PRIORITY SCHEDULING ALGORITHM FOR ATM WIRELESS NETWORK ACCESS

J. Meierhofer U. P. Bernhard P. Balmelli D. Bernasconi
Swiss Federal Institute of Technology, Communication Technology Laboratory
ETH Zentrum, Sternwartstr. 7, 8092 Zurich, Switzerland

Abstract – This paper discusses medium access issues for wireless asynchronous transfer mode (ATM) networks. In particular, a priority scheduling algorithm for a centralized access scheme has been analyzed using the Markov chain theory and the OPNET simulation package. The scheduler allocates resources to the terminals according to their priority class. Different algorithms (first-come first-serve, round robin, select largest queue) are proposed for scheduling the ATM traffic within the same priority class. The performance measures are cell delay, queue length, and cell loss ratio (CLR). It turns out that the scheduler algorithm strongly influences the cell delay which is most important for delay sensitive services. The CLR due to buffer overflows can be kept very low in general except for very bursty sources.

I Introduction

High-speed wireless data transmission and networking are key technologies for advanced portable/mobile computing and telecommunications applications. In the past years cellular phones and laptop computers became very popular which clearly shows that mobility will be a key feature of future communications systems. On the fixed network side developments concentrated on the evolution of a wireline infrastructure to support broadband multimedia traffic. These efforts for a single infrastructure for a wide range of services including data, video, and voice resulted in the development of the asynchronous transfer mode (ATM) technology.

In view of the emerging multimedia applications and the demand for mobility it is apparent that wireless broadband networks will become more important in the future. For mobile multimedia applications in general the same characteristics apply as in wireline ATM networks: high data rates for bursty traffic and end-to-end quality-of-service (QoS) guarantees. Wireless ATM has therefore been proposed for mobile multimedia applications and various research activities are now going on in this field [1, 2, 3, 4, 5]. Target data rates of 20 Mbps up to 155 Mbps are foreseen for these systems. Clearly, the standard ATM protocol stack has to be expanded by wireless-specific sublayers - medium access control (MAC) and data link control - in order to overcome the "shortcomings" of the radio channel.

Multimedia applications will make use of high-speed wide-area data networks. Wireless ATM therefore means wireless access to wireline ATM networks, i.e., mobile terminals will be attached to a wireline ATM network through high-speed radio links as shown in Figure 1. The wireless part consisting of the base station (BS) and several mobile terminals (MTs) is considered as a wireless access network. The MTs communicate only with (via) the BS. A centralized control scheme for a coordinated medium access is needed in order to efficiently distribute the scarce bandwidth resources while simultaneously taking into account the stringent QoS requirements of ATM connections.

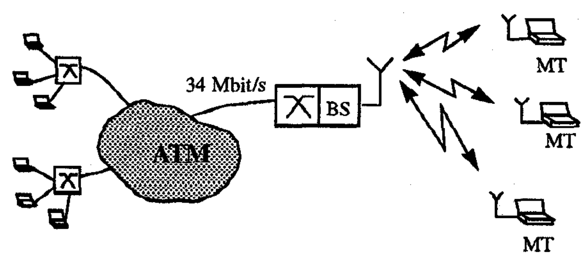

Figure 1: Wireless extension of ATM networks.

In the following we concentrate on the traffic coordination in the uplink (MTs to BS). We note that the situation at the air interface is similar to the one in an ATM switch for a fixed network when several incoming ATM connections have to be multiplexed to one outgoing line. The main difference between fixed and wireless networks is that in wireless networks packets have to be multiplexed at the air interface (before transmission), whereas this is done in the ATM switch in case of a wireline network. Consequently, buffering of data packets must be performed also at the terminals.

The BS must now control the medium access since it is the only station that can communicate with all users (see Figure 2). Each MT must inform the BS about buffered cells by sending INFO packets while the priority of each connection including the QoS parameters (the cell loss ratio CLR and the cell delay) is stored in the priority table. With this information the scheduler can read the buffers with high priority cells more often than others by sending GRANT packets to the corresponding MT. Concerning the rate of the signaling information exchange we note that there is a trade-off between two design objectives: little delay vs small signaling channel overhead.

