Camera ready version of paper submitted for

“IEEE ATM ’96 Workshop”

San Francisco, CA, 25-27 August 1996

Authors: Dietmar Petras, Andreas Krämling

Title: **MAC protocol with polling and fast collision resolution for an ATM air interface**

Address: Aachen Univ. of Technology (RWTH)
Communication Networks (COMNETS)
Kopernikusstr. 16
52074 Aachen, Germany

Tel.: +49-241-80-7928

Fax.: +49-241-8888-242

E-Mail: {petras|akr}@comnets.rwth-aachen.de

WWW: http://www.comnets.rwth-aachen.de/~{petras|akr}

MAC protocol with polling and fast collision resolution for an ATM air interface

Dietmar Petras, Andreas Krämling

Communication Networks
Aachen Univ. of Technology
E-Mail: {petras|akr}@comnets.rwth-aachen.de
WWW: http://www.comnets.rwth-aachen.de/˜{petras|akr}

Abstract — **The paper deals with optimizations of the transmission of capacity requests in a medium access control (MAC) protocol for an ATM air interface, which enables a full integration of mobile ATM terminals into a fixed ATM network by realizing statistical multiplexing of ATM cells on the air interface. The fast notification of the base station about ATM cells arriving in mobile terminals is a critical item of each ATM-based centrally controlled MAC protocol. The paper introduces a new algorithm, that combines random access with polling by dynamically selecting the access scheme with highest throughput. For fast collision resolutions a high speed splitting algorithm is used. A detailed performance evaluation of the algorithm and protocol is given.**

I. Introduction

Following the general trend of extending services of fixed networks to mobile users, an advanced medium access control (MAC) protocol for a mobile radio system is investigated in this paper, which enables the full integration of mobile ATM terminals into a fixed ATM network [1]. This full integration requires a transmission of ATM cells over the air interface in such a way that the protocols of the ATM adaptation layer (AAL) are not involved [2]. Thereby the radio link behaves like an ATM multiplexer and is integrated transparently into the ATM network. The resulting protocol stack is shown in Fig. 1. At the air interface, the ATM physical layer and parts of the ATM layer are replaced by a wireless physical layer as well as an additional data link layer consisting of a logical link control (LLC) and a MAC sublayer. Possible applications of such a mobile radio system are ATM LANs, cellular mobile radio and Radio in the Local Loop (RLL) networks.

The transmission of ATM cells by means of virtual channels (VC) over an ATM air interface between several mobile terminals (MT) and a central base station (BS) leads to a MAC protocol, that realizes an ATM-like statistical multiplexing in the specific wireless, mobile scenario characterized by multiple access of not easy to coordinate terminals. By means of the MAC protocol the MTs have to share the common channel capacity fairly and according to their negotiated Quality of Service (QoS). Special attention has to be payed to the *maximum cell delay* of real-time oriented (CBR, VBR) services. Statistical multiplexing on a TDMA channel is used with a slot length τ_{slot} able to carry one ATM cell together with the necessary overhead for training sequence, synchronization, FEC and guard time.

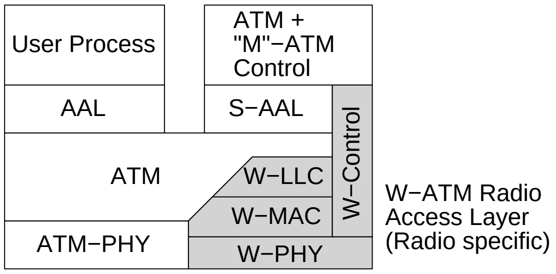

Figure 1: Protocol stack for full integration of mobile ATM terminals into a fixed ATM network

A fair (and efficient) medium access control based on single ATM cells is possible only, if the allocation of slots is controlled by a central instance (*central MAC coordinator*) [3]. In the system considered, this central MAC instance is running in the BS. Full duplex transmission by frequency division duplexing (FDD) is assumed. This avoids performance loss because of transmitter turn-around time. But it should be noticed, that the MAC protocol described below can easily be adapted to a time division duplexing channel.

The MAC protocol considered is called DSA++ (Dynamic Slot Assignment). Its functionality together with several variants and improvements to enable a co-operation with the LLC layer has been introduced in [4]. This paper focuses on the transmission of capacity requests from MTs to the central MAC coordinator located in the BS.

