Contribution to the Conference on the Performance of Distributed Systems and Integrated Communication Networks, Kyoto, Japan, Sept 1991

Performance Analysis of the CRMA-Protocol in High-Speed Networks

Phuoc Tran-Gia and **Rainer Dittmann**

Institute of Computer Science, University of Würzburg, Am Hubland, D-8700 Würzburg, Germany Telephone: x-49-931-8885509, Telefax: x-49-931-8884601, email: trangia@informatik.uni-wuerzburg.dbp.de

Abstract

In this paper a performance analysis of the CRMA medium access protocol is presented. The CRMA (Cyclic Reservation Multiple Access) scheme is proposed as an access mechanism for high-speed LANs and MANs. An approximate computational method is derived to obtain the distribution functions of performance measures of interest like the medium access delay and the packet transfer time. The analysis is done using a decomposition approach for the access delay in conjunction with a G/G/1 queue with control-feedback and a M/G/1 queue with server vacation, observed in discrete-time domain. The reservation-cancelation backpressure mechanism is also taken into account in the model. Numerical results are obtained to investigate the efficiency of the backpressure scheme and the scaling issues of the interreserve interval under various load conditions and system configurations. Furthermore, results addressing performance aspects like the fairness issues, the jitter of maximum access delay and the system behavior under station-wise saturated conditions are also discussed.

1. THE CRMA-PROTOCOL

The Cyclic-Reservation Multiple-Access (CRMA) protocol has been proposed recently as access scheme for high-speed LANs and MANs, especially in the network capacity region beyond 1 Gbit/sec. The proposed protocol can be used in unidirectional folded-bus or dual-bus topologies.

Detail descriptions of the CRMA access scheme with various bus structures can be found in [1] and [2]. From performance analysis point of view, there is a few studies dealing with modelling aspects of the CRMA protocol. In [3] analytic results for the mean values of the access delay are discussed, comparing different reservation mechanisms. A simulation study of the CRMA protocol is given in [4]. Simulation results are also obtained in [5] to compare the end-to-end delay performance of CRMA with a modified version of the DQDB access scheme (DQDB: distributed queue dual bus) called DQMA (DQMA: distributed queue multiple-access).

To consider the efficiency of overload control mechanisms like the backpressure scheme in CRMA, it is necessary to know the entire state distributions as well as delay and transfer time distribution functions, rather than only the mean values and some higher moments. Thus, the major aim of this paper is to derive an approximate computational method to obtain distribution functions of performance measures of interest. By doing this, questions concerning the fairness issues, the jitter of maximum access delay or the necessity of the backpressure mechanism for a given system and traffic configuration can be quantitatively answered.

As the details of the CRMA scheme can be found in [1] and [2], we will summarize below only those CRMA properties and features, which are relevant in the system modelling context. Since the basic operations are quite analogous for both, unidirectional folded-bus and dual-bus structures, we restrict the following description on the folded-bus configuration, as depicted in Fig. 1.