II Model Assumptions

The source model for the process generating the ATM cells and the model defining the traffic in the system are based on identical assumptions as used in [6] for the performance evaluation of a first-come first-serve (FCFS) scheduler. It should be notified that a time division multiple access scheme is considered here and that the BS and all MTs are synchronized to the time slot schedule. The length of one time

0-7803-3777-8/97/$10.00 © 1997 IEEE

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

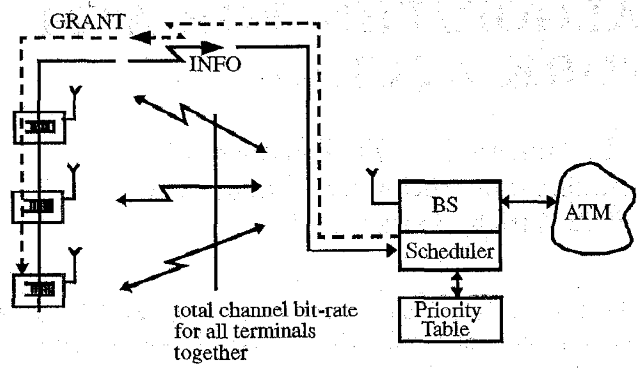

Figure 2: Multiplexing traffic at the air interface.

slot is equal to the duration T_c of one ATM cell. Each MT is modeled as an interrupted Bernoulli process (IBP) (see Fig. 3). In the on-state the IBP acts like a Bernoulli source with parameter ρ, in the off-state no packets are generated.

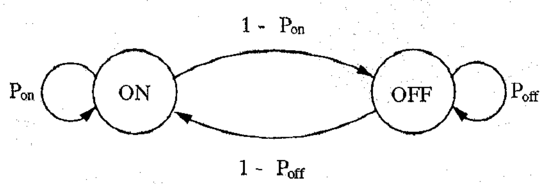

Figure 3: State diagram for the IBP.

The parameters τ_{on} and τ_{off} denote the time the source resides in the on or off-state; they are geometrically distributed with parameters P_{on} and P_{off}. With the slot time T_c the average on- and off-time become $\overline{\tau}_{\text{on}} = \frac{T_c}{1-P_{\text{on}}}$ and $\overline{\tau}_{\text{off}} = \frac{T_c}{1-P_{\text{off}}}$, respectively. The burstiness β of the source is defined as the fraction of time during which the IBP is in the off-state, i.e., $\beta = \frac{\overline{\tau}_{\text{off}}}{\overline{\tau}_{\text{on}}+\overline{\tau}_{\text{off}}}$. For the performance evaluation a source with an average bit rate independent of β is required. Therefore, the Bernoulli parameter is scaled according to $\rho = \frac{\rho_0}{1-\beta}$, with ρ_0 being the Bernoulli parameter for a burstiness $\beta = 0$.

To specify the traffic in the overall system we assume a total data rate of 34 Mbps for the radio link (see Fig. 1). The traffic model assumes $N = 12$ identical and independent video-phone terminals each transmitting at a mean bit rate of 2 Mbps (resulting in 2.208 Mbps including the ATM overhead of 5 bytes per cell). For the 34 Mbps data link this yields a Bernoulli parameter $\rho_0 \approx (2.208\text{ Mbps}/34\text{ Mbps}) = 0.0649$ and - with $N = 12$ - a system load of about 78%. To evaluate the influence of the burstiness we consider IBP sources with different values of β and an average on- and off-time of $\overline{\tau}_{\text{on}} + \overline{\tau}_{\text{off}} = 40$ ms (corresponding for example to a PAL video source with 25 frames per second).

For the performance evaluation we will assume that the reliability of the radio link is good enough such that retransmissions of erroneous packets may be neglected. This corresponds to a quasi-stationary situation where the MTs are located relatively close to the BS and do not move during data transmission. If bit errors and cell retransmission become significant, then these effects should be included in the model. Furthermore, delays due to radio propagation are not taken into account.

