a system with increased complexity*.

The only case where our scheme does not outperform [28] is the case where 20 video terminals are present in the system; as shown in the first row of Table 5, the system is unable to support this video load. The reason for this is that the 25 request slots are too many (i.e., the remaining 566-25=541 information slots are too few) for the 20 video terminals to transmit their data; the strict bound for the video packet dropping probability is violated. Nevertheless, we have attempted to simulate the same traffic load situation with less request slots and the MAC scheme again exhibited better results than [28] for any number R of request slots, if $5 \leq R \leq 20$. Still, as the choice of keeping the number of request slots constant in our scheme is one of the basic differences with [28], we present the specific load in Table 5 as a load that the system can not support.

As in Part I:

a. our scheme achieves the worst throughput results when the uniform error model is adopted.
b. the data message delay in our scheme is in most cases larger than the scheme in [28] by about 1 second, which is a totally acceptable delay for email traffic.

It should be noted that the small “fluctuations” in data message delay, which in few cases in Part I of this work resulted in an increase of 4-5 seconds for the modified

MAC, are not observed in our results for the fully developed MAC scheme. The reason is the abundance of request slots (25 request slots are more than enough in most cases of traffic loads examined), and therefore the data *access* delay to the channel (time until the successful transmission of the request packet) is minimized, resulting in a respective decrease of the data *message* delay.

| Start Point for PreeVo | λ (mess./frame) | Video Terminals | Voice Terminals (Original) | Channel Throughput (%) (Original) | Voice Terminals (PreeVo) | Channel Throughput (%) (PreeVo) |
|---|---|---|---|---|---|---|
| 0.14 | 0.5 | 20 | 0 | 51.24 | 15 | 51.74 |
| 0.4 | 0.5 | 15 | 288 | 61.85 | 342 | 65.21 |
| 0.4 | 1.5 | 15 | 94 | 61.4 | 342 | 79.31 |
| 0.53 | 0.5 | 10 | 496 | 66.44 | 547 | 69.63 |
| 0.53 | 1.5 | 10 | 298 | 65.69 | 547 | 83.45 |
| 0.8 | 0.5 | 5 | 780 | 76.75 | 829 | 79.82 |
| 0.8 | 1.5 | 5 | 586 | 76.3 | 743 | 87.38 |
| - | 0.5 | 1 | 971 | 82.27 | 1020 | 85.29 |
| - | 1.5 | 1 | 771 | 81.37 | 869 | 88.05 |
| - | 1.5 | 0 | 938 | 91.72 | 982 | 94.34 |

**Table 4. Voice Capacity and Channel Throughput results for [28] and for our MAC scheme with voice preemption.**

| Start Point for PreeVo | λ (mess./frame) | Video Terminals | Voice Terminals (Upreevo) | Channel Throughput (%) (Upreevo) | Voice Terminals (Gpreevo) | Channel Throughput (%) (Gpreevo) |
|---|---|---|---|---|---|---|
| - | 0.5 | 20 | - | - | - | - |
| 0.4 | 0.5 | 15 | 302 | 62.45 | 306 | 62.81 |
| 0.4 | 1.5 | 15 | 300 | 76.37 | 304 | 76.69 |
| 0.53 | 0.5 | 10 | 497 | 66.32 | 518 | 67.77 |
| 0.53 | 1.5 | 10 | 496 | 80.31 | 518 | 81.82 |
| 0.8 | 0.5 | 5 | 829 | 79.82 | 829 | 79.82 |
| 0.8 | 1.5 | 5 | 742 | 87.26 | 743 | 87.34 |
| - | 0.5 | 1 | 1020 | 85.29 | 1020 | 85.29 |
| - | 1.5 | 1 | 868 | 88.01 | 869 | 88.07 |
| - | 1.5 | 0 | 981 | 94.32 | 982 | 94.34 |

**Table 5. Voice Capacity and Channel Throughput results for our MAC scheme with voice preemption and channel errors.**

Similarly to Figure 3, Figure 4 presents the video packet dropping probability in our MAC scheme versus the number of voice users, when 10 video terminals are present in the system and the email message arrival rate is equal to 0.5 messages/frame. The comments on the respective plot of Figure 3 stand for Figure 4 as well.