II. The DSA++ MAC Protocol

The basic concept of DSA++ is to consider a radio cell, consisting of a central BS and several MTs, as a distributed queueing system. The central MAC coordinator within the BS follows a service strategy that considers the QoS requirements of each VC and their instantaneous capacity requirements separately.

A BS operates a physical channel accessible by multiple MTs after executing a registration procedure. The allocation of capacity of the physical channel takes place slot–by–slot (horizontal reservation). It is controlled by the BS as central instance. Reservation of slot positions within a frame structure (vertical reservation) as performed in PRMA-like protocols [5] is not used, because this STDM (Synchronous Time Division Multiplexing) behaviour contradicts the statistical multiplexing of ATM cells and leads to higher queueing delays and wastes capacity due to reserved but unused slots .

The DSA++ protocol functions can be described as follows:

- Signaling of capacity (slot) assignments/reservations on the downlink by the BS
- Transmission of capacity requests on the uplink (by inband signaling, random access, polling) by the MT
- Service strategy in the BS to determine the order of ATM cell transmissions on uplink and downlink
- Random access versus polling (or both together)
- Fast collision resolution algorithm and stability control of random access protocol

The DSA++ protocol groups the signaling messages for several consecutive slots in a *downlink signaling burst* opening a signaling period of a specific length (see Fig. 2). Because only this downlink signaling burst is send in broadcast mode, power control can be used for all other bursts. A downlink signaling burst contains the following messages:

- a reservation message for each uplink slot of the signaling period
- an announcement message for each downlink slot of the signaling period
- a feedback message for each random access slot of the previous signaling period
- a field of other system signaling messages (paging channel, info channel, collision resolution, etc.)

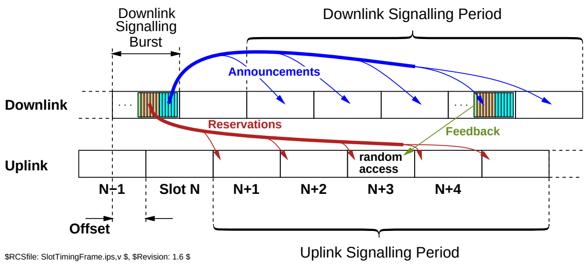

Figure 2: Downlink signaling scheme of DSA++ protocol

The number of messages within a downlink signaling burst and therefore the length of a signaling period depends on the available number of bits in a burst. Realistic period lengths are in the range from 8 to 15 slots. Transmission of announcement messages permits MTs to leave a physical channel for short time intervals (e.g. to scan other channels or switch to power save mode) without loosing synchronism or missing a message. The ternary feedback messages (collision, success, empty) are used to enable fast collision resolution for capacity requests transmitted in contention slots.

The influence of the period length, announcement messages and transmission of capacity requests on the performance of ATM cell transmission over uplink and downlink will be evaluated in section VI.

III. ATM Cell Scheduler at the Air Interface

The capacity of an ATM air interface (e.g. 34 Mbit/s per carrier [6]) is relatively low compared to multiplexers of fixed ATM networks. To ensure the negotiated QoS for each VC, the centralized MAC coordinator uses an ATM cell scheduler based on static or dynamic priorities [7].

The priority calculation for a VC or MT is based on their capacity demands. They are expressed by the so called *dynamic parameters* characterizing e.g. the number of waiting cells and their due-dates. The dynamic parameters representing the downlink are always up-to-date. The priority of a MT has to be based on the last transmitted dynamic parameters. The BS can estimate the number of newly arrived cells since the last transmission of the dynamic parameters by extrapolating the past history of the associated VCs, the accuracy of which depends on the type of VC. For CBR-like VCs nearly exact predictions are possible. For VCs of other types prediction cannot be as exact. Therefore it is useful, to transmit the newest dynamic parameters as frequent as necessary. This is performed in two steps: First, each MT adds the dynamic parameters to the header of each burst carrying an ATM cell. Second, a BS can request the actual dynamic parameters by polling or it can ask the MTs to transmit their parameters by random access. In order to minimize the overhead, polling and random access always take place in special short slots with $\frac{1}{8}$ to $\frac{1}{4}$ the length of a normal slot (depending on the used modem) [4].

