Journal of Interconnection Networks

PERFORMANCE ANALYSIS OF A MAC PROTOCOL FOR BROADBAND WIRELESS ATM NETWORKS WITH QUALITY OF SERVICE PROVISIONING

B. VAN HOUDT, C. BLONDIA
University of Antwerp
Department of Mathematics and Computer Science
Performance Analysis of Telecommunication Systems Research Group
Universiteitsplein, 1, B-2610 Antwerp - Belgium
{*vanhoudt,blondia*}*@uia.ua.ac.be*

O. CASALS, J. GARCÍA
Polytechnic University of Catalunya
Computer Architecture Department
C/ Jordi Girona, 1-3, E-08034 Barcelona - Spain
{*olga,jorge*}*@ac.upc.es*

Received 3-7-2000
Revised 2-5-2001

This paper presents a Medium Access Control (MAC) protocol for broadband wireless LANs based on the ATM transfer mode, together with the evaluation of its performance in terms of throughput and access delay. Important characteristics of the MAC protocol are the way information between the Mobile Stations (MSs) and the Base Station (BS) is exchanged and the algorithm used to allocate the bandwidth in order to support the service categories. A detailed analytical evaluation, both on cell level and packet level, leads to an assessment of the efficiency and the access delay of the system.

Keywords: Quality of Service, Medium Access Control, Wireless ATM

1. Introduction

The development of powerful high performance portable computers and other mobile devices such as palmtops, have motivated the increasing interest in wireless communication systems, in particular for LANs (e.g., in an office environment). This evolution has to be combined with the trend towards high capacity, service integration and Quality of Service (QoS) provisioning, currently supported in fixed networks by the Asynchronous Transfer Mode (ATM). A seamless connection between these wireless LANs and the fixed network requires the definition of an ATM based transport architecture for an integrated services wireless network.

In a wireless network, the broadcast nature of the radio channel requires the introduction of a Medium Access Control (MAC) layer, in order to coordinate the access to the shared radio channel. A MAC protocol should not only avoid collisions and distribute the available bandwidth in an efficient way, but it is also a key component in the support of QoS provisioning.

Consider a cellular network with a centralized architecture, i.e., the area covered by the wireless ATM network is subdivided into a set of geographically distinct cells each with a diameter of approximately 100m (slight overlaps are allowed to facilitate the handovers from one cell to a neighboring cell). Each cell contains a base station (BS) serving a finite set of mobile stations (MSs). This BS is connected to an ATM switch, which supports mobility, realizing seamless access to the wired network (see Figure 1). Two logically distinct communication channels (uplink and downlink) are used to support the information exchange between the BS and the MSs. ATM cells arriving at the BS are broadcasted downlink, while upstream ATM cells must share the radio medium. The BS controls the access to the shared radio channel (uplink) (i.e., the MAC has a centralized control architecture). The access technique is Time Division Multiple Access (TDMA) combined with Frequency Division Duplex (FDD) to separate the uplink and downlink channels.

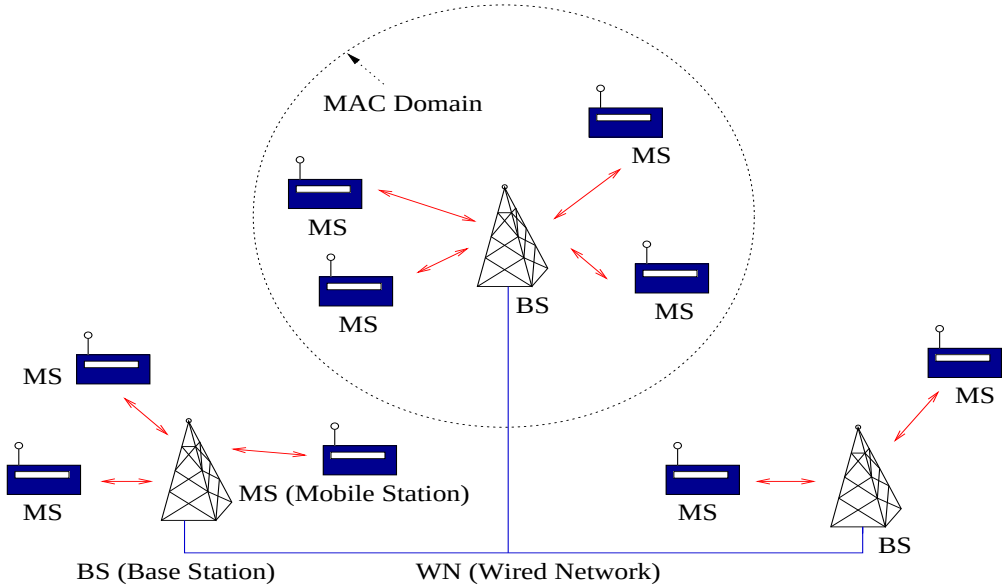

Figure 1: Reference configuration of the system

The MAC protocol discussed in this paper was first introduced in [12]. Partial performance evaluations have been conducted in [14,15], where we focused on the delay and throughput of the contention channel—that is, we studied the performance of the ISAP contention resolution algorithm described in Section 3.2. In this paper we consider traffic generated by higher layers in the MS and propose an analytical

model to compute the packet access delay caused by the proposed MAC protocol. We investigate the influence of the traffic characteristics on the efficiency and on the delay of upstream packets.

Sections 2 and 3 introduce the MAC protocol together with the Identifier Splitting Algorithm with Polling (ISAP). Section 4 shortly discusses the support of ABR congestion control. A short summary of the results obtained in [14,15] is presented in Section 5. Next, Section 6—the main contribution of the paper—proposes a queueing model to derive a number of performance measures. In particular, the model is used to illustrate the influence of the statistical parameters of the packet arrival process on the efficiency and delay of the MAC protocol. Conclusions are drawn in Section 7.

2. Support of ATM Service Categories

In this section we discuss how the ATM service categories, as defined by the ATM Forum, may be supported. To simplify the discussion, we first propose a simple system model.

2.1. *A Model for the Wireless ATM Access Network*

The aim of the MAC protocol is to allocate, in a fair and efficient way, the uplink bandwidth among the active MSs such that the QoS requirements of the various traffic streams are satisfied. From this point of view, the wireless access system can be considered as a two stage multiplexer, with distributed buffers (each MS has its own buffer) and a two level scheduler : a scheduler in the BS and a scheduler in each MS (see Figure 2).

The scheduler in the BS is responsible for allocating the bandwidth among the various active MSs. We assume that this scheduler operates at the level of mobile station and service category (e.g., CBR, ABR,...). Hence, the BS can allocate bandwidth to all the CBR connections (as a group) of a certain MS, but not to a particular VC. The selection of a particular VC is the responsibility of the scheduler in the MS. This scheduling mechanism in an MS may be a simple FCFS algorithm or a more complex per-VC algorithm (such as weighted fair queueing). When buffers are shared by multiple VCs, a buffer management mechanism may control the buffer occupancies of individual VCs by dropping cells selectively. These per-VC buffer management schemes are very important to support the GFR and UBR service category.

In the next subsection we derive the requirements on the BS scheduler, the MS scheduler and the MS buffer management scheme to support the various service categories.

2.2. *ATM Service Categories and their Requirements*

We consider three classes of ATM service categories. The first class contains the

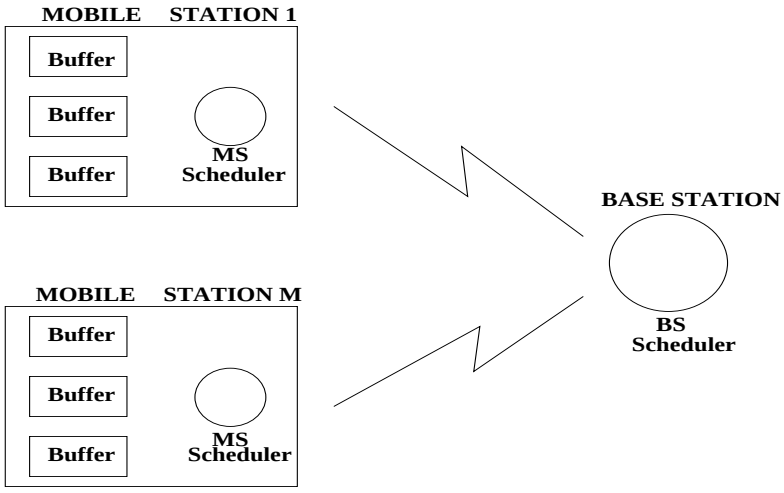

Figure 2: The WATM Access Network Model

services with real-time constraints, known as CBR and rt-VBR. They are also referred to as *stream* traffic. A second class deals with traffic that is able to adapt its rate to the available bandwidth. It is often referred to as *elastic* traffic, and contains the ABR and GFR service categories. Finally, the *best effort* traffic class covers the UBR service category. In what follows we discuss each service category in some more detail and derive the requirements the proposed MAC protocol has to fulfill in order to support these categories.

2.2.1. *CBR and rt-VBR Traffic*

For CBR and rt-VBR traffic, the network provides firm guarantees with respect to the cell loss, the cell delay and in case of CBR, the cell delay variation. It is well known that when multiplexing a number of CBR and/or VBR sources, a simple FCFS scheduling discipline satisfies the loss and delay requirements, provided that an appropriate CAC algorithm is used. This leads to the following guidelines for our MAC protocol.

(i) *Scheduling in the BS* : The bandwidth for CBR and rt-VBR traffic is allocated in the BS among the various MSs according to a simple FCFS discipline.

(ii) *Scheduling in the MS*: The above-mentioned principle also applies within an MS where we propose two queues: one for the VCs carrying CBR traffic and one for the VCs carrying rt-VBR traffic. Within each queue a simple FCFS scheduling mechanism is used.