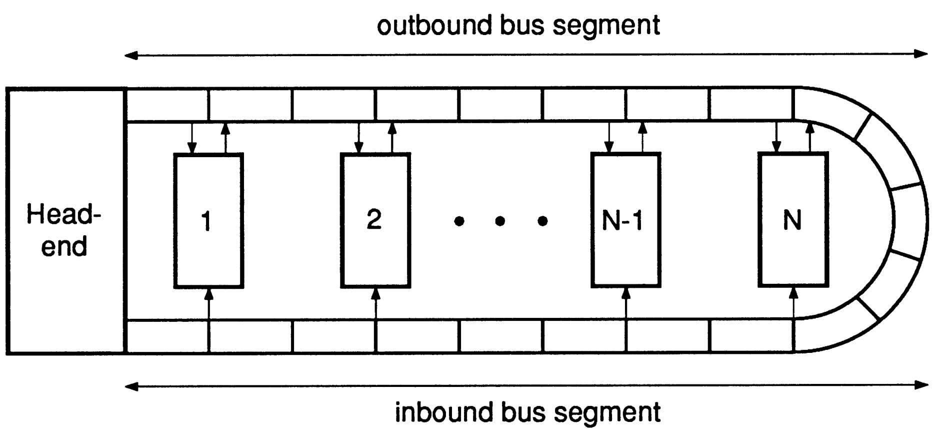

Figure 1: *A CRMA system with folded bus*

Physically, the headend is a part of node 1 and controls the basic access mechanism (cyclic reservation) and the reservation cancelation (backpressure treatment). As illustrated in Fig. 1, each station is connected to the outbound bus segment, where packets are transmitted, and the inbound bus segment, where they are received. The system operates slot-wise; a slot consists of an access control field (ACF) and a segment field (SF) for payload data. The ACF consists of a busy/free-bit (B/F) to indicate if SF is empty and an access command with its arguments. Examples for the access command format can be found in [1, 2].

In the following we look at a set of basic commands which are used to control bus access:
i) the *reserve* and *start* commands for the basic access mechanism and
ii) the *confirm* and *reject* commands to provide the backpressure mechanism. In the basic access mechanism the headend sends *reserve* commands in a periodic manner, which have a cycle number and a cycle length as arguments. For the basic access case, the interreserve interval Q is set to a constant value ($Q = \Theta$ slots). A node reserve slots

by increasing the cycle length by the number of required slots on the *reserve* command (R) flowing by (cf. Fig.2 a,b and c). For each reservation, a station is allowed to reserve slots for only one packet. A packet consists of X slots (X will be assumed as a random variable), which can have a maximal size of X_{MAX} slots. Furthermore the station stores this number as the reserve length in accordance to the cycle number in a local reservation queue. The cycle length can only be modified on the outbound bus segment. When the *reserve* returns to the headend, it is ranged in a global reservation queue. This queue is maintained by the headend station and operates in a FIFO order.