Based on the dynamic parameters, the scheduler executes a service strategy to determine for each slot which MT should transmit or receive. Between ATM service classes static priorities are used (CBR $>$ VBR $>$ ABR $>$ UBR). Within the CBR and VBR classes the *relative urgency* discipline [8] is considered, where the priorities of ATM cells depend on their waiting time and their connection specific QoS requirements. Under this strategy the probability for cells being late (exceeding their due-dates) is minimized.

IV. Polling versus Contention

The fast transmission of dynamic parameters is of crucial importance for the observance of the required maximum cell delay τ_{dmax} of real-time oriented services. Therefore, their transmission is performed with high priority. The MTs are deciding about the frequency and time of transmissions. The reason for transmission of dynamic parameters is usually, that an ATM cell arrives at an empty queue so that this MT did not request further slots when transmitting the previous ATM cell. This kind of MTs is called to be in *contention mode*. $\mathfrak{M}_{rac}$ is the set of all MTs which are in contention mode at a moment. Note, that only those contention mode terminals have to transmit their dynamic parameters, in which a new ATM cell has arrived.

If a MT is operating multiple VCs at the same time, e.g. a mixture of ABR and CBR/VBR channels, it may happen, that the last transmitted dynamic parameters are only an inaccurate description of the real capacity requirements. For instance, a MT can request a slot for an ABR cell which will be handled with low priority. After the new arrival of an CBR/VBR cell with a much smaller due-date, the dynamic parameters expressing the increased urgency have to be transmitted again. It is difficult to describe the arrival process of the multiplexed cell stream and to determine its maximum cell delay τ_{dmax}. These descriptions are necessary for the below described algorithm inside the BS. Solutions for handling such parallel VCs have to take into account the functionality of the LLC layer. A detailed discussion on this topic can be found in [7]. To simplify things, the following algorithm is considering only one VC per MT.

It is the task of the BS to provide a sufficient number of short slots for transmission of dynamic parameters, which is performed by a combination of polling and random access. Therefore, it determines for each new signaling period the number of polling and random access slots to be provided on the uplink. The used algorithm has to be executed in real-time and thus has to be very runtime efficient.

The BS determines for each MT in contention mode the probability of a new arrival since the last transmission of their dynamic parameters. In order to simplify the algorithm and reduce the calculation effort, but also for lack of detailed specifications of the arrival process in MTs, we currently use only the mean cell rate for this estimation. Thus the arrival of ATM cells in MT k is modelled by a Poisson sources with mean rate $\lambda(k)$. The probability of at least one arrival during the interval $\tau_{idle}(k)$ since the last transmission of its dynamic parameters and with it the probability of success $p_{succ}(k)$ when polling this MT is:

$$p_{succ}(k) = 1 - e^{-\lambda(k)\cdot\tau_{idle}(k)}, \quad k \in \mathfrak{M}_{rac} \tag{1}$$

The precision of this estimation has to be improved especially for correlated sources like CBR [7] and for very bursty sources like some ABR or VBR services by using more precise models. They can easily be implemented in the algorithm if the necessary parameters are accessible from the QoS parameters.

The determination of the probability of success in random access takes place by estimating the number of MTs N_{rac}, which are willing to access. This is done by adding the probability of access $p_{succ}(k)$ of all MTs in contention mode:

$$N_{rac} = \sum_{k\in\mathfrak{M}_{rac}} p_{succ}(k) \tag{2}$$

With N_{rac}, the expected number of short slots necessary for the full resolution of the collision can be determined, which depends on the used collision resolution algorithm. From it, the throughput of random access ρ_{rac} can be derived. To simplify the algorithm, ρ_{rac} could be estimated to be constant with a value between 0.35 and 0.45. The influence of this simplification is of further study.

In general, the access mode with highest probability of success is chosen. But the connection specific maximum cell delay $\tau_{d_{max}}$ has also be taken into consideration. Therefore it is necessary, to poll the dynamic parameters of a time-critical VC in MT k after a certain time, independent of $p_{succ}(k)$. Related to the moment of the last transmission of its dynamic parameters, a MT has to be polled after $a_{poll} \cdot \tau_d(k)$ with meaningful values of parameter a_{poll} between 0.4 and 0.7.