(iii) *Buffer Management in the MS*: These service categories experience virtually no loss, and therefore a simple tail drop buffer management mechanism is sufficient.

2.2.2. *ABR Traffic*

ABR uses a rate-based end-to-end closed loop control mechanism which aims at dividing the available bandwidth fairly and efficiently among the active users. The allowed cell rate (ACR) is sent to the source as feedback using resource management (RM) cells. The ACR varies between a minimum cell rate (MCR) and a peak cell rate (PCR). For a source emitting cells conform to the ACR, the network guarantees the MCR and a low cell loss probability. We assume that the wireless ATM access system, as a network component with a multiplexing function, implements an ABR traffic management scheme (see Section 4). Such a scheme ensures that the available uplink bandwidth is distributed fairly between all ABR VCs.

(i) *Scheduling in the BS*: The bandwidth allocation between CBR/rt-VBR traffic and ABR traffic is based on a simple priority mechanism: bandwidth is allocated to the ABR service class when there is no bandwidth needed for CBR/rt-VBR traffic. The CAC algorithm is responsible for the guarantee of the minimum cell rate of the ABR traffic (i.e., no MCR guarantee is provided on a cell time scale). Since the access network implements an ABR traffic management scheme, the bandwidth available for ABR traffic can be allocated between the competing MSs according to a simple FCFS scheduling strategy.

(ii) *Scheduling in the MS*: For the same reason as in the BS, the MS allocates the bandwidth between the ABR VCs on a FCFS basis. This means that no per-VC queueing is required.

(iii) *Buffer Management in the MS*: Since for the ABR service category virtually no loss occurs, a simple tail drop buffer management mechanism is sufficient.

2.2.3. *GFR Traffic*

The GFR service provides minimum rate guarantees to ATM VCs at the frame (i.e., AAL5) level. Contrary to ABR, GFR does not provide any feedback to the sources. These sources are supposed to be responsive to congestion by adapting their transmission rate using a higher layer networking protocol like TCP. Traffic sent in excess of the guaranteed minimum rate should receive a fair share of the unused bandwidth.

(i) *Scheduling in the BS*: GFR traffic receives bandwidth after CBR/rt-VBR and ABR traffic has been served. In [7], Goyal et al. show that when an appropriate buffer management scheme is used, a FIFO buffer shared by several GFR VCs is sufficient to provide the guaranteed rate. Therefore, the bandwidth available for GFR traffic can be allocated between the competing MSs according to a simple FCFS scheduling strategy.

(ii) *Scheduling in the MS*: For the same reason as in the BS, the MS allocates the bandwidth between the VCs on a FCFS basis. This means that no per-VC queueing is required.

(iii) *Buffer Management in the MS*: In order to guarantee a per-VC minimum rate, the differential fair buffer allocation (DFBA) strategy, described in [7], is proposed as the buffer management scheme to be utilized in the MS. DFBA is designed to allocate buffer capacity fairly amongst competing VCs, while utilizing the network efficiently.

2.2.4. *UBR Traffic*

UBR is the best effort ATM service class. It utilizes the remaining bandwidth without offering any guarantees. This means that scheduling in both the BS and MS is achieved using a FCFS strategy. Each MS may implement an intelligent cell drop mechanism such as early packet discard (EPD), in order to improve the goodput. If a certain level of fairness has to be achieved, a fair buffer allocation drop policy (see [6]) may be implemented as buffer management scheme.

2.3. *Scheduling Strategies and Buffer Management Schemes: Summary*

Based on the above discussion, we propose the following scheduling strategies and buffer management schemes to provide support to the various service categories.

(i) *Scheduling in the BS*: The BS scheduler operates at the level of MSs and ATM service categories—that is, it does not distinguish different VCs belonging to the same MS and the same ATM service category. Among service categories, the bandwidth is allocated according to a non-preemptive priority system, where CBR/rt-VBR traffic has the highest priority, followed by ABR, GFR and finally UBR. Within a service category, the bandwidth is allocated among the active MSs using a FCFS strategy.

(ii) *Scheduling in the MS*: The MS scheduler is responsible for distributing the bandwidth allocated to each service category among the VCs carrying traffic of that service category. We propose, in each MS, five FIFO queues, one for each ATM service category. Hence, no per-VC queueing is required.

(iii) *Buffer Management in the MS*: The five queues in each MS utilize the following buffer management schemes. For the CBR, rt-VBR and ABR FIFO queue, a simple tail drop scheme is needed. GFR requires per-VC accounting to implement a selective cell drop mechanism such as Differential Fair Buffer Allocation. Finally for UBR, the Early Packet Discard mechanism is proposed.

3. The MAC Protocol

In this section we give a detailed description of the proposed MAC protocol. First, we discuss how the information regarding the requested and allocated bandwidth is exchanged between an MS and the BS, together with the uplink and downlink frame structure. Next, we describe, based on the discussion in the previous section, the bandwidth allocation algorithm.

3.1. *Information Exchange Between MS and BS Regarding Bandwidth Allocation*

In this section we describe the mechanisms used to exchange information regarding the requested and allocated bandwidth between an MS and the BS. Assuming a MAC protocol with a centralized controller located in the BS, each MS must be able to inform the BS about its bandwidth needs and the BS should be able to inform the MSs about the received bandwidth. This information exchange is based on a request/permit mechanism.

3.1.1. *Permits*

An MS is allowed to use the uplink channel, that is, to send an ATM cell, whenever it receives a *permit* from the BS. A permit has a length of 3 bytes and contains the following information:

(i) the address of the permit's destination MS (1 byte)

(ii) the service category of the connection receiving the permit: CBR, rt-VBR, ABR, GFR or UBR (3 bits)

(iii) an indication of the instant the MS can send an upstream cell (i.e., the sequence number of the slot in the next upstream frame that may be used to send the upstream cell) (13 bits)

3.1.2. *Requests*

The MS declares its bandwidth needs to the BS by means of requests. There are different ways for an MS to send requests to the BS, depending on the ATM service category that requests bandwidth. A request consists of 8 bytes and contains the following information:

(i) the address of the MS that is issuing the request (1 byte)

(ii) the number of cells that are waiting in the respective queues for each service category (VBR, ABR, GFR and UBR). This field requires 4 times 14 bits.

There are two different ways to send requests:

(i) Piggybacked with upstream cells : when an MS is allowed to transmit a cell, a request is added to that upstream cell by means of piggybacking.

(ii) Using the contention resolution protocol : specific time intervals—referred to as contention slots—are used to allow MSs, that are unable to use piggybacking, to transmit a request to the BS.

Depending on the service category, a combination of these mechanisms is used to declare the bandwidth needs of an MS.

Request Mechanism for CBR Traffic: Due to the regular arrival instants of cells in the MS of a CBR connection, and in order to reduce the overhead introduced by the request mechanism, no requests are sent for CBR traffic. Instead, the Bandwidth Allocation Algorithm (see Section 3.4) generates permits at regular time instants. Hence, if an MS has a single CBR connection, these permits are generated according to the Peak Emission Interval agreed upon at call setup (and maintained in a table in the BS). In case of multiple CBR connections per MS, the permits are generated according to the sum of the Peak Emission Intervals of these CBR connections.

Request Mechanism for rt-VBR, ABR, GFR and UBR Traffic: Due to the variability of the cell rate, we can no longer use the above-mentioned scheme. In principle, a piggybacking scheme is proposed for this type of services as this introduces a minimal overhead. This means that an MS can add a request to each upstream cell it is allowed to send. However, this scheme fails in case the last upstream cell leaves an empty MS transmission buffer behind, while the connection is still active (i.e., it will generate a cell in the future). In particular the first cell of a new burst needs a mechanism to inform the BS about its presence. For this we propose a combination of a contention resolution algorithm and a polling scheme, called the *Identifier Splitting Algorithm combined with Polling (ISAP)* introduced in the next section.

3.2. *The Contention Resolution Scheme for Sending Requests*

The Identifier Splitting Algorithm (ISA) proposed by Petras in [10], is based on the well known tree algorithm [2]. A contention cycle (CC) is defined as a number of consecutive upstream frames during which the contention is resolved for all requests that want to make use of this scheme at the beginning of the cycle. The system is gated in the sense that requests that are generated during a CC and that intend to use the contention resolution scheme, have to wait for participation until the start of the next CC.

In the first frame of a cycle, a number of contention minislots are available (e.g., one slot consist of 4 minislots; see Section 3.3, Frame Structure) which can be used for contention resolution (we state that we start at level 2 of the tree becuase we have 2^2 minislots available). An MS selects a minislot according to its MAC address—that is, the first two bits of its MAC address determine which of the 4 minislot it will use—and transmits a request in this minislot. The BS checks which transmissions

have been successful and informs the MSs that were involved in the scheme in the next downstream frame using a feedback field (see Section 3.3, Frame Structure). Two situations are possible:

(i) An MS sending in minislot k, $1 \leq k \leq 4$, was successful (i.e., no collisions occurred). In this case the MS will eventually be granted a permit by the BS.

(ii) An MS sending in minislot k, $1 \leq k \leq 4$, was not successful, i.e., a collision with one or more MSs occurred. If there where l minislots $0 \leq l \leq 4$, holding a collision, then the next level (level 3) of the CC provides $2 \times l$ minislots for contention resolution and the involved MSs apply the same scheme again, each time using the next bit of the MAC address to decide which minislot (of the two minislots used to resolve the collision) is used.