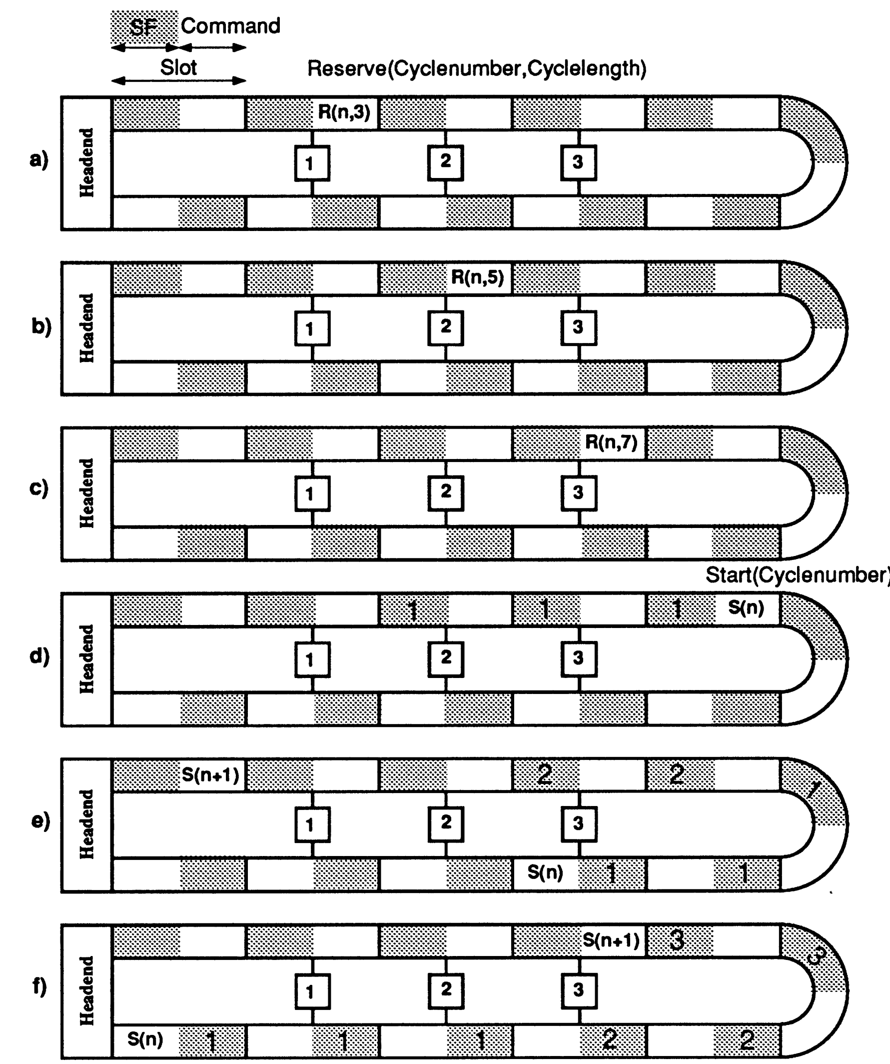

Figure 2: *Example of CRMA basic access*

For each reserve cycle the headend sends a *start* command (S) consisting of the cycle number followed by as many empty slots as indicated by the cycle length. After a node has received a *start*, it searches its local reservation queue for a entry with the current cycle number. If there is one, the station uses as many empty slots as indicated by reserve length (cf. Fig.2 d,e and f).

In the basic access mechanism the *reserve* commands are sent periodically. The fixed interreserve interval $Q = \Theta$ should be shorter than the maximal packet length X_{MAX} to make sure that full bus utilization can be achieved by a single node. In the case that several nodes reserve in consecutive cycles packets with maximal length, the cycle length is temporarily much larger than the interreserve interval Q. This situation can occur when a few stations start e.g. a file transfer phase. Thus the access delay increases accordingly due to the overload condition of the global reservation queue.

A reservation-cancelation backpressure mechanism is introduced to minimize the worst-case access delay by varying the length of the interreserve interval according to the traffic demands. In other words, the backpressure mechanism prolongs the interreserve intervals, if a headend queue overload condition is detected. As overload indicator, the state U of the headend queue is used, i.e. the number of reserved slots waiting in the headend queue to be issued. For that purpose the headend checks after processing a *reserve* the overload indicator U. If U exceeds an overload threshold L, the backpressure scheme becomes active. As discussed in [1] a natural choice of the threshold L is the bus turnaround time τ. In the backpressure case, no *reserve* is issued until the number of reserved slots is again less than L. During this phase, all *reserve* commands already issued but still on the bus are cancelled.

The reservation-cancelation backpressure mechanism requires two additional access commands: the *confirm* command having the cycle number as argument and the *reject* command without any argument. In the normal case, i.e. the headend is operating with the basic access mechanism, the headend issues a *confirm* to confirm the reservation after receiving a returning *reserve*, using the same cycle number as argument. If after the reception of a *reserve* the number of waiting slots in the headend queue exceeds the threshold L, the headend will work in reservation-cancelation mode. The headend issues now a *reject*. Each station receiving a *reject* has to remove all unconfirmed reservations from its local reservation queue. The headend will return to the basic access mode when the headend reservation queue size drops below the threshold L. Obviously the first regular *reserve* will return at the headend when the headend queue becomes empty, if the threshold is chosen as the bus latency τ,

In the next section, the delay analysis for the general case, i.e. under consideration of the backpressure mechanism, will be described, following by the simpler analysis for the case without backpressure. Therefor we use time discrete methods, because we are interested in the whole distribution functions of the delays. Subsequently, numerical results will be presented for different traffic and system configurations. Finally, conclusions and a outlook will be given in the last section.

2. PERFORMANCE MODEL AND DELAY ANALYSIS

2.1. Delays and decomposition

The major aim of this paper is to compute the distribution functions of the medium access delay and the transfer time of data packets in a CRMA network. In fact it is important to know mean values of these measures. However, to investigate fairness issues of the protocol and extreme values of the access delay and its jitter, the whole distribution function has to be considered. For these purposes, we approach the problem with an analytical treatment, since simulations for the range of probability distribution functions required here might be intractable or lead to excessive computation times.