Based on the above considerations, the algorithm for providing slots on the uplink for transmission of dynamic parameters works according to the following rules:

1. MTs with $p_{succ}(k) > \rho_{rac}$ will be polled.
2. MTs with $\tau_{idle}(k) > a_{poll} \cdot \tau_d(k)$ will also be polled.
3. MTs, which are to be polled, are taken out of the arrival process for random access, and a recalculation of N_{rac} is performed. The collision resolution algorithm uses this new N_{rac} to determine the necessary number of random access slots.
4. Each signaling period has to provide at least one random access slots to work around the inaccuracies of estimations using the Poisson model, and to allow access for contention mode terminals, at which a new ATM cell has arrived during the last signaling period.

The results of access in polling and random access slots have to be evaluated at the end of each signaling period.

But especially the results of random access have to be evaluated very carefully. Due to fading, a free random access slot does not indicate, that no MT has to transmit its dynamic parameters. Furthermore, a successful transmission does not necessarily mean, that there was exactly one MT, because of capture.

MTs, which were able to successfully request capacity for transmission of ATM cells (by transmitting dynamic parameters in polling or random access slots) are removed from $\mathfrak{M}_{rac}$. If no answer is received in a polling slot, a transmission error on the forward or backward direction has occurred and the previous status of the MT remains unchanged. If a MT transmits dynamic parameters with no capacity request, then $\tau_{idle}(k)$ and $p_{succ}(k)$ have to be reset to zero and the MT is left in $\mathfrak{M}_{rac}$. The same is performed for MTs, which have entered contention mode during the last signaling period.

Based on the results in random access slots, the collision resolution algorithm determines a new *a posteriori* estimation N_{rac+}. This value is then used to correct the estimations of $p_{succ}(k)$. Thereby, all MTs in $\mathfrak{M}_{rac}$ have to be treated in the same way, since no knowledge about the identity of the collided MTs is available.

The correction of $p_{succ}(k)$ is performed by the following steps:

1. Calculation of $N_{rac-} = \sum_{k\in\mathfrak{M}_{rac}} p_{succ}(k)$
2. Calculation of $c = \frac{K - N_{rac+}}{K - N_{rac-}}$, with K being the size of $\mathfrak{M}_{rac}$

In the next signaling period $\widetilde{p}_{succ}(k)$ is used instead of $p_{succ}(k)$ for all MT in $\mathfrak{M}_{rac}$:

$$\widetilde{p}_{succ}(k) = 1 - c \cdot e^{-\lambda(k)\cdot\tau_{idle}(k)} \tag{3}$$

After the collision has been completely resolved ($N_{rac+} = 0$), c is reset to 1 so that $p_{succ}(k)$ is used again for all MTs.

It is important to note, that the calculation of the correction factor c is always based on the original $p_{succ}(k)$ and not on $\widetilde{p}_{succ}(k)$ of the previous period.

A performance evaluation of this algorithm together with the description of the simulation model is presented in section VI.

V. Fast Collision Resolution Algorithm with Delayed Feedback

The signaling period of the DSA++ protocol gives rise to following constraints for random access:

- The result of random access is broadcasted to the MTs as feedback message inside the next downlink signaling burst. Thus, there is the situation with delayed feedback as described in [9].
- Each signaling period can provide nearly any number of random access slots. The maximum number of random access slots is only limited by the size of a downlink signaling burst, because it has to carry the necessary signaling messages.

Due to the previous described urgency of the transmission of dynamic parameters, the random access does not have to be optimized to maximum throughput but to short delays. A critical item is the delayed feedback, because a second random access of the same terminal is only useful, if the feedback of the first access has been evaluated before. Therefore, a dedicated short signaling period can be used to enable fast transmission of feedbacks [4]. This is especially useful during a collision resolution phase, when a period contains multiple random access slots. The resulting shorter duration of the collision resolution has to be payed with an increased signaling effort on the downlink.