III Priority Scheduler

The MAC protocol coordinates the access to the radio link. The main objectives of the MAC and the scheduler (which is part of the MAC) are to maximize the utilization of the radio channel capacity (time slots in our case) and to minimize packet delay and loss. A priority scheduler is used to determine which connections are served first. The main design objectives for the scheduler are:

- Fairness: the scheduler must serve MTs of the same priority class with equal probability. MTs of a higher priority class are served before those of the lower priority classes.
- Little delay: the scheduler should minimize the cell delay for all MTs.
- Small signaling overhead: the information flow between scheduler and MTs should be as small as possible.

The priority scheduler controls the channel access for N MTs which are grouped into $\mathcal{P}$ priority classes. The priority class of each MT is marked by the parameter p with $p = 0, ..., (1-\mathcal{P})$ (decreasing priority with increasing p). Each MT sends an INFO packet to the BS to let it know the number of cells that have been generated since the last INFO packet was sent. The rate of these INFO packets is constant. The MTs then listen for scheduling assignments (GRANT packets) to transmit their cells in accordance with the slot allocation initiated by the scheduler (see Figure 2). GRANT packets are transmitted at an unspecified rate depending on the traffic demand announced by the INFO packets and the priority of the connection.

Two parallel processes run at every MT: The first process enqueues the generated ATM cells in the local buffer. INFO packets are then sent to the BS at a constant rate which is specified by the information period I_p. The information period (a certain number of time slots) is different for each priority class. The second process looks for GRANT packets coming from the BS. When such a packet arrives the assigned number of ATM cells is removed from the buffer and transmitted to the BS.

For every priority class p the BS maintains a table named Array_p. Each entry of Array_p corresponds to the number of cells enqueued in one MT of priority category p. Two parallel processes are running at the BS. The first process retrieves the INFO packets coming from the MTs and updates the table entry for the corresponding MT. The second process is the scheduling algorithm.

Scheduling algorithm

First, a search for a non-zero entry in the table of priority class 0 is executed. If all entries in Array_0 are zero, then the table for priority class 1 is examined. Generally, only if all entries in priority class p are zero, the table of the next priority class $(p+1)$ is examined. In case all tables are empty, in the next time slot the search starts again at Array_0. If at some priority class there exists a non-zero entry then the corresponding MT will be allowed to send some cells as described below. After that a new search for non-zero entries will always start with the table of priority

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

class 0 in order to guarantee that high priority cells are scheduled and transmitted before the others. Within one priority class the following scheduling mechanisms may be used to determine which terminal is allowed to transmit:

- First-come first-serve (FCFS): the cells are scheduled according to the FCFS principle which requires $I_p = 1$ because the scheduler must be informed about new cells as soon as they are enqueued in the buffers of the MTs. Also GRANT packets will be sent at every time slot. Hence, this scheme requires quite a large signaling overhead.
- Round robin: the MTs within one priority class are sequentially served every time this priority class is checked for non-zero entries. For example, after transmitting some cells from the second MT in class $p = 4$ the scheduler will start checking the third MT next time it returns to category $p = 4$.
- Select largest queue: the MT with the largest queue will be allocated the next time slots.

The following two chapters discuss the performance evaluation of the scheduler using Markov chain theory and Monte Carlo simulations. First, in Chapter IV we investigate the performance of an FCFS scheduler by applying Markov chain theory and Monte Carlo simulations. Second, in Chapter V the priority scheduler with a round robin scheme is described and studied by means of Monte Carlo simulations.

IV Markov Chain Theory for FCFS Scheduler

The Markov chain theory may be generally used to evaluate the performance of any scheduling scheme. However, for systems with many states the numerical evaluation of the result would require an extreme computational effort. We therefore study the FCFS scheduler in case all $N = 12$ MTs belong to the same priority class.