We observe a test packet to be transmitted from a sending station i to a receiving station j. The breakdown of delays and transfer time into subintervals is illustrated in Fig. 3, where the following time instants are marked:

- (1) test packet generated in station i, coming e.g. from higher protocol layer
- (2) station i processes reservation for the test packet
- (3) the reservation for the test packet arrives at the headend station
- (4) the cycle containing the test packet is processed in the headend station and the first slot of the cycle is issued
- (5/5b) first/last slot reserved for the test packet is issued by the headend station
- (6/6b) first/last slot of the test packet is accessed by the station i for test packet transmission
- (7/7b) first/last slot of the test packet arrives at the receiving station j

We denote the time duration between two marked time epochs (x) and (y) with the random variable (r.v.) T_{xy}. The time interval T_{12} is the prereservation delay, i.e. the waiting time of a packet in the sending station until its reservation joins a *reserve* passing by. T_{23} and T_{56} represent the propagation delays and form together the turnaround time $\tau = T_{23} + T_{56}$ along the bus system. Clearly, the duration τ depends on the bus configuration and the CRMA operation mode under consideration. For the folded bus configuration, τ is just the bus latency, i.e. the propagation delay along the entire length of the bus. In the case of dual-bus structure two operation modes are distinguished: the outbound-reservation mode, where τ is identical for all stations, and the inbound-reservation mode, where τ is dependent on the position of the station on the bus (cf. [1], [2]). The propagation time T_{67} can be determined according to the position of sending and receiving stations i and j as well as to the bus operation mode.

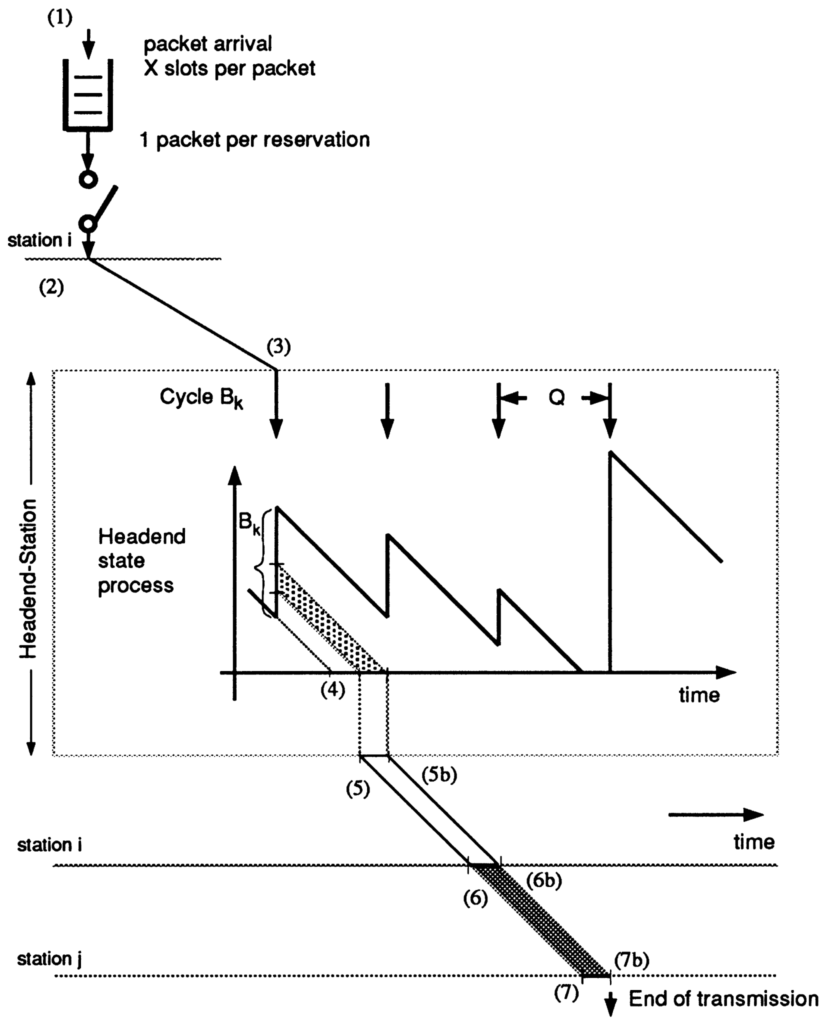

Figure 3: *Components of the medium access delay*

The headend waiting time T_{34} stands for the interval between the arrival instant of the *reserve* containing the test packet and the time instant, at which the first packet of the cycle containing the test packet is processed. This time instant corresponds to the initialisation of the *start*. Depending on the number i of the station and the number of packets reserved by the stations (1, ..., i - 1) the intra-cycle delay T_{45} can be evaluated.