The duration of a collision resolution is also influenced by the used collision resolution algorithm. The DSA++ protocol uses a splitting algorithm [10, 9]. Due to delayed feedbacks, multiple collision sets are resolved in parallel, corresponding to parallel splitting instances. Fig. 3 depicts, that a collision instance can be generated from the waiting set or from a collision set of the previous step/period by splitting. Several optimizations of the original splitting algorithm described in [9] cannot be utilized because of the requirement on short delay.

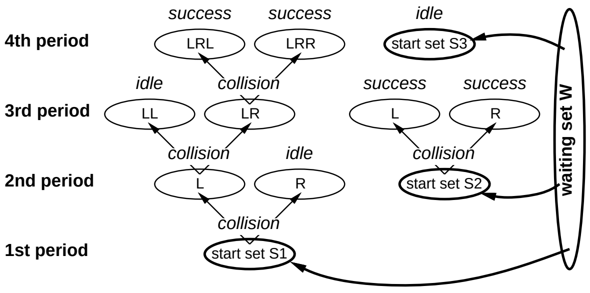

Figure 3: Splitting algorithm with parallel instances

Therefore, the maximum stable throughput/efficiency of the employed algorithm is 43%.

The ternary splitting algorithm [11] is a variant of the original binary algorithm. After a collision, the collision set is split into three subsets, resulting in more slots required for a collision resolution (40 % efficiency) compared to binary splitting, but less steps or periods, because more slots can be accessed in parallel. The delays with the DSA++ protocol are more resulting from the necessary number of signaling periods to resolve a collision and less from the number of used short slots. In section VI it is shown by simulations, that the ternary splitting algorithm leads to a better performance of the DSA++ protocol than the binary algorithm.

VI. Performance evaluation

The performance of the presented protocol with its several parameters has been evaluated by computer simulations using the SIMCO3++ tool [12]. The physical channel is assumed to be error free. The default parameters of the MAC protocol, used in all simulations, are as follows:

- Length of signaling period $L_P = 8$
- Ternary splitting algorithm
- Usage of announcement messages
- Usage of dedicated shortened signaling periods during collision resolutions

The following performance parameters are measured:

delay distribution of ATM cells: Delay from the arrival at the sender to its reception at the receiver

ratio of lost cells: Ratio of ATM cells exceeding their $\tau_{d_{max}}$

downlink signaling overhead: Ratio of downlink slots used for signaling bursts

The delays are compared to an ideal multiplexer with the same capacity utilizing a FCFS discipline, which gives the lower bound of delays.

The length of a simulation run is automatically controlled by the LRE algorithm [13] which limits the relative error to a predetermined value, and which is able to handle correlations in the measurement values. The relative error of the diagrams presented in this paper is always lower than 10%.

A. Scenario Description

Two scenarios are considered:

Scenario 1 is used to evaluate the performance of random access and collision resolution. It consists of 6 MTs each operating a bidirectional VBR VC with an overall load of 75% of the channel capacity on uplink and downlink each. The arrival process is modelled by a Poisson source with QoS requirements according to a maximum delay of $200\ \tau_{slot}$ (4 ms at 34 Mbit/s). The Poisson source is used because of its high variation of inter-arrival times resulting in a large population of contention mode terminals.

The following parameters can be varied:

- Length of signaling period L_P (see Fig. 5)
- Order of splitting algorithm (see Fig. 4)
- Usage of polling (see Fig. 6)
- Usage of announcement messages (see Fig. 8)

In Scenario 2 the following more realistic traffic mixture is used, assuming a channel capacity of 50.000 ATM cells per second corresponding to a gross bit rate of 34 Mbit/s [6]:

service	ATM class	#MT	load per VC	$\tau_{d_{max}}$
voice	CBR	5	0.35%	5 ms
video	VBR	5	11%	30 ms
data	ABR	2	15%	2 s

The overall load results to 85% of the channel capacity on uplink and downlink each. The voice service corresponds to a 64 Kbit/s PCM codec, the video service is according to [14], and the data service is modelled by a Poisson source.

B. Interpretation of Simulation Results

1) Length of Signaling Period The length of a signaling period affects the delays by different effects:

- The dynamic of the scheduler is influenced, because with longer periods the interval between a reservation or announcement message send in a downlink signaling burst and the associated slot becomes larger. This results in longer delays, because urgent new arrivals have to wait for the next period.
- The signaling overhead on the downlink decreases with increasing periods.
- Shorter periods allow earlier feedbacks resulting in faster collision resolution.