The FCFS scheduler maintains a queue of the requests in the order as they arrive at the BS. The number of pending requests in the queue is equal to the total number of enqueued cells in all mobile stations. In case all MTs have the same source characteristics and priority the average number of requests in the scheduler equals N-times the average number of cells waiting in the MTs. The average time elapsed between the arrival of a request and the transmission of the corresponding cell is equal to the average cell delay in the MT. It can be computed from the average number of requests by using Little's law [7]. Thus, the investigation of the scheduler queue gives insight in the mean values of queue length and delay in the MTs.

The number of requests in the scheduler queue is a Markov process with a steady state probability vector yielding the distribution of the queue length. The states of the Markov process are defined by the number of entries in the queue and the number of sources in the on-state. Since for the IBP the number of sources in the on-state is independent of the number of entries in the scheduler queue, the problem can be divided into two independent Markov processes, i.e., the process for the number of entries in the scheduler queue given that M sources out of N are permanently active and the process determining the number of active sources.

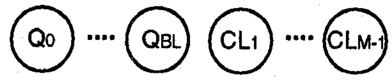

Figure 4: State diagram of the scheduler queue with M active sources.

In Fig. 4 the states of the Markov process are defined for the case where the scheduler queue has length BL and M sources are permanently active (no IBPs). Note that the states $CL_k, 1 \leq k \leq (M-1)$ define the cases where k cells are lost due to buffer overflow. Using these states it is possible to derive the CLR of the connection. Arranging the elements $\boldsymbol{\pi}_M[i]$ (with $i = 0, \ldots, BL + M - 1$) of the steady state probability vector $\boldsymbol{\pi}_M$ in the same order as the states in Fig. 4 and using the binomial distribution $b_{n,p}(k) = \binom{n}{k} p^k (1-p)^{n-k}$, the transition probability matrix becomes

$$\boldsymbol{P}_M = \begin{pmatrix} b_{M,\rho}(0) & b_{M,\rho}(1) & \cdots & b_{M,\rho}(M) & 0 & \cdots & 0 \\ b_{M,\rho}(0) & b_{M,\rho}(1) & \cdots & b_{M,\rho}(M) & 0 & \cdots & 0 \\ 0 & b_{M,\rho}(0) & b_{M,\rho}(1) & \cdots & b_{M,\rho}(M) & \cdots & 0 \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ 0 & \cdots & \cdots & 0 & b_{M,\rho}(0) & \cdots & b_{M,\rho}(M) \\ 0 & \cdots & \cdots & 0 & b_{M,\rho}(0) & \cdots & b_{M,\rho}(M) \\ \vdots & & & \vdots & \vdots & & \vdots \\ 0 & \cdots & \cdots & 0 & b_{M,\rho}(0) & \cdots & b_{M,\rho}(M) \end{pmatrix}.$$

The Markov process for the number of active sources is determined by the source parameters P_{on} and P_{off}. Assume that at some time t, n_1 sources out of N are active, i.e., $N - n_1$ sources are in the off-state. Then, the probability that i sources stay active and j sources become active in the next time slot is given by $b_{n_1,P_{\text{on}}}(i) \cdot b_{N-n_1,1-P_{\text{off}}}(j)$. Thus, the probability of $i + j = k$ sources being active in the next slot becomes $u_{n_1}(k) = \sum_{i,j} b_{n_1,P_{\text{on}}}(i) \cdot b_{N-n_1,1-P_{\text{off}}}(j)$ where the summation indices satisfy $0 \leq i \leq n_1$, $0 \leq j \leq (N - n_1)$ and $i + j = k$.

Combining the two independent processes yields a two-dimensional Markov process with the steady state probabilities $\pi_{M,n}$, where a state is being defined by the number of active sources M and the number of entries n in the scheduler queue (or the number of lost cells, respectively). By specifying the order in which $\pi_{M,n}$ appear in the steady state probability vector $\boldsymbol{\pi}$, we may now find the transition probability matrix

$$\boldsymbol{P} = \begin{pmatrix} u_0(0) \cdot \boldsymbol{P}_0 & u_0(1) \cdot \boldsymbol{P}_0 & \cdots & u_0(N) \cdot \boldsymbol{P}_0 \\ u_1(0) \cdot \boldsymbol{P}_1 & u_1(1) \cdot \boldsymbol{P}_1 & \cdots & u_1(N) \cdot \boldsymbol{P}_1 \\ \vdots & \vdots & & \vdots \\ u_N(0) \cdot \boldsymbol{P}_N & u_N(1) \cdot \boldsymbol{P}_N & \cdots & u_N(N) \cdot \boldsymbol{P}_N \end{pmatrix}.$$