This process of generating two minislots in the $i+1$-th level for each minislot of level i in which a collision occurred, is repeated level after level, each time using the next bit of the MAC address in case of a collision. Thus, during the i-th level of a CC, two MSs can only collide if their MAC addresses have the same i first bits. Therefore, provided that the address that uniquely identifies an MS is n bits long, all collisions are always resolved at level n. Also, notice that for every level i the number of minislots equals twice the number of collisions of the previous level (level $i-1$). To clarify all this, Figure 3 shows an example of a CC with 6 participants. In this figure **CO** refers to a collision, **SU** to a success and **EM** to an empty minislot. The MAC addresses of the successful MSs are added to the corresponding slot. Notice that the first two levels (0 and 1) in Figure 3 are only there to clarify the tree structure, but in the actual algorithm they do not exist (because we start with 4 slots, that is, at level 2).

In general, every level of the tree corresponds to a single frame, except when the number of minislots at some level i is larger than some predefined value L. This parameter L defines the maximum number of minislots that we allow in a single frame. Thus, if a certain level of the tree requires $x = mL + j$ minislots, with $1 \leq j \leq L$, then $m + 1$ frames are required to support this level.

One of the advantages of ISA is that as the scheme is being resolved, the BS obtains more and more knowledge about the address space of the MSs that are still competing. For example, if the BS notices that the tree at level i (recall that the top of the tree containing 4 minislots is referred to as level 2) contains k collisions and the MAC-addresses are n bits long, then the BS concludes that the remaining competing MSs can only have $k2^{n-i}$ possible addresses. This follows from the fact that each slot at level i corresponds to 2^{n-i} addresses. This knowledge can be used by the BS in order to improve the performance of the ISA scheme as follows. When the size of the remaining MAC-address space R_p becomes smaller than some predefined value, say N_p, the contention resolution algorithm is no longer applied. Instead R_p minislots are used to poll the MSs that were still competing (it is possible that a number of frames are required to do so, this depends on the relationship between

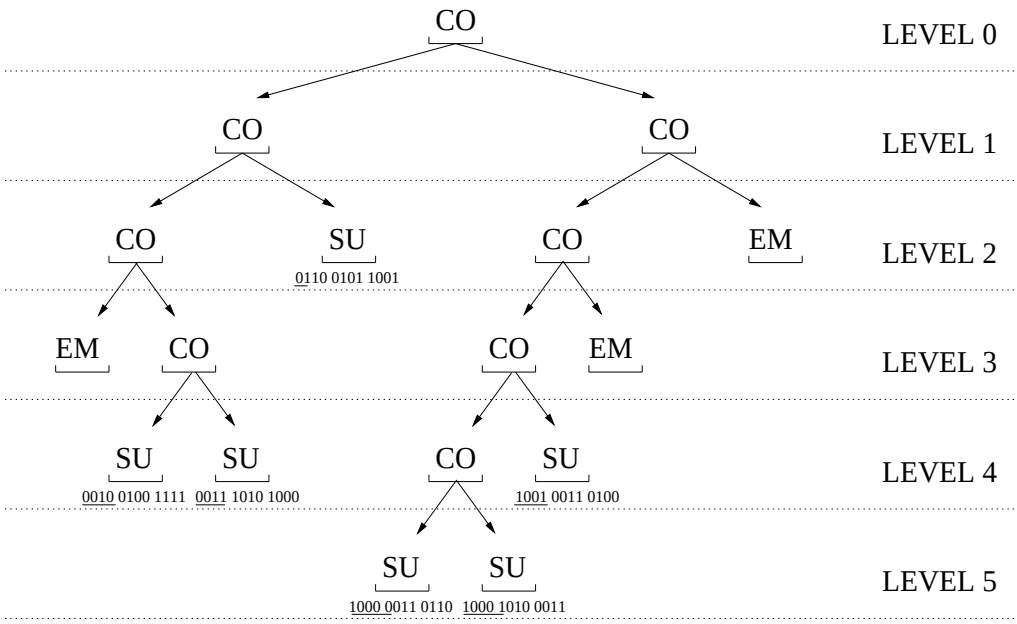

Figure 3: Demonstrating ISA

R_p and L). Based on its MAC address, each MS that is involved in the polling procedure knows exactly which minislot it is allowed to use. We refer to this variant of the ISA protocol as *ISA combined with polling (ISAP)*.

The ISAP scheme can be further optimized by considering a different (higher) starting level. Instead of starting with just 4 minislots (level 2), we might provide $8, 16$ or more minislots for the first attempt. Let the starting level be fixed at a predefined value $S_l \geq 2$. It has been shown in [14], that a higher starting level has a positive impact on the access delay, but may lead to poor throughput results, in particular for low load conditions. To solve this problem we proposed the following dynamic scheme [14], where the starting level depends on the length of the previous contention cycle (CC). Suppose that at some point in time the starting level equals S_l and T is the length of this CC. Then, the new starting level S_l' obeys the following equation

$$S_l' = \begin{cases} \max(S_l - 1, S_{min}) & T \leq B_l \\ S_l & B_l < T < B_m \\ \min(S_l + 1, S_{max}) & T \geq B_m \end{cases} \tag{3.1}$$

where B_l and B_m are two predefined values. Remark that in the above-mentioned equation the system load ρ is not needed, as this value is hard to measure or predict in real systems.

3.3. *Frame Structure*

In this section we give a detailed description of the uplink and downlink frame

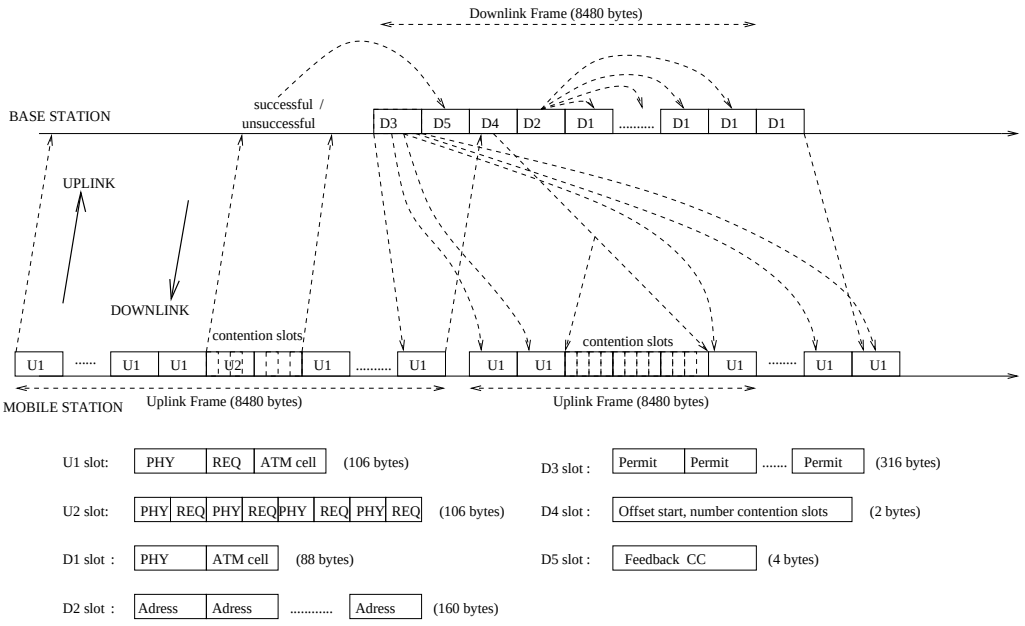

Figure 4: The Frame Structure

structure. Both frames have the same fixed length.

3.3.1. *Uplink Frame Structure*

The uplink frame consists of two slot types (U1 and U2), each having a length of 106 bytes. The total number of slots in an uplink frame is set to 80, resulting in a constant frame length of 8480 bytes

U1 slot: This slot is used to transmit an uplink ATM cell (53 bytes), together with a piggybacked request (8 bytes). A physical layer overhead of 45 bytes is used for error detection (FEC), a save guard time and sufficient training sequences. This results in a total length of 106 bytes.

U2 slot: A U2 slot is used to allow bursty VBR, ABR, GFR or UBR connections to transmit requests to the BS (see Sections 3.1.2 and 3.2). Each U2 slot is subdivided into 4 minislots and has the same length as a U1 slot. A minislot is used by one or more stations during a contention cycle (CC) to transmit a request. However, the requests transmitted within a minislot are further reduced in size and consist of the address of the MS using the minislot (1 byte), an indication of the ATM service category the permit is needed for (VBR, ABR, GFR or UBR: 2 bits) and 8 bits to indicate the queue length of this category in the MS. Hence, 194 bits remain to implement a safe guard time, some training sequences and some form of error control (FEC). This results in a total length of 106 bytes. We limit the number of U2 slots to 8, leading to a maximum of 32 minislots per frame (i.e., the parameter L defined in Section 3.2 equals 32).

3.3.2. *Downlink Frame Structure*

The downlink frame consists of five slot types. The D1 slots contain the downstream ATM cells, while the other four slot types are used for control and feedback information. These slots are grouped together and will be treated together with respect to training sequence and error correction.

D1 slot: This slot contains a downstream ATM cell (53 bytes), accompanied by the necessary physical layer overhead (training sequences, error detection: 35 bytes), resulting in a total of 88 bytes. Each downlink frame contains 80 D1 slots.

D2 slot: This slot is sent before the first D1 slot in a downlink frame and it is used to specify the destination addresses of MSs that are about to receive an ATM cell from the BS (in a D1 slot). Using a D2 slot avoids the need for each active MS to listen to the medium continuously in order to detect which of the D1 slots is destined for them. This leads to an important power consumption reduction. This slot has a length of 80 bytes, one for each reference to a D1 slot.

D3 slot: This slot is used to inform an MS of the permission to transmit a cell in the next upstream frame. Thus, it contains a variable number of permits, between (80 - 1) and (80 - 8) (80 being the total number of slots in an upstream link, whereas 1 and 8 are the lower and upper bounds for the number of U2 slots in an uplink frame). Each permit requires 3 bytes; hence, the length of a D3 slots requires at most 237 bytes.