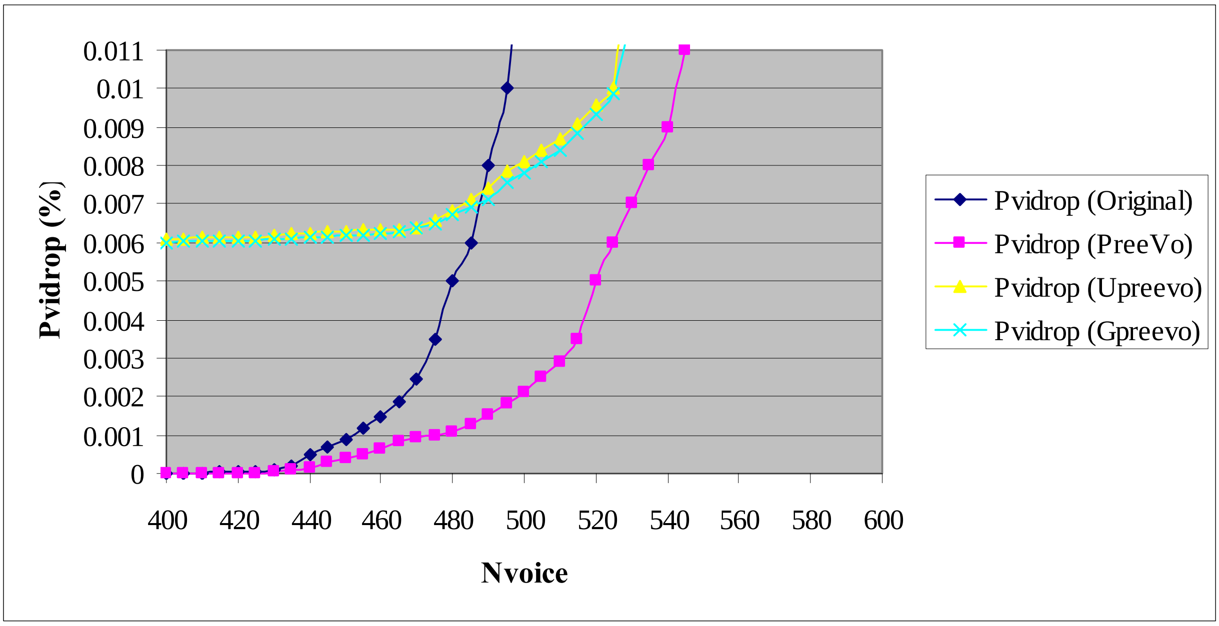


**Figure 4. Video packet dropping probability versus number of voice users (Nvid=10, λ=0.5 messages/frame).**

# Chapter 3

# Conclusions

In this work, we have considered the design and performance evaluation of multiple access schemes that multiplex voice traffic at the talkspurt level, to efficiently integrate voice, data and video traffic in indoor picocellular environments. The *design goals* included maximizing the system capacity and integrating voice, video and data traffic, while satisfying quality of service requirements such as voice packet dropping probability, mean voice access delay, video packet dropping probability, mean video access delay. These goals were complicated by the limited radio channel bandwidth and by the contradictory nature of all three different traffic types. Still, our results show that our schemes' efficiency is very satisfactory in all cases, and the result comparison with a previously proposed efficient MAC protocol shows that, regardless of the much more severe (and realistic) channel conditions examined in our work, our scheme clearly outperforms the previously proposed protocol in terms of channel throughput.

# REFERENCES


1. H. Armbruster, ″The flexibility of ATM: Supporting Future Multimedia and Mobile Communications″, *IEEE Communications Magazine*, Vol. 33, No. 2, Feb. 1995, pp. 76-84.
2. T. M. Chen and S. S. Liu, ″ATM Switching Systems″, 1995.
3. ATM Forum Technical Committee, ″Traffic Management Specification Version 4.0″, April 1996.
4. W. Stallings, ″ISDN and Broadband ISDN with Frame Relay and ATM″, Prentice Hall, 1995.
5. N. Passas, ″Multiple Access Control in Wireless ATM Networks″, *Ph.D. Dissertation, Dept. of Informatics, University of Athens*, Dec.1997.
6. K. Sriram, ″Methodologies for bandwidth allocation, transmission scheduling, and congestion avoidance in broadband ATM networks ″, *Computer Networks and ISDN Systems*, Vol. 26, No. 1, pp. 43-59, 1993.
7. O.T.W. Yu and V.C.M. Leung, ″Adaptive resource allocation for prioritized call admission over an ATM-based wireless PCN ″, *IEEE Journal on Selected Areas in Communications*, Vol. 15, No. 7, pp. 1208-1225, September 1997.
8. B. L. Mark and G. Ramamurthy, ″Joint source-channel control for real-time VBR over ATM via dynamic UPC renegotiation″, *IEEE Globecom ’96*, Vol. 3, pp. 1726-1731, London, UK, November 1996.
9. J. E. Padgett et al., ″Overview of Wireless Personal Communications″, *IEEE Communications Magazine*, Vol. 33, No. 1, pp. 28-41, January 1995.
10. F. Bauchot, ″MASCARA: A Wireless ATM MAC Protocol″, *Proceedings of the Wireless ATM Workshop*, Helsinki, Finland, September 1996.
11. H. Mitts et al., ″Lossless handover for wireless ATM″, *ACM Mobile networks and Applications Journal*, Vol. 1, No. 3, pp. 299-312, December 1996.
12. C.-K. Toh, ″A hybrid handover protocol for local area wireless ATM networks″, *ACM Mobile networks and Applications Journal*, Vol. 1, No. 3, pp. 313-334, December 1996.
13. A. Kaloxylos et al., ″Smart Buffering Technique for Lossless Hard Handover in Wireless ATM networks″, *Proceedings of the IEEE International Conference on Communications ‘96*, pp. 240-244, Dallas, TX, June 1996.
14. V. H. MacDonald, ″The Cellular Concept″, *Bell Sys. Tech. Journal*, Vol. 58, No. 1, pp. 15-41, January 1979.

