MAC protocol for Wireless ATM: contention free versus contention based transmission of reservation requests

Dietmar Petras, Andreas Krämling, Andreas Hettich

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

Abstract — **In general, the users of wireless ATM terminals request the same functionality and Quality of Service as users of wired terminals. These user requirements can be transformed into the demand for building an ATM multiplexer *around* the air interface which is characterized by a radio channel inside the ATM layer. The main difference between this logical ATM multiplexer around the air interface and a normal ATM multiplexer is the distribution of the multiplexing function between wireless terminals and the base station. For the uplink this requires a frequent notification of the ATM cell scheduler in the base station about the status of the incoming queues inside the wireless terminals. This paper focuses on different transmission methods for transmitting capacity request messages (also designated as reservation request) over the uplink.**

I. Introduction

After the success of the *asynchronous transfer mode* (ATM) in the area of multimedia networks, a demand for the transparent integration of wireless ATM terminals into fixed ATM networks has become visible during the last years [1]. In general, the users of wireless ATM terminals request the same functionality and *Quality of Service* (QoS) as users of wired terminals. These user requirements can be transformed into the demand for building an ATM multiplexer *around* the air interface which is characterized by a radio channel inside the ATM layer [2]. At the ATM air interface inside this logical ATM multiplexer, the 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 *medium access control* (MAC) sublayer (cp. Fig. 1). Possible applications of such a mobile radio system are ATM LANs, cellular mobile radio and Radio in the Local Loop (RLL) networks.

The MAC protocol at the ATM air interface has to realize

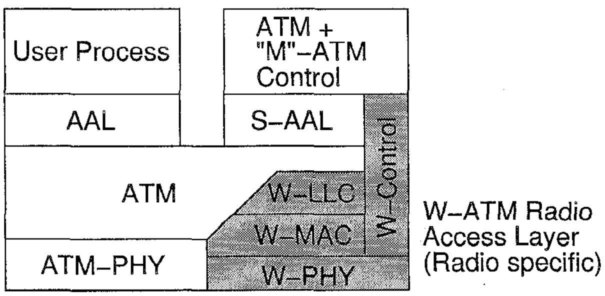

Fig. 1: Protocol stack of the logical ATM multiplexer at the ATM air interface

the statistical multiplexing inside ATM multiplexers in the specific scenario which is characterized by the competition of wireless terminals which are hard to co-ordinate. The terminals have to share the common channel capacity fairly and according to their negotiated 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 of the physical layer for synchronization, FEC, guard time, etc. [3].

The logical ATM multiplexer of one radio cell, consisting of several wireless terminals and a central base station, can be modelled as a distributed queueing system. As in normal ATM multiplexers with low data rates (e.g. 34 Mbit/s per carrier [1]) the observance of the negotiated QoS for each VC is only possible, if an ATM cell scheduler based on static or dynamic priorities is employed. Thus, the assignment of slots to virtual channels by the MAC protocol has to be controlled by a central instance. In the system considered, the ATM cell scheduler is running in the base station.

The MAC protocol considered is called DSA++ (Dynamic Slot Assignment). The frame structure with its different burst types as well as the signalling procedures for slot reservations on uplink and downlink have been introduced in [3]. Although the DSA++ protocol has originally been developed for a frequency division duplexing system, it can also be used for a time division duplexing system without major modifications. This paper focuses on the transmission of capacity requests from wireless terminals to the ATM cell scheduler located in the base station.

II. Requirements of the ATM cell scheduler

The ATM cell scheduler of the logical ATM multiplexer around the air interface executes a service strategy to determine for each slot the terminal which shall transmit or receive an ATM cell. Static priorities are used between ATM service classes (CBR $>$ VBR $>$ ABR $>$ UBR). Within the CBR and VBR classes the *relative urgency* discipline [4] 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. The scheduling of ABR cells can be combined with the execution of generic flow control. In this way the closed loop rate based flow control can be terminated in the base station and the load of looped back resource management cells is avoided [2].

The main difference between the logical ATM multiplexer around the air interface and a normal ATM multiplexer is the distribution of the multiplexing function between wireless terminals and the base station. This requires a frequent notification of the scheduler in the base station about the status of the incoming queues inside the wireless terminals, which is performed by transmitting capacity request messages (also designated as reservation request) over the uplink. There are the following constraints:

in time signaling To guarantee the required maximum delay $\tau_{d\,max}$ of realtime-oriented services the signaling of new arrivals has to be performed in time.

low signaling overhead The signaling overhead has to be minimized, since signaling of capacity requests and transmission of ATM cells are using the same radio resource.