D4 slot: This slot informs the MS of which of the 80 slots in the next upstream frame are declared as U2 slots, i.e., can be used for contention resolution. Since all U2 slots are positioned back-to-back, it suffices to specify an offset and the number of U2 slots. The offset can be specified by means of 13 bits, whereas the number of slots used for contention can be coded in 3 bits. This results in a total of 2 bytes for a D4 slot.

D5 slot: This slot contains the feedback information for the MSs that transmitted in a minislot in the previous uplink frame. For each minislot in the previous uplink frame, a nine bit feedback field is provided: the first bit indicates whether the transmission was successful or not and if so, the next eight bits repeat the MAC address of the successful MS. This is done for reasons of robustness and to avoid capture effects. As the maximum number of minislots in a frame is at most 32, 36 bytes are sufficient for a D5 slot.

The control and feedback slots (D2, D3, D4, D5) together are protected by a forward error correction (FEC) code. Moreover, they contain training sequences as well. A part of the remaining 1085 bytes is used for this purpose, the rest is used for signaling channels (synchronization, paging and others). The total downlink frame length is then 8480 bytes, which is exactly the same as the uplink frame length. Choosing equal lengths solves a number of synchronization problems, in particular with respect to the provided feedback (D5) and permit information (D3).

3.4. *The Bandwidth Allocation Algorithm*

The bandwidth allocation algorithm has to distribute permits among the active connections based on the service class the connection belongs to, the individual contract parameters of the connection and the current state of the different queues in the MSs (i.e., the bandwidth requirements of the MSs for each service category). In the following sections we describe how the permit distribution algorithm operates (also see [3,4] for related work on bandwidth allocation algorithms in an APON system).

The BS is provided with four permit FIFO queues: the CBR/VBR Permit FIFO, the ABR Permit FIFO, the GFR Permit FIFO and the UBR Permit FIFO queue. In addition, for each active MS, the BS maintains the following counters, the operation of which is explained in what follows: a CBR Permit Generation Counter (CBR-PGC) and for each of the service categories (rt-VBR, ABR, GFR and UBR) a Request Counter and a Countdown Counter.

3.4.1. *The Permit Generation Process*

We denote the net ATM capacity of the shared medium by C cells/sec.

- *CBR Connections*: Assume that for an active Mobile Station, referred to as MS_i, the sum of the peak cell rates of its active CBR connections is p_i cells/sec. The CBR Permit Generation Counter of that MS, $\mathrm{CBR\text{-}PGC}_i$, is initially set to the value C/p_i and is counted down at each slot. Whenever the $\mathrm{CBR\text{-}PGC}_i$ counter reaches a value less than or equal to zero, a permit is generated and put in the CBR/VBR Permit FIFO queue, and the $\mathrm{CBR\text{-}PGC}_i$ counter is increased by its initial value and starts counting down again.

- *rt-VBR, ABR, GFR and UBR Connections*: We describe the mechanism for the rt-VBR traffic only, the other cases operate in a similar way. Assume that for an active Mobile Station MS_i the sum of the peak cell rates of its active rt-VBR connections is p_i cells/sec. Requests that are received by the BS are stored in the rt-VBR Request Counter, namely, whenever a request arrives the BS updates the Request Counter accordingly. The rt-VBR Countdown Counter is initially set to the value $\min(C/p_i, P_m)$ (with P_m a predefined value) and is counted down at each slot. Whenever it reaches a value less than or equal to zero, a permit is generated and put in the CBR/VBR Permit FIFO queue, the rt-VBR Request Counter is counted down by 1 and the rt-VBR Countdown Counter is increased by its initial value. This process is repeated until the rt-VBR Request Counter is zero. Remark that ABR, resp. GFR and UBR permits are put in the corresponding FIFO queues. The reason for *spacing* the permits of an active MS (using the Countdown Counter) is to protect the wireless access system against violating sources. The aim is not to obtain a smooth traffic flow, as this could be achieved more effectively by a traffic spacer located in the network after the BS. The parameter P_m

is mainly introduced to improve the delay characteristics of slow connections (see Section 6.5).

3.4.2. *The Permit Allocation Process*

According to the guidelines obtained in Section 2.2, we distribute the permits by emptying the permit FIFO queues in a strict priority order: the CBR/rt-VBR FIFO queue has highest priority, followed by the ABR, GFR and UBR FIFO queues.

4. Support of ABR Congestion Control

As discussed in Section 2.2.2, the wireless access system has to implement an ABR traffic management scheme to ensure that the available bandwidth is distributed fairly among all ABR virtual circuits. The congestion control mechanism can be a Relative Rate Marking scheme or an Explicit Rate Marking scheme. In what follows we briefly show how an Explicit Rate scheme may be implemented.

The BS computes the available bandwidth regularly and distributes it fairly among the active MSs. To achieve this, the BS counts the total number of CBR and rt-VBR requests and the number of ABR requests that arrive during a certain period of time. Using these numbers, the BS may compute an overload factor by which a fair bandwidth share of each MS carrying ABR traffic may be computed. This fair share is then used to compute an Explicit Rate for each MS. Such an Explicit Rate algorithm has been proposed for an APON system in [5]. It is also applicable to a wireless ATM access system.

5. Performance Analysis of the ISA Scheme with Polling (ISAP)

This section investigates the impact of the parameters of the ISAP algorithm on the system performance, in particular on the delay and throughput characteristics. In Section 6 we investigate the performance of the overall MAC protocol on a packet level.

5.1. *System Description*

An analytical model to compute the delay distribution and the throughput when using the ISA algorithm combined with polling has been developed in [14] and is presented briefly in what follows. This short discussion is included in this paper because the analytical model presented in Section 6 makes use of these results.

The Address Space: For this analysis we assume that each MS has, at most, one connection of each traffic class. We define n as the size of the MAC-addresses (in bits). When an MS connects to the BS, possibly due to a handover, a unique MAC address is assigned in a random way similar to the procedure used to generate the flow label in IPv6. For the analysis we assume that there are 2^n MSs within the observed cell (i.e., all MAC addresses are being used).

The Input Traffic: We assume that the aggregated traffic generated by all the MSs follows a Poisson distribution with a mean of λ requests per frame. As the number of MSs is finite and equals 2^n, the probability mass lying beyond the value of 2^n is added to that of 2^n to make the distribution finite. In reality there exists a dependency between the addresses that compete during consecutive collision resolution cycles (CCs), nevertheless, we assume that this is not the case. Thus, the addresses of the MSs taking part in the scheme at the beginning of a collision resolution cycle are uniformly distributed over the complete address space.

The Number of Slots: To make the model more tractable, we assume that a frame can allocate enough $U2$ slots to support a full level of the tree. Thus, if the tree is resolved at level i we need $i + 1 - S_l$ frames for that purpose, where S_l is the starting level. The influence of this assumption on the performance was addressed in [15]. In order to make a fair comparison between the ISA protocol with and without polling we assume that polling requires at most one frame. The size of the remaining address space that triggers the polling mechanism is denoted by N_p. At each level i, the BS counts the number of collisions N_C at level i and depending on the result of this counting process it decides whether a switch to polling occurs.

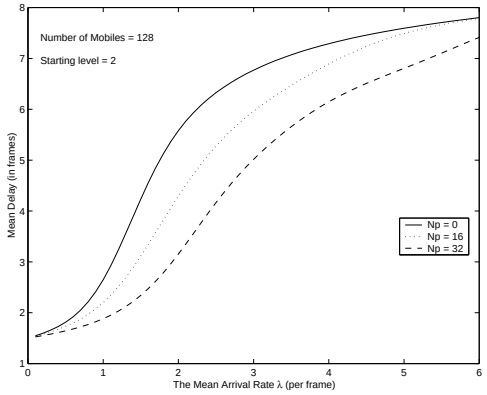

Figure 5: The impact of Polling on the mean delay.

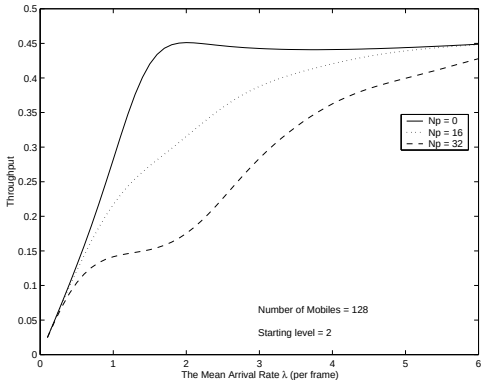

Figure 6: The impact Of Polling on the throughput

5.2. *Numerical Results*

In this section we study the impact of the offered traffic load λ, the trigger value N_p, the starting level S_l and related values B_l and B_m, on the mean delay, the delay density function and the throughput of the contention resolution scheme. The system parameters are set as follows: the number of mobiles is 128, the arrival rate λ of the generated traffic varies between 0.1 and 6 per frame, the three values studied for N_p are 0, 20 and 40, the starting level S_l varies from level 2 to 4, corresponding to 4 to 16 minislots (or 1 to 4 slots) and when studying a system with a dynamic starting level, B_l and B_m are set to 1 and 4 respectively. The boundary values are

$S_{min} = 2$ and $S_{max} = 4$.

5.2.1. *The Influence of the Polling Threshold*

Figures 5 and 6 show the influence of the polling feature of the ISA protocol on the mean delay and the throughput. As expected we observe a tradeoff between the delay and throughput characteristics: the sooner the ISA protocol switches to polling, the shorter the mean delay, but the lower the throughput. This tradeoff depends upon the value of N_p.