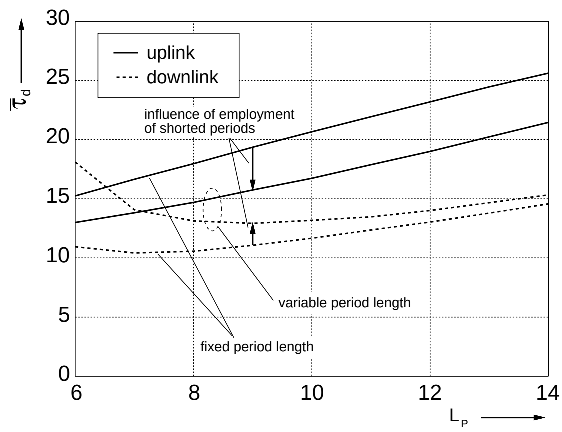

Figure 5: Mean delay $\overline{\tau}_d$ depending on length of signaling period L_P

For the downlink the effects are partly compensating each other (Fig. 5). For longer periods the negative influence of the decreasing dynamic dominates.

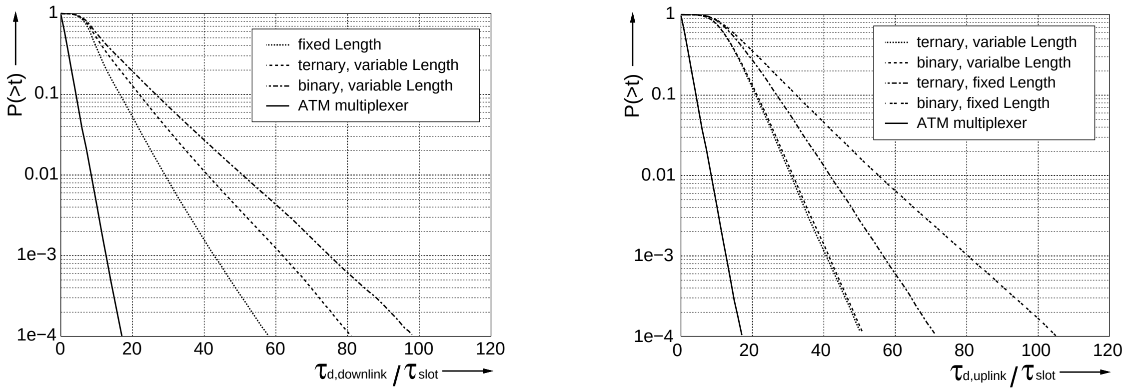

Figure 4: Complementary cell delay distribution function for binary and ternary splitting algorithm (left diagram: downlink, right diagram: uplink)

The usage of dedicated short signaling periods for fast collision resolution leads to a considerable reduction of the delay on the uplink, but increases the delay on the downlink because of the additional signaling effort. Therefore, shortening a period has to be used carefully, e.g. gradually decreasing length with ongoing collision resolution.

2) Order of the Splitting Algorithm The ternary splitting algorithm requires less steps or periods but more short slots to resolve a collision than the binary algorithm. With fixed period lengths the signaling effort is constant, and therefore the delay of the downlink is not affected by the order of splitting (Fig. 4). With shortened periods, the delay on the downlink is shorter for the ternary algorithm, as expected.

The influence of the splitting order on uplink delay is opposing. With shortened period, the order is of minor affect, but with fixed periods, the ternary algorithm leads to remarkable shorter delays.

Therefore, all other simulations have been carried out with the ternary splitting algorithm.