The complex problem of signaling capacity requests can be divided into following aspects:

- Coding the status of the incoming queues
- Determining frequency and times of transmission of capacity requests
- Transmission mode of capacity requests

The procedure for coding the status of the incoming queues in wireless terminals is adjusted to the different service strategies for the ATM service classes in the ATM cell scheduler. Notice, that the scheduler distinguishes between virtual channels, whereas the MAC protocol distinguishes between terminals. Thus it is useful to combine the capacity requirements of virtual channels of the same terminal, at least for all virtual channels of the same service class. Furthermore, this will result in a reduced signaling overhead.

Since VBR ATM cells are served by the relative urgency strategy, their capacity request messages are containing the due-date of the most time-critical ATM cell together with the number of further cells with comparable urgency. In general, the same procedure applies to CBR. But since the inter-arrival times of CBR cells are deterministic, the base station has not to be informed about the arrival of each new cell. Instead, the base station can estimate the time of the next arrival, presumed that the arrival time of the previous cell has been transmitted together with the cell. Only in case of a faulty estimation, the wireless terminal has to transmit an explicit capacity request in order to resynchronize the estimation algorithm. The capacity requirements of ABR virtual channels can also be combined to one capacity request message which mainly contains the number of waiting ABR cells in order to avoid overflow of the ABR queues.

Since for each service class a specific coding algorithm is used, the term *dynamic parameters* has been introduced. It represents a generic data structure consisting of approximately two octets. It is used in contrast to the static (connection specific) QoS parameters and describes the dynamic behaviour of the capacity requirements of a terminal. The previous considerations result in a set of dynamic parameters in a terminal, one for each service class. When transmitting a capacity request, a terminal will send the dynamic parameters of the highest priority level, for which the request of capacity is necessary. The transmission can happen when requested (polled) by the base station. But whenever necessary, a terminal can also initiate the transmission by itself. This is useful especially for signaling the arrivals of ATM cells of realtime-oriented services.

The most efficient transmission of dynamic parameters is piggybacked to other packets (e.g. ATM cells or acknowledgements), whereas the unattached transmission in short bursts leads to a considerable overhead because of guard time and synchronization symbols. Thus there are the following possibilities for transmitting dynamic parameters:

- Piggybacked to ATM cells or other long messages in normal bursts
- Piggybacked to acknowledgements or other short messages in short bursts
- Unattached in short bursts

The transmission can be performed in reserved slots or in special random access slots. In order to reduce the amount of information lost in a collision, only the transmission of short bursts with random access is useful.

In the following it is assumed, that dynamic parameters are transmitted piggybacked to ATM cells whenever possible. If a terminal wants to initiate their transmission by itself, it has to use random access. Furthermore, the base station may poll the dynamic parameters of a specific terminal by assigning a short slot.

A terminal may initiate the transmission of its dynamic parameters for two reasons:

1. If it did not request further capacity, when transmitting its dynamic parameters last time, it has to signal the next arrival to the base station in order to be taken into consideration appropriately by the scheduling algorithm. This kind of terminals, which did not request further capacity, is called to be in *contention mode*. The other terminals, which did request further capacity and thus will be considered by the scheduler, are called to be in *reservation mode*.
2. If a terminal is operating virtual channels of different service classes, the situation may occur, that it did request capacity for a low priority class (e.g. ABR) and that a cell of a higher priority class (e.g. VBR) arrives afterwards. Although the terminal is in reservation mode, the scheduler will consider it with too low priority. Thus the transmission of the new VBR dynamic parameters piggybacked to an ABR cell may happen too late. Therefore it is useful, that the terminal transmits its VBR dynamic parameters in random access.

III. Contention free transmission of capacity requests

In contention free MAC protocols the dynamic parameters are always transmitted by polling (or piggybacked to ATM cells). This reduces the complexity of the MAC layer, since complex collision resolution algorithms are not to be implemented. Wireless terminals are only allowed to transmit their dynamic parameters after an explicit or implicit invitation of the base station, but they can no longer initiate the transmission by themselves. Therefore, each station has to be polled at least after a specific interval, the length of which depends on the virtual channel with the shortest maximum delay operated by this station.