5.2.2. *The Influence of Skipping Levels*

Figures 7 and 8 illustrate the impact of S_l on the average delay and the throughput. From these results, we may conclude that a higher starting level has a positive impact on the delay especially for larger values of λ. Unfortunately, a high price is paid for this in terms of throughput if λ is small. Figures 7 and 8 show (for $N_p = 20$) that the dynamic scheme as proposed at the end of Section 3.2 solves this problem.

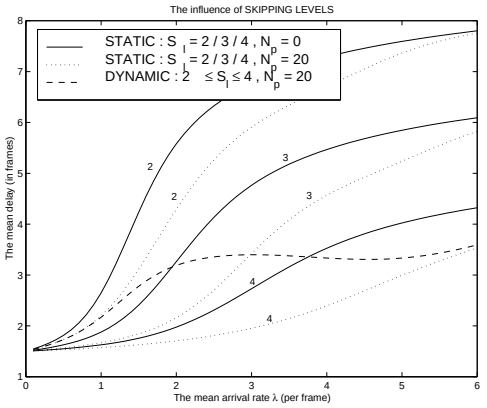

Figure 7: The influence of skipping on the mean delay.

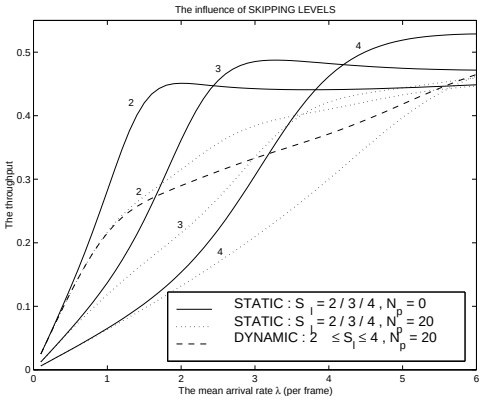

Figure 8: The influence on the throughput for $N_p = 0$, 20 and 40.

6. Packet Level Performance Characteristics

This analytical model for evaluating the packet level performance was presented in [13]. The description given below does not incorporate the parameter P_m defined in Section 3.4.1. It is however straightforward to generalize this description such that the parameter P_m is taken into account.

6.1. *System Description*

Packets offered by higher layer protocol are segmented at the AAL layer into a number of ATM cells. In general this type of traffic is very well suited for a piggybacking scheme, because a single request suffices to notify the BS of the presence of all the

ATM cells belonging to a packet. Clearly, there are two possible ways to transmit this request. First, if the packet was generated before the last ATM cell(s) of the previous packet(s) is (are) transmitted, piggybacking is used to deliver the request to the BS. Otherwise, the request has to be delivered to the BS using the contention channel, for which the ISA protocol combined with polling is employed. Moreover, the following assumptions are made:

- The real time permit queue, containing the CBR and rt-VBR permits, cannot be congested. This is easily guaranteed by making sure that the sum of the peak cell rates of all the real time connections is less than the link rate. Congestion is allowed in all permit queues other than the CBR/rt-VBR permit queue.
- In the protocol definition the frame length is fixed, while the number of $U1$ ($U2$) slots in such a frame is variable and determined by the ISA protocol. For the analysis, we assume that the number of $U1$ slots is fixed to F, while the number of $U2$ slots is still determined by ISA, resulting in a slightly varying frame length. As frames contain mostly $U1$ slots (the number of $U1$ slots is between 72 and 80), this assumption should hardly have any influence on the results. Moreover, the $U1$ slots in a frame are numbered from 0 to $F-1$.
- We assume that bit errors caused by the wireless medium can be fixed by the receiving node by means of the forward error correction code (FEC).

The system described above is modeled using a single server discrete time queue. The queue is fed by fixed length packets, although it is easy to incorporate any type of distribution for the packet lengths. A counter is associated to the service process. This counter is incremented by one every time unit and is reset at zero when it reaches a value of F. Clearly, it keeps track of the number of a $U1$ slot in a frame.

Furthermore, the discrete time process that governs the packet arrivals has a different time scale from the service process. One time unit for the arrival process corresponds to Q time units for the service process, with Q a divisor of F. Therefore, packet arrivals can only occur if the value of the counter associated to the service process is divisible by Q. Ideally Q equals one, meaning that arrivals can occur at any time instance. The reason for choosing $Q > 1$ is explained in Section 6.4.

Finally, the service time of a single packet depends upon: the packet length L, the PCR of the rt-VBR connection, the delay distribution W of the contention channel, the remaining service time of the preceding packets and the value of the counter associated to the service process. The first three values are identical for all packets.

For the system to be analytically tractable, the packets generated at the MS must be sufficiently large such that the time in-between the generation of the first and last permit destined for a packet of length L, that is, $\frac{L-1}{PCR}$ according to Section 3.4.1, is at least one frame time.

6.2. *The Packet Arrival Process*

The arrival process in an MS is a discrete-time Markov process belonging to the class of D-MAPs [1] and is similar to the Markov Modulated Bernoulli Process (MMBP). An m-state arrival process that belongs to this class is characterized by a rate vector $(\beta_1, \ldots, \beta_m)$, that contains the mean arrival rate associated with each of the m states, and an $m \times m$ transition matrix $\mathbf{D}$ that governs the transition probabilities between the m states. The matrix $\mathbf{D}$ can be written as the sum of two matrices $\mathbf{D}_0$ and $\mathbf{D}_1$:

- $(\mathbf{D}_0)_{i,j}$ equals the probability that no arrival occurs and a transition from state i to state j takes place.
- $(\mathbf{D}_1)_{i,j}$ equals the probability that an arrival does occur and a transition from state i to j takes place.

As opposed to the MMBPs, transitions between states are only allowed at arrival times. Therefore, $\mathbf{D}_0$ is a diagonal matrix $diag(1-\beta_1, \ldots, 1-\beta_m)$. Next, we denote $(\hat{\mathbf{D}}_1)_{i,j}$ as the probability that a transition occurs from state i to j provided that an arrival occurs. Thus, we have $(\hat{\mathbf{D}}_1)_{i,j} = (\mathbf{D}_1)_{i,j}/\beta_i$. Finally, the $m \times m$ matrices $\mathbf{A}_i^{(n)}$ contain the probabilities of having i arrivals during n time units of the arrival process. Because $\mathbf{D}_0$ is a diagonal matrix, it is easy to find the distribution for I, the interarrival time:

$$P[I = k] = \vec{\alpha}_1 \begin{pmatrix} (1-\beta_1)^{k-1}\beta_1 \\ \vdots \\ (1-\beta_m)^{k-1}\beta_m \end{pmatrix}, \tag{6.2}$$

where $\vec{\alpha}_1$ is the left stochastic steady state vector of $\hat{\mathbf{D}}_1$. What makes this arrival process interesting is that we can change the correlation between consecutive interarrival periods in a systematic way without changing the distribution I. By definition this correlation equals

$$corr = \frac{E[I_n I_{n+1}] - E[I_n]E[I_{n+1}]}{\sqrt{VAR[I_n]}\sqrt{VAR[I_{n+1}]}}, \tag{6.3}$$

where I_n is the interarrival time between packet n and $n+1$. The mean value $\mu = E[I_n] = E[I_{n+1}]$ and the variation $\sigma^2 = VAR[I_n] = VAR[I_{n+1}]$ can be computed from the matrices $\mathbf{D}_0$ and $\mathbf{D}_1$.

$E[I_n I_{n+1}]$ is found using the partial derivatives of the joint generating function $f(z_1, z_2) = \sum_i \sum_j P[I_n = i \cap I_{n+1} = j] z_1^i z_2^j$. Hence,

$$E[I_n I_{n+1}] = \vec{\alpha}_1 \; diag(1/\beta_1^2, \ldots, 1/\beta_m^2) \; \mathbf{D}_1 \begin{pmatrix} 1/\beta_1 \\ \vdots \\ 1/\beta_m \end{pmatrix}.$$

To obtain this result the identity $\sum_{i\geq 1} i(\mathbf{D}_0)^{i-1} = diag(1/\beta_1^2, \ldots, 1/\beta_m^2)$ was used.

We now demonstrate how the correlation can be changed in a systematic way without changing the distribution I. We define an infinite set of arrival processes $A(r), r \geq 1$ with the same rate vector $(\beta_1, \ldots, \beta_m)$. The matrices $\mathbf{D}_0$, $\mathbf{D}_1$ and $\hat{\mathbf{D}}_1$ corresponding to the process $A(r)$ are denoted by $\mathbf{D}_{0,r}$, $\mathbf{D}_{1,r}$ and $\hat{\mathbf{D}}_{1,r}$. For all the processes $A(r), r > 1$, the matrix $\mathbf{D}_{0,r}$ is the same diagonal matrix. $\mathbf{D}_{1,r}$ is defined as

$$(\mathbf{D}_{1,r})_{i,j} = \frac{(\mathbf{D}_{1,1})_{i,j}}{r} \qquad i \neq j$$
$$(\mathbf{D}_{1,r})_{i,i} = \beta_i - \sum_{j\neq i} (\mathbf{D}_{1,r})_{i,j}.$$

Thus, all arrival processes $A(r)$, $r > 1$, are determined by the choice of $A(1)$. It is easy to show that the interarrival distribution $I(r)$ is the same for all the processes $A(r)$. On the other hand, the correlation between successive interarrival periods increases with increasing r.

The sustainable cell rate SCR and the peak cell rate PCR of a connection that corresponds to an arrival process characterized by the matrices $\mathbf{D}_0$ and $\mathbf{D}_1$, are chosen as follows: $SCR = L/(\mu Q)$, $PCR = L/Q \max_i \beta_i$, where L equals the packet length and μ is the mean packet interarrival time. Notice, both the PCR and the SCR are expressed in time units of the service process.

6.3. *The Service Time*

The analysis is performed on a packet level, therefore we are only interested in the service time of packets and not in the service time of individual ATM cells.