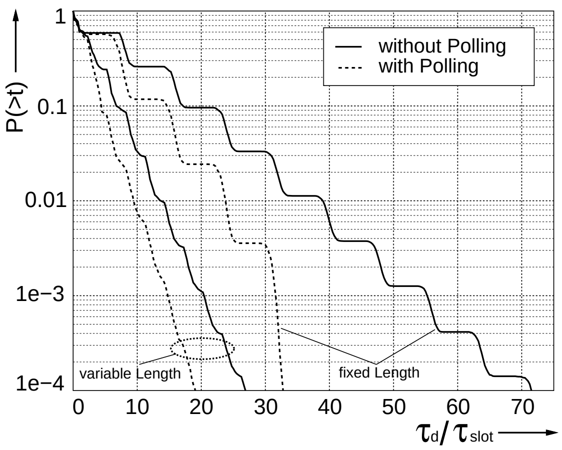

Figure 6: Influence of polling on delay of capacity requests

3) Polling The combination of random access with polling reduces the number of dynamic parameters transmitted in random access slots. Thus, the delay from the arrival of a new ATM cell at a terminal to the successful transmission of the corresponding capacity request is shorter, which is shown in Fig. 6. The steps in the curves result from the length of the signaling period. Each step represents one step of the collision resolution.

The positive effect of the combination of polling and random access especially occurs at lower load, when more terminals are in contention mode, but highly depends on the chosen scenario and will be smaller for non-Poisson sources.

4) Maximum Throughput Signaling reduces the capacity of the downlink whereas random access and polling slots require capacity of the uplink. Fig. 7 shows the mean delay over an overall load ρ when utilizing polling. It can be seen, that the capacity of the downlink is reduced by $\frac{1}{L_P}$, as expected.

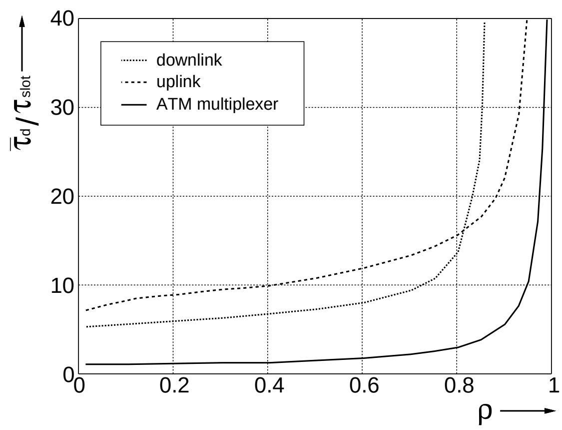

Figure 7: Mean delay $\overline{\tau}_d$ over overall load ρ

5) Influence of the Announcement Information In section II the advantage of announcement messages has been described. Here the costs of these messages are evaluated. The influence on the downlink delays is shown in Fig. 8. If announcement messages are used, only the ATM cells available at generation time of a downlink signaling burst can be considered. If not enough cells are available at generation time to fill the whole period, slots will be unused. New arriving cells cannot be transmitted in these free slots and have to wait for the next period resulting in lost efficiency and longer delays.

This effect can be minimized by declaring free slots for broadcast transmission. As a result all MTs have to listen to these slots and thus the receiver of bursts to be sent in these slots can be chosen later. The complementary delay distribution function which results from this improvement depends

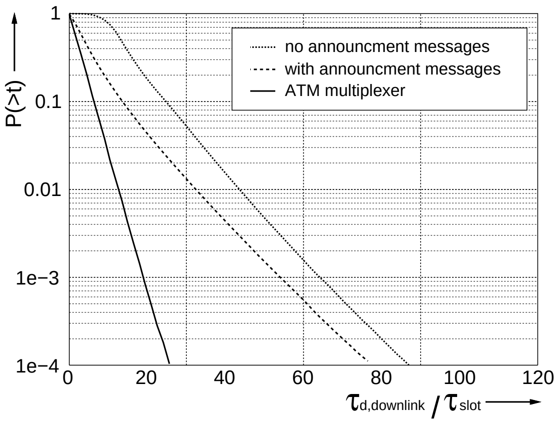

Figure 8: Influence of announcement messages on complementary downlink cell delay distribution function

on the scenario and will lie between the two curves shown in Fig. 8.

6) Scenario 2 with mixed traffic of different ATM classes Because of the different requirements on maximum delay and caused by the high overall traffic load, the ideal ATM multiplexer based on a FCFS discipline is not able to deliver all ATM cells in time. Since the MAC scheduler (utilizing a service strategy with a combination of static priorities between service classes and the relative urgency discipline inside a class) prefers urgent cells, no cells exceed their $\tau_{d_{max}}$.