Using the transition probability matrix $\boldsymbol{P}$ and the fact that the steady state probabilities have to sum up to unity, the desired steady state probability vector $\boldsymbol{\pi}$ can be computed numerically. With this vector it is then straightforward to compute performance measures such as the CLR, for example.

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

we consider the traffic model with $N = 12$ IBP sources as described in Section II. A rather small buffer length of $BL = 24$ has been assumed for the scheduler queue in order to gain some insight in the effect of buffer limitations on the queue length and the CLR.

We first consider the probability distribution $P(Q)$ of the number of cells Q in the scheduler queue. Figure 5 shows a comparison of $P(Q)$ as obtained by applying Markov chain theory and Monte Carlo simulations for a burstiness of $\beta = 0.75$. The simulation results are in very good agreement with the numerical results obtained from Markov chain theory.

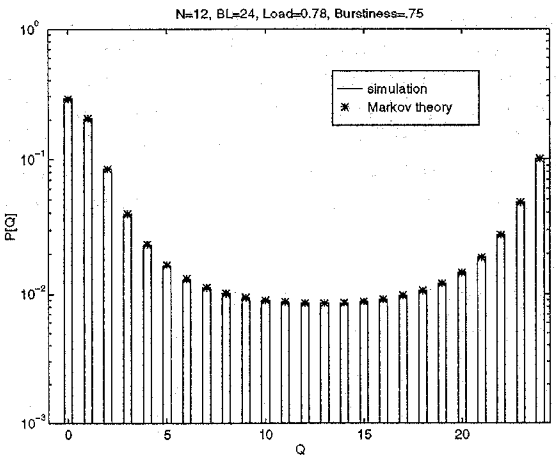

Figure 5: Probability distribution of the number of cells Q in the scheduler queue.

This figure shows a characteristic behavior of $P(Q)$ for situations with very bursty traffic (large value of β) and short buffers: when all sources are quiet (which is quite probable for bursty sources) the buffer is empty; on the other hand, if two or more active sources want to transmit data at a maximum rate in the same time interval the buffer fills very fast. As arriving cells are lost when the buffer is full it can be anticipated that the CLR will be rather high for that case. It is clear that the burstiness is much smaller than $\beta = 0.75$ in most practical cases. We observed that in case of low burstiness the buffer length becomes less important. In case of $\beta = 0$ a comparison of the cases $BL = 24$ and $BL = \infty$ from [6] shows hardly any difference in $P(Q)$ which means that even a small buffer of length $BL = 24$ will be seldom full.

The burstiness β has a big influence on the mean queue length $\bar{Q}$ and the CLR as it can be seen in Figures 6 and 7. The first measure is most important for delay sensitive services such as high data rate video because the queue length is directly related to the cell delay. Compared to the case of an infinite buffer length (see [6]) where $\bar{Q}$ grows exponentially for a high burstiness, $\bar{Q}$ grows only linearly here, resulting in a considerably reduced cell delay. The drawback of having a small mean queue length $\bar{Q}$ and a small cell delay is the high CLR. As it turns out from Figure 7 the CLR increases rapidly even for a small burstiness. The CLR can be reduced only by increasing the buffer length cell delay.