By definition of the traffic scheduler in the BS, see Section 3.4.1, the permits for the different ATM cells belonging to the same packet are placed in the CBR/rt-VBR permit queue according to the PCR of the connection. From here on the CBR/rt-VBR permit queue is simply called the permit queue, except when stated otherwise. Although the permits are generated periodically by the BS, the presence of the permit queue introduces some jitter. As a result the interdeparture times of the corresponding ATM cells at the MS is not exactly 1/PCR.

As we observe the queue on a packet level, we are not interested in the interdeparture times of consecutive ATM cells of a packet, but only in the interdeparture time of the first and the last ATM cell of a packet. Because we assumed that the permit queue is never congested, we can approximate this interdeparture time by the time between the generation of their corresponding permits; the approximation improves as packets become larger.

To find the service time of a packet, the following two Remarks must be made:

(i) Piggybacking is possible if a packet finds a non empty transmission queue upon arrival, otherwise the MS makes use of the contention channel.

(ii) During frame n, the BS schedules the uplink transmissions for frame $n+1$, that is, a permit for the i-th $U1$ slot of frame $n+1$ is generated during the i-th $U1$ slot of frame n. Therefore, once the BS is notified of an ATM cell arrival at the MS, at least a full frame length passes before the actual transmission can occur.

Therefore, we distinguish three scenarios:

Scenario 1: The packet finds the transmission queue empty upon arrival. Piggybacking is no longer an option and the contention channel is used. Once the request is successfully transmitted, at least one frame time will elapse before the first ATM cell, that is part of this packet, is transmitted (see Remark 2). Therefore, the service time S_1 is chosen as follows: $S_1 = R + W + F + (L-1)/PCR$. The random variable R denotes the remaining time until the counter of the service process reaches zero again (in order to transmit the request the MS has to wait until a new frame starts to know where the $U2$ slots are located), W is the delay suffered on the contention channel (a multiple of F), F is a fixed value that corresponds to one frame (see Remark 2), L is the packet length and PCR is the peak cell rate of the connection ($(L-1)/PCR$ is the time between the transmission of the first and the last ATM cell part of the packet).

Scenario 2: The packet arrives in a non empty transmission queue, but the remaining service time R_S of the preceding packet(s) is smaller than one frame time. Due to the assumption on the packet length L, that is, $L - 1/PCR$ is at least one frame time (see Section 6.1), this scenario can never occur if more than one packet is backlogged at the MS. This observation is important for Section 6.4, where we solve the queueing model. Taking Remark 2 into account, all preceding ATM cells, that are part of the remainder of the preceding packet, have been scheduled for transmission by the BS. Thus, the service time S_2 of this packet depends on the remaining service time R_S of the preceding packet and is defined as: $S_2 = F - R_S + (L-1)/PCR$. Indeed, a request that informs the BS of the newly arrived packet is piggybacked to the next ATM cell that leaves the MS, the time until the next ATM cell leaves the BS is considered as negligible. The presence of the $F - R_S$ is due to Remark 2.

Scenario 3: The packet arrives in a non empty transmission queue and the remaining service time for the packet(s) in front is at least a frame time. Therefore, not all preceding ATM cells have been scheduled for transmission, otherwise the remaining service time would be less than a frame time (because of Remark 2). Also, due to the assumption on the packet length L, namely, $\frac{L-1}{PCR}$ is larger than F, we define the service time S_3 by L/PCR. Indeed, due to the assumption on the packet length L, an MS transmits an ATM cell approximately every $1/PCR$ time slot during the last frame time of the service time of a packet. Therefore, the newly arrived packet is able to inform the BS about its presence at least one frame time before the service time of the preceding packet ends. As a result, the permit for the first ATM cell of the newly generated packet is generated approximately $1/PCR$

time slots after the permit for the last ATM cell of the preceding packet.

6.4. *Solving the Queueing Model*

By observing the system at the time instants O_n that correspond with the transmission epochs of the first ATM cell of packet n, we can describe the system by the vector (N_n, P_n, q_n), where N_n denotes the number of backlogged packets (the one being served is not accounted for), q_n is the phase of the arrival process ($1 \leq q_n \leq m$) and P_n is the value of the counter associated with the service process at time O_n. To further reduce the state space, P_n is rounded to the nearest multiple of Q and therefore can be denoted as an integer value in the set $\{0, \ldots, F/Q - 1\}$. Obviously, the most accurate results are obtained if $Q = 1$. Notice that by observing the system at these epochs O_n, we know that the remaining service time of the currently serviced packet, that is, packet n, equals $\frac{L-1}{PCR}$ time units (of the service process). Moreover, the distance D_n between the observation points O_n and O_{n+1} equals

$$D_n = S_{n+1} + [A_{n+1} - C_n]^+,$$

where S_{n+1} is the service time of packet $n+1$, i.e., the next packet to be serviced, A_{n+1} is the arrival instant of packet $n+1$ and C_n the service completion time of the currently serviced packet, i.e., packet n. Finally, $[x]^+$ denotes $\max(x, 0)$. Thus, we can make use of the expressions for S_1, S_2 and S_3 to determine the transition probabilities.

Assume that we are in state $S = (N, P, q)$ with $N \geq 1$ at time O_n. Then, packet $n+1$ has arrived before the service completion time of packet n. Hence, $[A_{n+1} - C_n]^+ = 0$. Moreover, the remaining service time of packet n, at time A_{n+1}, is at least $(L-1)/PCR$ which is assumed to be larger than F. Therefore, scenario 3 of Section 6.3 applies for S_{n+1}, i.e., $D_n = S_3$. Therefore, in view of the expression for S_3, the transition probability $P(S, S')$ from state S to state $S' = (N + i - 1, P', q')$, with $i \geq 0$, equals

$$1[P' = (P + [\frac{L}{PCR}\frac{1}{Q}]) \bmod \frac{F}{Q}] \quad (\mathbf{A}_i^{([\frac{L}{PCR}\frac{1}{Q}])})_{q,q'},$$

where $[x]$ denotes x rounded to the nearest integer and $1[condition]$ obtains a value of one if *condition* is true and zero otherwise. The presence of the $1/Q$ in the superscript of $\mathbf{A}_i$ is due to the fact that the time unit of the arrival process differs from the time unit of the service process.

In state $S = (0, P, q)$ all three scenarios are possible, depending on the arrival time A_{n+1} of the next packet (see Figure 9). We let S' be (i, P', q'). In view of the expressions for S_1, S_2 and S_3 and keeping in mind that $\frac{L-1}{PCR}$ is the remaining service time of packet n, at time O_n, we obtain the following expression

$P(S, S') =$

$$1[P' = (P + [\frac{L}{PCR}\frac{1}{Q}]) \bmod \frac{F}{Q}] \sum_{s=1}^{[(\frac{L-1}{PCR}-F)\frac{1}{Q}]} \left(\mathbf{D}_0^{s-1}\mathbf{D}_1\mathbf{A}_i^{([\frac{L}{PCR}\frac{1}{Q}]-s)}\right)_{q,q'} +$$

$$\sum_{s=[(\frac{L-1}{PCR}-F)\frac{1}{Q}]+1}^{[\frac{L-1}{PCR}\frac{1}{Q}]} \left(\mathbf{D}_0^{s-1}\mathbf{D}_1\mathbf{A}_i^{(F/Q)}\right)_{q,q'} \quad 1[P' = P + F/Q + s \bmod \frac{F}{Q}] +$$

$$1[P' = 0] \sum_{s>[\frac{L-1}{PCR}\frac{1}{Q}]} \left(\mathbf{D}_0^{s-1}\mathbf{D}_1 \sum_x P[W = x]\mathbf{A}_i^{([\frac{F+xF}{Q}]+F/Q-(P+s) \bmod F/Q)}\right)_{q,q'},$$

with W the number of frames needed to successfully transmit a request to the BS using the contention channel. The first line of the equation above corresponds with Scenario 3, the second with Scenario 2 and the last with Scenario 1. In this equation, the difference between the arrival instants A_{n+1} and O_n, in time units of the service process, is denotes as s.

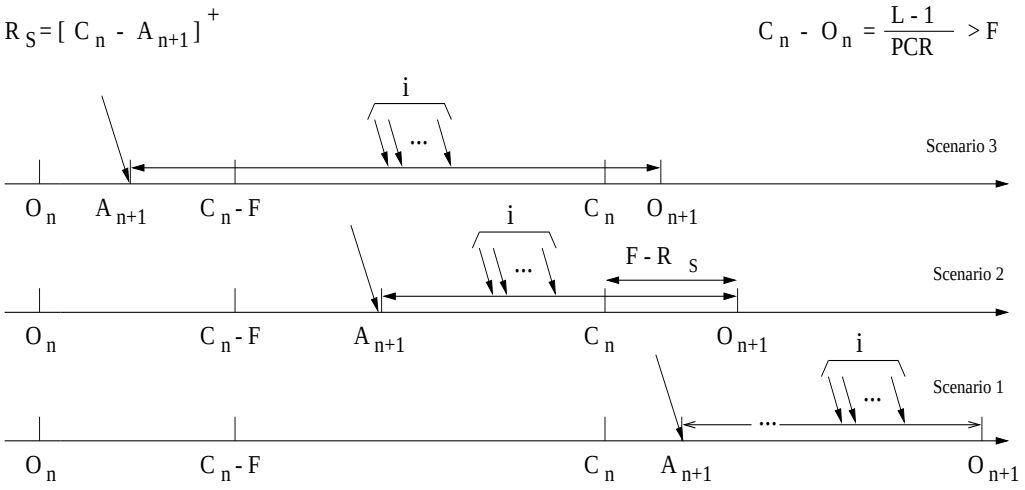

Figure 9: The scenario's for $N = 0$

By ordering the states (N, P, q) lexicographically, the probabilities $P(S, S')$ define a stochastic transition probability block matrix $\mathbf{P} = (\mathbf{Q}_{m,n})$. The matrices $\mathbf{Q}_{m,n}$ govern the state (P, q) transitions when the queue length changes from m to n. From the expressions for P(S,S'), it is clear that the resulting Markov Chain is of the M/G/1-type. To solve such a Markov chain the algorithm of Ramaswami is used [11] combined with [8] to find the required normalization factor. Due to the periodic nature of the frames, the matrix G, required to solve this type of Markov process, may become reducible in some rare cases.