It has to be noted that even in contention free MAC schemes a specific narrow-band channel with random access for registration of new terminals is necessary. It can be realized in combination with a beacon and paging channel superimposed on the normal MAC protocol by a super frame structure.

In this paper we consider a polling algorithm with implicit reservation of poll slots. This has the advantage of a very low signaling effort at the downlink for slot reservation messages resulting in a reduced signaling overhead. In every frame, each terminal will have at least one possibility for transmitting its dynamic parameters. This is normally performed piggybacked to ATM cells. If a terminal is not allowed to transmit an ATM cell in a frame, it will get assigned a poll slot at the end of the frame. If the terminals are numbered by a continuous, increasing scheme, an explicit signaling of poll slot reservations is not necessary. In Fig. 2 the frame structure with the reservation scheme for ATM cell slots on downlink and uplink and poll slots on uplink is illustrated. A frame (in [3] designated as signaling period) is opened by a *downlink signaling burst*, that contains one announcement message for each downlink ATM cell slot and one reservation message for each uplink ATM cell slot. This burst is followed by the downlink ATM cell bursts. Ahead of the uplink ATM cell bursts, a transceiver turn-around interval is inserted. The frame is finished with the poll slots and another transceiver turn-around interval.

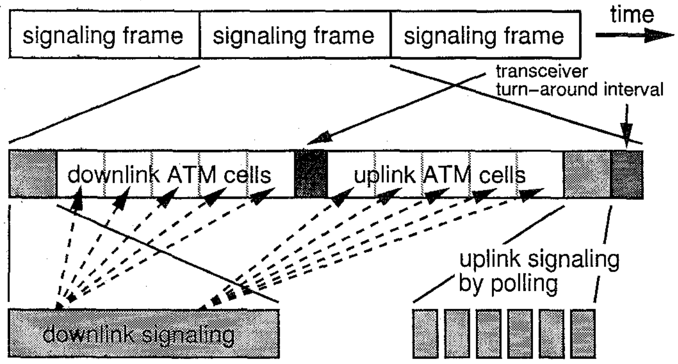

Fig. 2: Signaling frame structure of contention free MAC scheme

In section VI the performance of this MAC scheme will be evaluated and compared with a contention based MAC scheme.

IV. Contention based transmission of capacity requests

The usage of random access gives terminals the possibility to initiate the transmission of their dynamic parameters by themselves. This has a considerable advantage for bursty and realtime-oriented VC especially for VBR and narrowband CBR services. With random access a stable and fast algorithm for collision resolution is necessary.

The frame structure of the DSA++ protocol gives rise to following constraints for random access [5]:

- The result in a random access slot is broadcasted to the wireless terminals as feedback message inside the next downlink signaling burst. Thus, there is the situation with delayed feedback as described in [6].
- Each frame can provide nearly any number of random access slots. The maximum number is only limited by the size of a downlink signaling burst, because it has to carry the necessary signaling messages.

Due to the urgency of the transmission of dynamic parameters, the random access is not to be optimized for throughput but for 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, dedicated shortened frames may be used to enable fast transmission of feedbacks [3]. This is useful especially during a collision resolution phase, when a frame contains multiple random access slots. The resulting shorter duration of the collision resolution has to be payed with an increased signaling effort.

The duration of a collision resolution is also influenced by the used collision resolution algorithm. The DSA++ protocol uses a splitting algorithm [7, 6]. 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/frame by splitting. In [5] it has been pointed out, that the ternary splitting algorithm leads to a better performance of the DSA++ protocol than the binary algorithm, since delays are more resulting from the necessary number of frames to resolve a collision and less from the number of used short slots.