ATM Class	DSA++ Down $\overline{\tau}_d$	DSA++ Down p_{CLR}	DSA++ Up $\overline{\tau}_d$	DSA++ Up p_{CLR}	ATM-Mux $\overline{\tau}_d$	ATM-Mux p_{CLR}
CBR	0.1ms	0%	0.1ms	0%	1.8ms	42.0%
VBR	4.5ms	0%	4.3ms	0%	7.7ms	0.2%
ABR	0.13s	0%	0.8s	0%	6.3ms	0%

Table 1: Mean delay $\overline{\tau}_d$ and cell lost rate $p_{CLR} = p(\overline{\tau}_d > \tau_{d_{max}})$ depending on the ATM service class (overall load ρ=85%)

The results show the needs for an ATM cell scheduler at an ATM air interface in a dramatical way. When not distinguishing between real-time and non real-time services, the QoS requirements can only be met for some *good-natured* services.

VII. Conclusions

The paper has described an improved algorithm for the DSA++ MAC protocol, which uses a combination of random access with fast collision resolution and polling for fast transmission of capacity requests.

Further studies will focus on models of realistic sources, used for the estimation of the probability of success of polling. Also analyses will be carried out to optimize the generation of new splitting instances from the waiting set.

VIII. Acknowledgements

The authors would like to thank Prof. Walke and all colleagues of the RACE II MBS project for their suggestions. Furthermore this work would not have been possible without the hard work and enthusiasm of the students working on their diploma thesis.

IX. References

[1] L. Fernandes, "Developing a system concept and technologies for mobile broadband communications," *IEEE Personal Communications*, vol. 2, pp. 54–59, Feb. 1995.

[2] J. Immonen, J.-M. Romann, and D. Plassmann, "Requirements and Protocol Architecture for MBS access to ATM network," in *RACE Mobile Telecommunications Summit*, (Cascais, P), pp. 324–328, Nov. 1995.

[3] B. Walke, D. Petras, and D. Plassmann, "Wireless ATM: Air Interface and Network Protocols of the Mobile Broadband System," *to be published in IEEE Personal Communications Magazine*, Aug. 1996.

[4] D. Petras, "Medium Access Control Protocol for wireless, transparent ATM access," in *IEEE Wireless Communication Systems Symposium*, (Long Island, NY), pp. 79–84, Nov. 1995. available at http://www.comnets.rwth-aachen.de/˜petras.

[5] D.-J. Goodman and S.-X. Wei, "Efficiency of packet reservation multiple access," *IEEE Trans. on Vehicular Technology*, vol. 40, pp. 170–176, Feb. 1991.

[6] M. Prögler, "MBS Air Interface Principles," in *RACE Mobile Telecommunications Summit*, (Cascais, P), Nov. 1995.

[7] D. Petras and A. Hettich, "Performance Evaluation of a Logical Link Control Protocol for an ATM air interface," in *to be published at PIMRC*, (Taipei, Taiwan), Nov. 1996.

[8] B. Walke, "Waiting-time distributions for deadline-oriented serving," in *Performance of Computer Systems* (M. Arato, A. Butrimenko, and E. Gelenbe, eds.), pp. 241 – 260, North-Holland Publishing Company, 1979.

[9] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[10] J. I. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. Inform. Theory*, vol. 25, no. 5, pp. 319–329, 1979.

[11] P. Mathys and P. Flajolet, "q-ary collision resolution algorithm in random access systems with free or blocked channel access," *IEEE Trans. Inform. Theory*, vol. 31, pp. 217–243, 1985.

[12] C.-H. Rokitanski, A. Guntsch, B. Bjelajac, and C. Wietfeld, "Simulation of mobile communication for performance evaluation of cellular networks, short-range and satellite communications," in *RACE Mobile Workshop*, (Amsterdam, NL), pp. 578 – 584, May 1994.

[13] F. Schreiber, "Effective control of simulation runs by a new evaluation algorithm for correlated random sequences," in *Proc. 12th Int. Teletraffic Congr. (ITC)*, (Torino), pp. 4.3B.1.1–9, 1988.

[14] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. Robbins, "Performance Models of Statistical Multiplexing in Packet Video Communications," *IEEE Transactions on Communications*, vol. COM-36, 7, pp. 834–844, July 1988.