Having calculated the stationary probability vector of the process at these epochs O_n, we calculate the queue length distribution X at the service completion times

C_n as follows:

$$P(X=k) = \sum_{P=0}^{F/Q-1} \sum_{i=0}^{k} \vec{\pi}_i(P) \mathbf{A}_{k-i}^{([\frac{L-1}{PCR}\frac{1}{Q}])} \vec{e},$$

where $\vec{\pi}_i(P)$ is a row vector of length m that contains the stationary probabilities of being in the states $(i,P,j)_{j=1}^{m}$ and $\vec{e}$ is a column vector of size m with each entry equal to 1. This equation is a consequence of the fact that the remaining service time at the observed epochs O_n equals $\frac{L-1}{PCR}$. Also, $P(X=0)$ is the probability that a packet needs to make use of the uplink contention channel (Scenario 1 applies).

Moreover, one can show that for an infinite capacity FCFS stationary discrete time queue with no simultaneous departures or arrivals, both the queue length distribution at the departure times and the arrival times are identical. Thus $P(X=k)$ is also the probability that a packet finds k customers (packets) in front upon arrival.

6.5. *Numerical Results*

In this section we study the influence of the SCR, the PCR, the variation of the interarrival times and the correlation between consecutive interarrival times on a number of performance measures of an MS carrying a single rt-VBR connection. The system parameters for the ISA protocol are set as follows (see [14]): the number of mobile stations considered is 128, the aggregated arrival process of all the MSs on the contention channel is Poisson with a mean of $\lambda = 1$ request per frame, the starting level S_l is static and equal to two, the value N_p that triggers the polling mechanism is 20 and a single instance of the ISA protocol is used. The P_m parameter defined in Section 3.4.1 is only used in the last subsection.

To find the delay distribution W we refer to [14]. Apart from the piggybacking probability we define the following two performance measures:

$$\begin{aligned} E &= \frac{L/PCR}{P_n L/PCR + P_0(L/PCR + E[W])} \\ D &= P_0 E[W] + P_n E[X-1 \mid X>0] L/PCR, \end{aligned}$$

with $P_0 = P(X=0)$ and $P_n = P(X>0)$. E is a measure for the efficiency of the scheme, while D is a measure for the delay experienced by packets in the MS.

The system parameter F is fixed at 72. Ideally the parameter Q should be set at 1, meaning that arrivals can occur at any time instance and the frame position P is represented by its true identity (and is not rounded to the nearest multiple of Q). On the other hand, the smaller Q becomes the larger the block matrices $\mathbf{Q}_{m,n}$ become and the more block matrices $\mathbf{Q}_{m,n}$ differ from zero, making the analytical model less attractive. Therefore, we set $Q = 8$ to improve the efficiency of the model. Numerical investigations (not reported here) have shown that the results for smaller values of Q are very well approximated by the model with $Q = 8$.

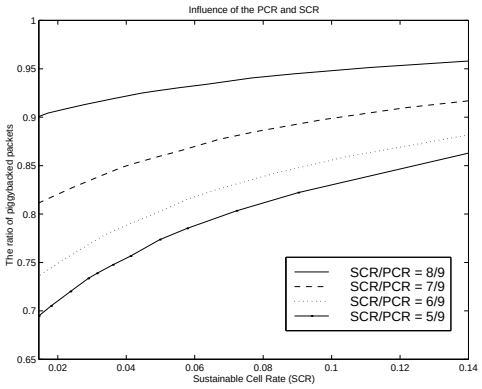

Figure 10: The impact of the SCR and the PCR on $1 - P(X = 0)$

6.5.1. *The Influence of the SCR and the PCR*

The packet length L in Figure 10 is set at 20, the rate vector β and the transition matrix $\mathbf{D}$ are the following:

$$\beta = (\ y \quad xy\ ), \quad \mathbf{D} = \begin{pmatrix} 1 - \frac{2xy}{5} & \frac{2xy}{5} \\ \frac{xy}{5} & 1 - \frac{xy}{5} \end{pmatrix}, \tag{6.4}$$

with $1/10 \leq y \leq 1/200$ and $x = 5/6, 2/3, 1/2$ and $1/2.7272$. When x is fixed and y changes the SCR and the PCR are varied proportionally, thus the ratio $\frac{SCR}{PCR}$ is fixed. When we change x with y fixed we get a fixed PCR but a variable SCR.

Figures 10 and 11 show that for a fixed SCR more piggybacking is used and a better efficiency E is realized as the ratio $\frac{SCR}{PCR}$ grows. Notice, this ratio is closely related to the burstiness of the traffic source. Secondly, although rt-VBR sources with a higher SCR achieve a higher piggybacking percentage for fixed $\frac{SCR}{PCR}$ ratios (an effect that increases with lower $\frac{SCR}{PCR}$ ratios), their efficiency E is smaller.

Figure 12 shows that better delays are achieved as the ratio $\frac{SCR}{PCR}$ decreases, a rather logical result as this ratio is a measure for the load of the queueing model. Secondly, the delay decreases as the SCR increases, because the workload is offered more gradually by higher bitrate sources, that is, as a results of the traffic scheduler in the BS the amount of work is offered to the server in multiples of L/PCR.

6.5.2. *The Influence of the Variation of the Interarrival Times*

In this section we keep the PCR and the SCR fixed and study the influence of any additional variation of the packet interarrival times that is not determined by the SCR and the PCR. As before, the packet length L is fixed at 20, the rate vector

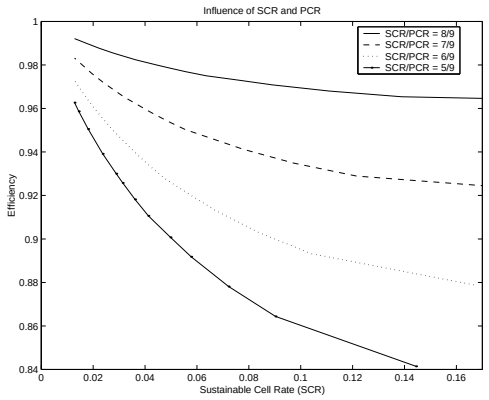

Figure 11: The impact of the SCR and the PCR on E

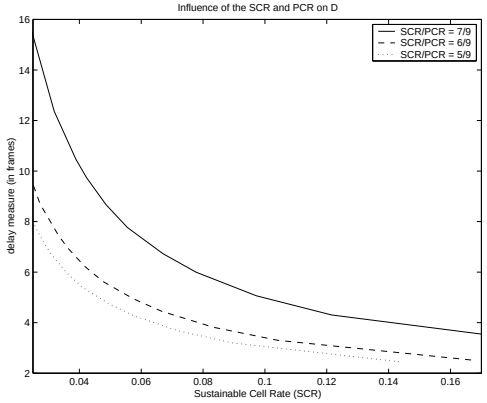

Figure 12: The impact of the SCR and the PCR on D

$\beta = (y, y/2 + x, y/2 - x)$ and the transition matrix $\mathbf{D}$ is the following:

$$\mathbf{D} = \begin{pmatrix} 1 - \frac{y}{50} & \frac{y}{100} & \frac{y}{100} \\ \frac{y}{200} & 1 - \frac{y}{100} & \frac{y}{200} \\ \frac{y}{200} & \frac{y}{200} & 1 - \frac{y}{100} \end{pmatrix} \tag{6.5}$$

with $\frac{1}{y} = 10, 13, 16, 20$ and x between 0 and $\frac{y}{2}$. We can change the variation while keeping the SCR and the PCR fixed by changing x with y fixed. On the other hand if x is fixed and y is increased the SCR becomes larger, whereas the ratio $\frac{SCR}{PCR}$ remains identical, namely, 0.6.

Figure 13 shows that the additional variation that is not caused by the choice of the SCR or the PCR has an important influence on the results. The larger the additional variation becomes, the better the efficiency E. Higher additional variation also results in a higher piggybacking percentage (this figure is not included). Therefore, the SCR and the PCR do not give a sufficient indication as to the piggybacking capabilities or the efficiency of an MS on a packet level. Figure 14 shows that the delay increases with increasing variation, as more variation means more bursty input traffic and thus longer queues.

A combination of the following observations might explain these results. Typically, having a higher variation with a fixed average and peak rate means that more traffic is sent at a high rate ($\geq$ SCR). Secondly, having large or exceptionally large interarrival periods makes little difference as both are very unlikely to result in piggybacking. On the other hand, to compensate for a single exceptionally large interarrival period, as opposed to a large period, we need much more short interarrival periods (because the SCR is fixed).