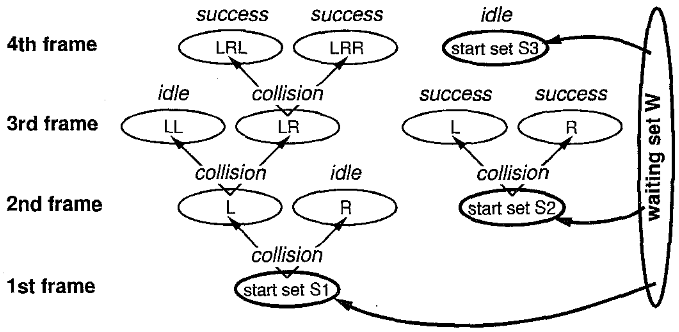

Fig. 3: Binary splitting algorithm with parallel instances

In the algorithm considered in [5] the splitting of the collided terminals into two or three subsets happens by independent stochastic experiments (e.g. coin flipping). But in our specific scenario with pico cells and thus a limited and small number of terminals per cell, splitting by node identifiers will often result in a better performance. As a prerequisite all active terminals of a radio cell have to be numbered by an as small as possible number set with the dimension n (number of binary or ternary digits, depending on the splitting order).

The gain in employing this identifier splitting has been evaluated by analysis and simulations. In [5] it has been described, how the complex arrival process of new terminals in the waiting set of random access can be estimated. Terminals from the waiting set are allowed to enter a new collision set with a specified probability that is calculated by the base station based on the estimated size of the waiting set. To simplify things for the analysis, we assume an equally distributed arrival rate of random access jobs in registered terminals resulting in a Binomial distributed number of terminals in the start set of a new collision phase.

In the case of binary coin flipping ($p_{left} = p_{right} = \frac{1}{2}$), the expected number of random access slots $S_c(m)$, necessary to resolve a start set with m terminals, is calculated by the recursive equation (1), which is valid for $m \geq 2$. The start conditions of the recursion are $S_c(0) = S_c(1) = 1$.

$$
\begin{aligned}
S_c(m) &= 1 + \sum_{i=0}^{m} \left(\frac{1}{2}\right)^m \binom{m}{i} \left(S_c(i) + S_c(m-i)\right) \\
&= \frac{1 + \left(\frac{1}{2}\right)^{m-1} \sum_{i=0}^{m-1} \binom{m}{i} S_c(i)}{1 - \left(\frac{1}{2}\right)^{m-1}} \qquad (1)
\end{aligned}
$$

For identifier splitting, the splitting of the collision set cannot be modelled by a set of independent stochastic experiments. With each further splitting, the set of remaining possible numbers in one of the new subsets is halved. The number of terminals in the left or right subsets is according to a hyper-geometrical distribution. It can be seen, that each collision is resolved after at least n splitting steps. The expected number of random access slots $S_{id}(m, n)$, necessary to resolve a start set with $m \leq 2^n$ terminals, is calculated by the recursive equation (2), which is valid for $m \geq 2$. The start conditions of the recursion are $S_{id}(0, x) = S_{id}(1, x) = 1, x \geq 0$.

$$
\begin{aligned}
S_{id}(m,n) &= 1 + \sum_{i=\max(0,m-2^{n-1})}^{\min(m,2^{n-1})} \frac{\binom{2^{n-1}}{i}\binom{2^{n-1}}{m-i}}{\binom{2^n}{m}} \left(S_{id}(i,n-1) + S_{id}(m-i,n-1)\right) \\
&= 1 + \frac{2}{\binom{2^n}{m}} \sum_{i=\max(0,m-2^{n-1})}^{\min(m,2^{n-1})} \binom{2^{n-1}}{i}\binom{2^{n-1}}{m-i} S_{id}(i,n-1) \qquad (2)
\end{aligned}
$$

The recursions have been solved numerically. For the determination of the throughput ρ_{id} of the identifier splitting algorithm the Binomial distribution of the number of terminals in the start set with mean $\overline{m}$ has been used:

$$
\rho_{id}(\overline{m}) = \frac{\overline{m}}{\sum_{i=0}^{2^n} S_{id}(i,n) \cdot \binom{2^n}{i} \left(\frac{\overline{m}}{2^n}\right)^i \left(1 - \frac{\overline{m}}{2^n}\right)^{(2^n-1)}} \qquad (3)
$$

The determination of the throughput ρ_c of the coin flipping splitting algorithms is based on a Poisson distribution of the number of terminals in the start set with mean $\overline{m}$:

$$
\rho_c(\overline{m}) = \frac{\overline{m}}{\sum_{i=0}^{\infty} S_c(i) \cdot \frac{\overline{m}^i}{i!} e^{-\overline{m}}} \qquad (4)
$$

The $\rho_{id}(\overline{m})$ curves for different dimensions of the number set are shown in Fig. 4. With increasing dimension, the curves converge to the $\rho_c(\overline{m})$ curve. For $M = 1.5$ the delays have been determined by stochastic simulations. Around this point, identifier splittings with all considered number sets have their maximum throughput. Fig. 5 contains the complementary distribution function of delay measured in number of frames till successful transmission. It can be seen that with employing identifier splitting, the delay of transmitting dynamic parameters can be bounded to a maximum value without increasing the signaling overhead. This leads to a dramatical improvement of the overall systems performance of the DSA++ protocol.