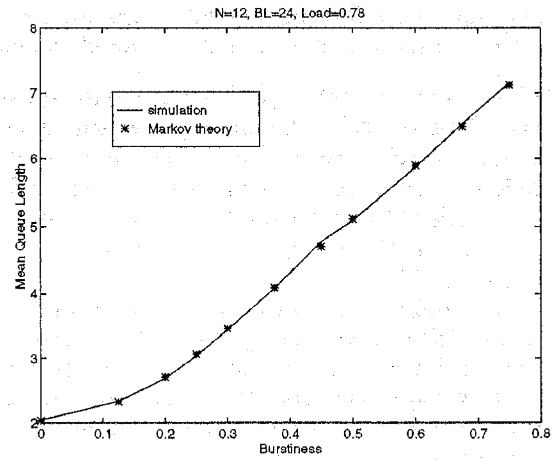

Figure 6: Mean queue length $\bar{Q}$ as a function of the burstiness β.

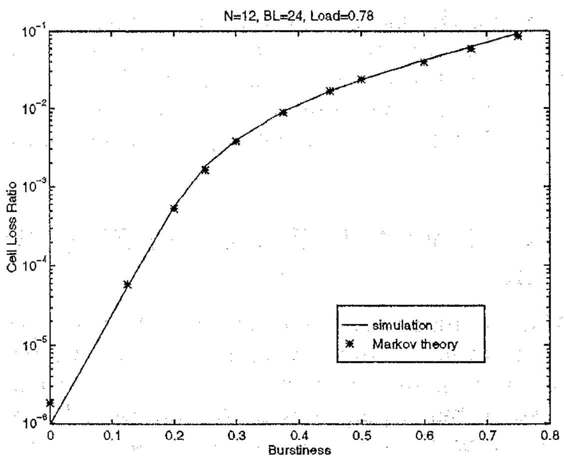

Figure 7: Cell loss ratio as a function of the burstiness β.

V Performance of the Priority Scheduler with Round Robin Scheme

We saw in the last chapter that in case of treating all sources equally (i.e., only one priority class) there is a clear trade-off between cell delay and cell loss. Multimedia traffic, however, can be divided into classes with different service requirements which implies to use priority schedulers. In the following we are going to investigate the suitability and the performance of the priority scheduler presented in Section III. The round robin scheme has been preferred over the FCFS for the traffic scheduling within the same priority class because it requires less signaling overhead. A detailed flow chart of the priority scheduler algorithm is depicted in Fig. 8. The round robin process in each priority class is implemented using a token pointing at the priority table entry of the next source to be served. If a non-zero entry has been

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

found, say Array_2[3] = x, a GRANT packet is sent to the third MT of priority class 2 to allow the transmission of n cells. The number n is determined by $\min\{x, X_p\}$, where X_p is the maximum number of cells that the scheduler may allow an MT of priority class p to transmit with one GRANT packet (this number should increase as p increases). After decreasing the table entry by n (n cells are scheduled for transmission now) the token is set to the next entry of that category (round robin). Before starting a new search for a non-zero entry with the table of priority 0 the scheduler waits n time slots (these are allocated for the transmission of the cells). During this waiting time the table entries will be updated by the INFO packets arriving from the different MTs.

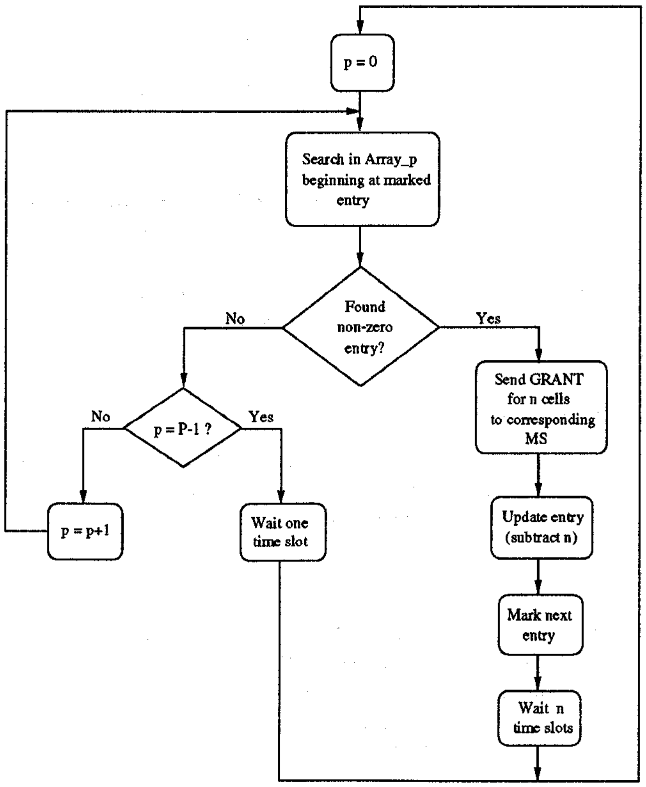

Figure 8: Flow chart of the priority scheduler.