6.5.3. *The Influence of Correlation*

In this section we study the influence of the correlation between consecutive inter-

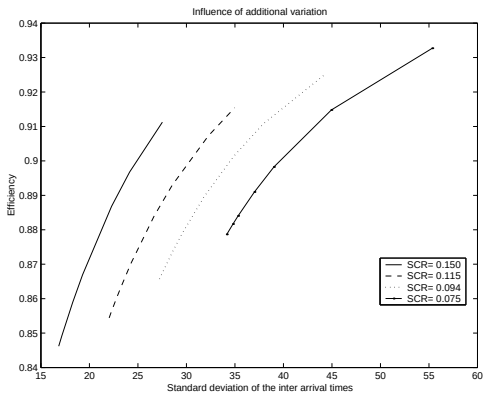

Figure 13: The impact of the standard deviation of the interarrival times on E

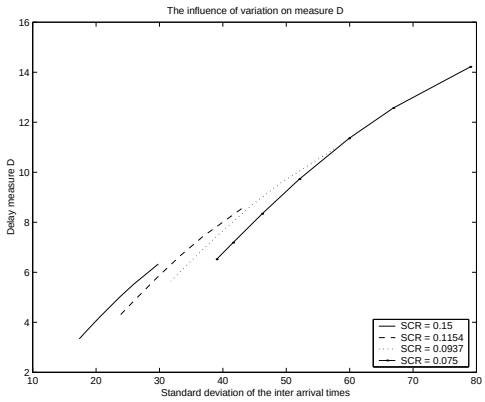

Figure 14: The impact of the standard deviation of the interarrival times on D

arrival times, while keeping the interarrival distribution I fixed. In Section 6.2 we developed a framework that allows us to do so. The arrival process $A(1)$ is the one that we used in Section 6.5.1 with x fixed at $1/2.7272$, thus $\frac{SCR}{PCR} = \frac{5}{9}$. We consider five different values for y resulting in as much different SCRs. In Figure 15 the parameter r is varied from 1 to infinity; notice that the correlation does not tend to one when r approaches infinity because of the geometric nature of the arrival scheme.

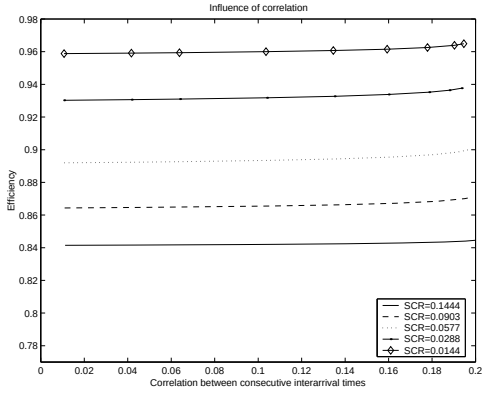

Figure 15: The impact of the correlation on the Efficiency E

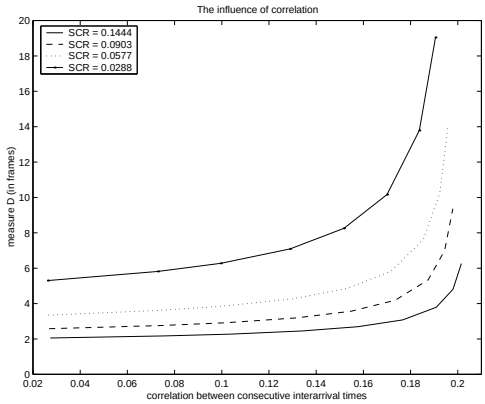

Figure 16: The impact of the correlation on the delay D

Figure 15 shows that this type of correlation is less important when studying the piggybacking capabilities or the efficiency of an MS (the figure that shows the impact of the correlation on $1 - P(X = 0)$ is not included). However, in Figure 16 it is shown that the correlation does have an important impact on the delay. Indeed, more correlation leads to longer delays especially for low bit rate traffic.

6.5.4. *The influence of the Parameter P_m*

The Figures 17 and 18 illustrate the influence of the SCR and the PCR on the efficiency and the delay for the same parameter choices as in Section 6.5.1, except that the parameter P_m defined in Section 3.4.1 is also being used.

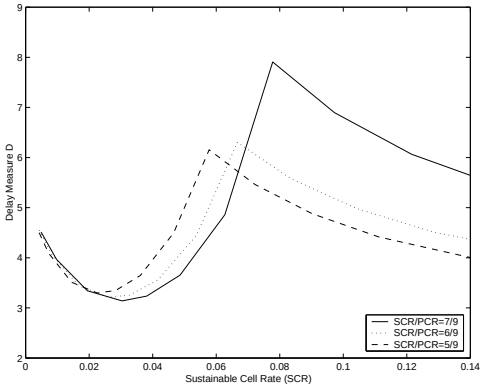

Figure 17: Delay for Data Traffic

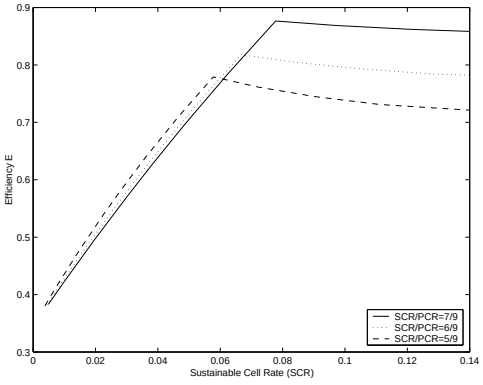

Figure 18: Efficiency for Data Traffic

The right hand side of both figures are identical to those in Section 6.5.1. The difference exists in the sudden change that occurs when the SCR decreases below the $0.06 - 0.08$ area. This knee is caused by the introduction of the parameter P_m (see Section 3.4.1). Without this parameter the delay of the low bit rate connections would become very high (we would also have a very high efficiency), i.e., the tendencies of the right half of the figure would continue in the left half (see Section 6.5.1). Thus, the introduction of P_m is a tradeoff between the efficiency and the delay characteristics. Figure 17 also shows that the parameter P_m has to be chosen carefully: a too large or too small value would result in high delays for slow connections. Indeed, when the value is too small, the contention channel is used for most of the ATM cells that are the first of a packet, while in the opposite case when the value is too large, arriving packets are spaced too much in order to achieve good delay bounds.

7. Conclusions

A MAC protocol for a wireless ATM network using the Identifier Splitting combined with Polling (ISAP) to inform the BS about the bandwidth needs of the MSs was proposed. Aside from a detailed description of the MAC protocol, the main contribution exists in an analytical performance evaluation of the influence of the packet arrival process characteristics in an MS, on the efficiency of the protocol and on the delay packets in an MS experience. In particular, the impact of the following parameters on the packet access delay and the protocol efficiency is investigated: the cell-level ATM traffic parameters, PCR and SCR, and the correlation between packet interarrival times.

Acknowledgements

The first two authors acknowledge the support of IWT project 980272, "Multi-service Network Technologies". We would like to take the oppertunity to express our gratitude to the reviewers of the paper for their valuable comments and remarks.

References

1. C. Blondia, "A discrete-time batch Markovian arrival process as B-ISDN traffic model", *Belgian Journal of Operations Research, Statistics and Computer Science*, **32**(3-4) (1993).
2. J.I. Capetanakis, "Tree algorithms for packet broadcast channels", *IEEE Trans. Inform. Theory*, **25**(5) (1979), 319–329.
3. O. Casals, J. García, and C. Blondia, "A Medium Access Protocol for an ATM Access Network", *Proc. of 5th Int. Conf. on Data Comm. Syst. and their Performance*, Raleigh, North Carolina (USA), Oct. 1993.
4. F. Panken, C. Blondia, O. Casals, and J. García, "A MAC Protocol for an ATM PONs supporting different service categories", *Proc. of the 15th ITC*, Washington, USA, June 1997, Eds. V. Ramaswami and P. Wirth, Elsevier, Vol 2, pp. 825–834.
5. F. Panken, C. Blondia, O. Casals, and J. García, "A MAC Protocol for an ATM PON supporting explicit rate congestion control for ABR traffic", *Proc. of Globecom '98*, Sydney, November 1998.
6. R. Goyal, R. Jain, S. Kalyanaraman, S. Fahmy, B. Vandalore, and S. Kota, "Selective Acknowledgement and UBR+ Drop Policies to Improve TCP/UBR Performance over Terrestrial and Satellite Networks", *Proc. of IC3N '97*, September 1997, pp. 67–88.
7. R. Goyal, R. Jain, S. Fahmy, and B. Vandalore. "Buffer Management for the GFR Service". *Submitted to Journal of Computer Communications*, (1998).
8. M.F. Neuts, *Structured Stochastic Matrices of M/G/1 type and their Applications*, Marcel Dekker Inc, New York, 1989.
9. D. Petras, "Medium Access Control Protocol for wireless, transparent ATM access in MBS", *RACE Mobile Telecommunications Summit*, (Cascais, Portugal) 1995.
10. D. Petras, and A. Krämling, "Collision Resolution in Wireless ATM Networks", *2nd MATHCOM*, Vienna, Austria, Feb 1997.
11. V. Ramaswami, "A Stable Recursion for the Steady State Vector in Markov Chains of M/G/1 Type", *Comm. Statist. - Stochastic Models*, **4**(1) (1988), 183–188.
12. B. Van Houdt, C. Blondia, O. Casals, J. García, and D. Vázquez, "A MAC protocol for wireless ATM systems supporting the ATM service categories", *Proc. of the 16th ITC*, Edinburgh, UK, Eds. P. Key and D. Smith, Elsevier, 1999, pp. 437–446.
13. B. Van Houdt, C. Blondia, O. Casals, and J. García, "Packet level performance characteristics of a MAC protocol for wireless ATM LANs", *Proc. of the 24th annual Conference on Local Computer Networks*, Lowell, USA, October 1999.
14. B. Van Houdt, and C. Blondia, "Analysis of an Identifier Splitting Algorithm combined with Polling (ISAP) for Contention Resolution in a Wireless Access Network", *IEEE JSAC*, **18**(11) (2000), 2345–2355.
15. B. Van Houdt, and C. Blondia, "Performance Evaluation of the Identifier Splitting Algorithm with Polling in Wireless ATM Networks", *Int. Journal of Wireless Information Networks*, **7**(2) (2000), 91–103.