In the DSA++ protocol, only stations in contention mode are allowed to transmit in random access slots. Thus, only a subset of the number set will be a candidate for the start set of a collision resolution phase. Furthermore, the assumption made for

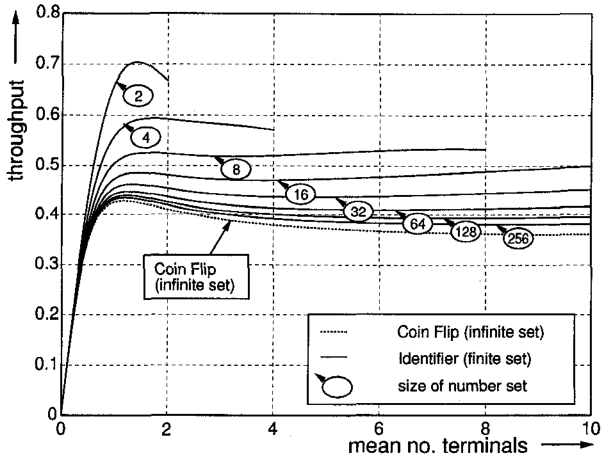

Fig. 4: Throughput rho dependent on mean number of terminals $\overline{m}$ for binary coin flipping and identifier splitting

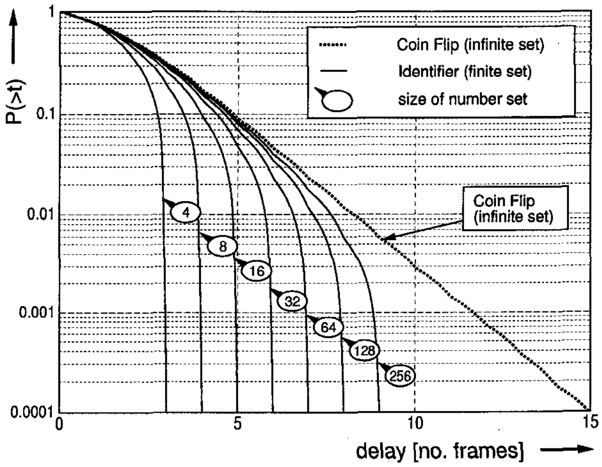

Fig. 5: Complementary distribution function of delay for $\overline{m} = 1.5$

the analysis of an equally distributed arrival rate of random access jobs in terminals is not correct because of the different behaviours of the inter-arrival rates of virtual channels. This will result in a smaller performance of the identifier splitting than determined by analysis. But this effect can partially be compensated, since the base station has knowledge about all stations in contention mode and can estimate the probability of a new arrival of an ATM cell in a specific terminal by taking into account the behaviour of the arrival process of the belonging virtual channels [5]. The base station can dynamically adjust the boundaries for splitting in order to get equal sized subsets after splitting. A drawback of this method is the additional overhead for signaling the splitting boundaries to the terminals. But, if shortened frames are used during collision resolution in order to reach shorter delays, enough capacity is available in the downlink signaling bursts.

V. Combination of Polling and Random Access

The performance of the previous described schemes for transmitting dynamic parameters strongly depends on the scenario. If only a small number of terminals is competing for the channel, polling will have a good performance. But in a terminal room scenario with 20 to 30 active terminals polling will result in a large overhead. Thus, a combination of polling with random access seams to be reasonable. Polling should only be used, if the probability of success is higher than the throughput of random access. An appropriate algorithm has been introduced in [5].