According to ATM Forum UNI 4.0 specifications ATM networks offer five service classes with different QoS parameters [8]. Based on this setup we investigated the performance of a scheduler with five priority categories. In particular the performance evaluation aims to show that the design objectives proposed in Section II can be attained and tries to highlight the influence of some of the parameters of the scheduling scheme. The following scenario is considered for the performance evaluation:

- Twelve sources are assigned to the five priority classes as follows: Class 0: MT 0 and 1; class 1: MT 2 and 3; class 2: MT 4 and 5; class 3: MT 6, 7 and 8; class 4: MT 9, 10 and 11.
- The buffer length varies according to the priority class as it can be expected that low priority traffic experiences higher delays: $BL_p = 32, 64, 128, 256, 512$ for $p = 0, \ldots, 4$.
- The information interval I_p depends on the priority class in order to guarantee that high priority traffic can be scheduled as soon as possible: $I_0 = 2$, $I_1 = 4$, $I_2 = 8$, $I_3 = 16$, $I_4 = 32$.
- The burstiness is $\beta = 0.75$ in all situations.

The results of the performance evaluation presented here consider the two most important performance measures: the cell delay and the CLR. With respect to the parameters of the scheduler that influence the performance we concentrate on the parameter X_p, which denotes the maximum number of cells that the scheduler may allow an MT of priority class p to transmit. Two cases are compared in detail:

- Case 1 - Extract all cells: In that case $X_p = BL_p$, i.e., all cells are transmitted once the buffer is served.
- Case 2 - Extract X_p cells: The maximum number of cells that can be transmitted is a fraction of the buffer length BL_p: $X_0 = 2$, $X_1 = 4$, $X_2 = 8$, $X_3 = 16$, $X_4 = 32$.

Figures 9 and 10 show the probability distribution of the cell delay for Case 1 and 2 (the time unit for the delay D is $T_c = 12.5\mu s$). It turns out that in Case 1 all priority classes have a similar probability $P(D)$ for high delay values. In contrast, in Case 2, the resulting delay for high priority traffic is reduced considerably. Therefore, also the mean delay is much shorter.

The CLR of the twelve MTs is compared in Figure 11. It can be seen that the fairness of the scheduler scheme inside one priority class is assured, i.e., all MTs of one priority class have a similar CLR. When comparing the CLR of the different MTs for Case 1 and 2 we observe that high priority traffic (MT 0 and 1) will suffer a high CLR in Case 1, whereas the CLR is very low in Case 2. This is due to the fact that the buffer length BL_p of MTs with high priority has been chosen to be much shorter than for low priority classes. Hence, it may happen that for high priority connections buffer overflow occurs during the depletion of the buffer of a low priority connection. Consequently, the scheduler configuration studied in Case 2 is preferable, which was also concluded above in the study of the cell delay.

Fairness within each priority class is guaranteed and fairness among different priority classes is also fulfilled because some fraction of the channel capacity is allocated to low priority traffic although in the scheduler scheme there is no mechanism to guarantee that in general. This is due to the fact that for a system load of 78% high priority traffic will never use the full channel capacity. There always exist time slots where low priority sources may transmit their cells. However, the proposed scheduler scheme relies on the condition that all sources stay within the limits imposed by their traffic contract. In practice this must be ensured by usage parameter control (UPC).

VI Conclusions

We discussed and investigated medium access aspects for wireless extensions of ATM networks. A centralized control scheme for a coordinated medium access is required in order to efficiently accommodate ATM traffic on wireless links.

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.