15. G. P. Pollini, ″Trends in Handover Design″, *IEEE Communications Magazine*, pp. 82-90, May 1996.
16. M. Rahnema, ″Overview of the GSM System and Protocol Architecture″, *IEEE Communications Magazine*, pp. 92-100, April 1993.
17. P. A. Ramsdale, ″Personal Communications in the UK-Implementation of PCN using DCS 1800″, *International Journal of Wireless Information Networks*, Vol. 1, No. 1, pp. 29-36, January 1994.
18. TIA/EIA IS-54, ″Cellular System Dual-Mode Mobile Station-Base Station Compatibility Standard″, Telecommunications Industry Association, April 1992.
19. A. J. Viterbi, ″Spread Spectrum Communications- Myths and Realities″, *IEEE Communications Magazine*, pp. 11-18, May 1979.
20. R. Kohno, R. Meidan and L.B. Milstein, ″Spread Spectrum Access Methods for Wireless Communications″, *IEEE Communications Magazine*, pp. 58-67, January 1995.
21. K. Kinoshita, M. Kuramoto and N. Nakajima, ″Development of a TDMA Digital Cellular System Based on Japanese Standards″, *Proceedings of the 41st IEEE Vehicular Technology Conference (VTC '91)*, pp. 642-645.
22. C. C. Evci, ″RACE-UMTS for Third Generation Wireless Networks″, *Ann. Telecommun.*, Vol. 47, No. 7-8, pp. 267-281, July-August 1992.
23. R. Pandya, ″Emerging Mobile and Personal Communication Systems″, *IEEE Communications Magazine*, pp. 44-51, June 1995.
24. D. D. Falconer et al., ″Time Division Multiple Access Methods for Wireless Personal Communications″, *IEEE Communications Magazine*, Vol. 33, No. 1, pp. 50-57, January 1995.
25. B. Mukherjee, ″Integrated voice-data communication over high speed fiber optic networks″, *IEEE Computer Magazine*, Vol. 24, No. 2, pp. 49-58, 1991.
26. D. A. Dyson and Z. J. Haas, ″ A Dynamic Packet Reservation Multiple Access Scheme for Wireless ATM ″, *Proceedings of the IEEE MILCOM'97*.
27. 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.
28. S. Psychis, P. Koutsakis and M. Paterakis, "Multiplexing Live Video Streams & Voice with Data over a High Capacity Packet Switched Wireless Network", *in Proceedings of the 5th European Wireless Conference*, February 2004, Barcelona, Spain. Available online at: http://research.ac.upc.es/EW2004/papers/78.pdf

29. 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.

30. D.A. Dyson and Z. J. Haas, "A Dynamic Packet Reservation Multiple Access Scheme for Wireless ATM", *ACM / Baltzer MONET Journal*, Vol. 4, No.2, pp. 87-99, 1999.

31. 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, September 1989, pp. 1124-1130.

32. Q. Pang, A. Bigloo, et al.,"Service Scheduling for General Packet Radio Service Classes", in Proc. of the IEEE Wireless Communications and Networking Conference 1999 (WCNC '99), New Orleans, USA.

33. [Online] http://trace.eas.asu.edu/

34. E. N. Gilbert, "Capacity of a burst-noise channel", *Bell Systems Technical Journal*, Vol.39, 1960, pp. 1253-1265.

35. E. O. Elliot, "Estimates of error rates for codes on burst-noise channels", *Bell Systems Technical Journal,* Vol.42, 1963, pp. 1977-1997.

36. F. H. P. Fitzek, M. Reisslein, "MPEG-4 and H.263 Video Traces for Network Performance Evaluation", *IEEE Network*, Vol. 15, No. 6, 2001, pp. 40-54.