VI. Simulation results

The performance of the three considered schemes for transmission of capacity request messages has been determined by computer simulations. The simulation model is according to [5]. The parameters of the simulation scenario are summarized in Table 1. We considered an TDD air interface (frame length = 30 slots) with an overall data rate according to 50.000 ATM cells per second (uplink and downlink together) resulting in a slot length for transmitting an ATM cell $\tau_{slot} = 20\mu s$. Since the length of a short slot τ_{short_slot} has a considerably influence on system performance, we parameterized it by $r_{short\_slot} = \frac{\tau_{slot}}{\tau_{short\_slot}}$. In Fig. 1 the mean delay of ATM cells from different service classes is shown for the contention free MAC scheme, when using different lengths of short slots.

Table 1: Parameters of simulation scenario

service	ATM class	λ p. VC	#WT	load	$\tau_{d\,max}$	$\frac{\tau_{d\,max}}{\tau_{slot}}$
voice	CBR	64 Kbps	4	1%	2 ms	100
video	VBR	2 Mbps	2	22%	20 ms	1000
data	ABR	0.9 Mbps	10	50%	∞	∞

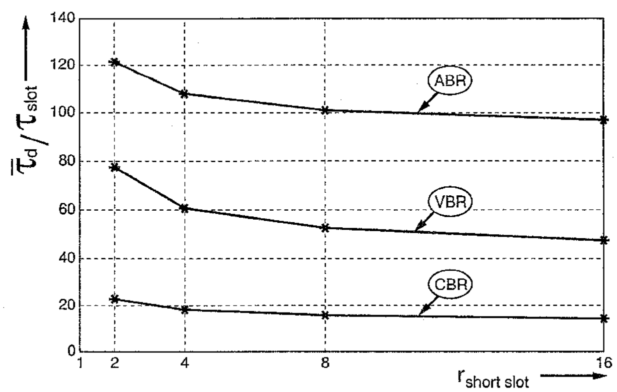

Fig. 6: Mean delays of ATM cells for contention free MAC scheme with different lengths of short slots $\tau_{short\_slot} = \frac{\tau_{slot}}{r_{short\_slot}}$

The comparison of the MAC schemes has been performed with 4 short slots per ATM cell slot. This seems to be a realistic value taking into account the overhead for training sequence, if an equalizer is employed, or for synchronization symbols, if OFDM is used.

The complementary distribution functions of delays from CBR are shown in Fig. 7. It can be seen, that with polling a maximum delay in the order of one frame length can be guaranteed. The maximum delay of identifier splitting is much higher. But the higher signaling overhead of the polling scheme results in longer mean delays for ABR cells in comparison with using identifier splitting ($\overline{\tau_d}$ of ABR cells for different MAC schemes: polling 108, random access 77, both 85)

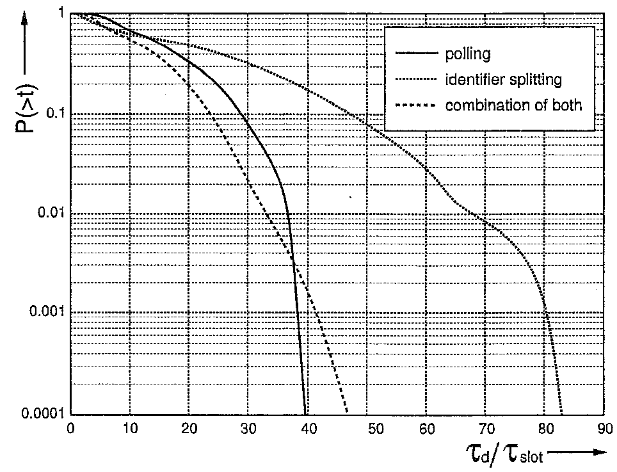

Fig. 7: Complementary distribution function of delays from CBR cells when employing different MAC schemes

The simulation results show, that even in a scenario with a high number of stations polling has a much better performance than simple random access, but that with choosing automatically between polling and random access, the advantages of both schemes can be combined.

VII. References

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

[2] D. Petras, A. Hettich, and A. Krämling, "Performance Evaluation of a Logical Link Control Protocol for an ATM air interface," in *PIMRC'96*, (Taipei, Taiwan), Oct. 1996.

[3] 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.

[4] 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.

[5] D. Petras and A. Krämling, "MAC protocol with polling and fast collision resolution for an ATM air interface," in *IEEE ATM Workshop*, (San Francisco, CA), Aug. 1996.

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

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