The medium access delay results finally in:

$$T_{ACC} = T_{16} = T_{12} + T_{23} + \cdots + T_{56} \ , \tag{1}$$

i.e. the time interval from the packet generation epoch until the time instant, at which the packet can be transmitted on the medium. The transfer time is accordingly:

$$T_{TRF} = T_{ACC} + T_{67b} \ . \qquad (2)$$

For comparison purposes in later sections of this paper we denote further the waiting part of the medium access delay as

$$T^*_{ACC} = T_{12} + T_{34} + T_{45} \ . \qquad (3)$$

This measure represents the medium access delay excluding the bus latency or propagation time and consists only of queueing process dependent components. This waiting part of the medium access delay T^*_{ACC} is therefore appropriate for comparisons of CRMA performance with different parameters, where the influence of the bus operation mode should be intentionally ignored.

From modelling viewpoint two model levels are considered to investigate the two major delay measures: the prereservation delay T_{12} and the headend waiting time T_{34}. The model for prereservation delay still takes packets as information units while in the headend model the state process has to be described on slot level. This distinction will lead to a decomposition approach in conjunction with an independence approximation as discussed in the following.

In this paper we shall consider two different types of attached stations: i) stations under normal load conditions (random data traffic) and ii) saturated stations (e.g. stations during a file transfer phase, which can be in a quasi-stationary state). Under normal traffic conditions, the packet arrival process in a station i is assumed to be Poisson with rate λ_i. The packet size X can be arbitrarily distributed with mean EX and the maximal packet length is X_{MAX}.

2.2. Headend-station state analysis

The state process development in the headend station is illustrated in Fig. 4, where the general case including the backpressure mechanism is considered. The underlying queueing system is a G/G/1 model with control-feedback which operates in discrete-time domain (cf. [6], [7]). Due to the presence of the backpressure mechanism, the arrival process depends on the actual state of the system. This will be described in more detail below.

We use the following discrete-time random variables in the subsequent analysis, with the slot length Δt as time discretisation

- $U(\mathrm{t})$ — time-dependent headend state, i.e. the total number of slots pending at time t
- B — length of the current reservation cycle (in slots)
- U_n^- — headend state observed immediately prior to the arrival epoch of reserve cycle n, distribution $u_n^-(i)$

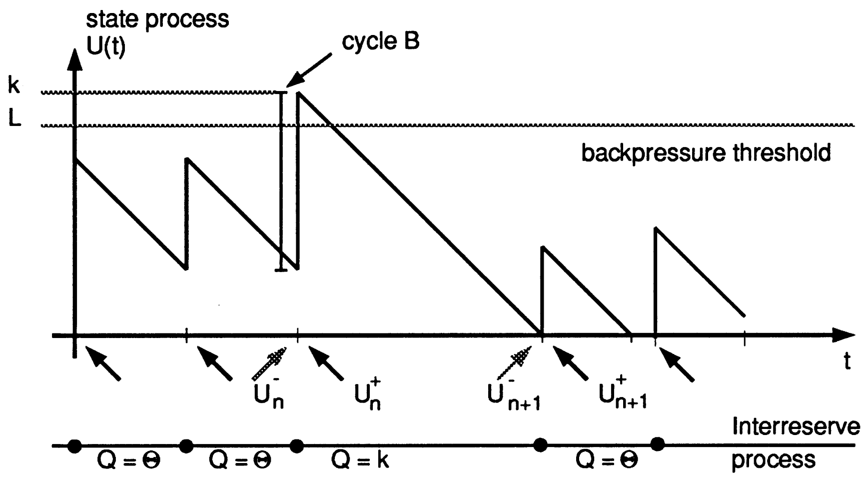

Figure 4: *Headend state process and interreserve interval*

U_n^+ headend state observed immediately after the arrival epoch of reserve cycle n, distribution $u_n^+(i)$

Q interreserve interval, distribution $q(i)$. Without the backpressure mechanism, Q is set to be Θ slots.

The backpressure mechanism with threshold L is considered in the model as follows:

i) normal case: $U_n^+ < L$
The next *reserve* will arrive after Θ slots, i.e. the interreserve interval is $Q = \Theta$. The next cycle length is said to be of type B_Θ.

ii) backpressure case: $U_n^+ = k \geq L$
According to the backpressure mechanism, the next interreserve interval is of length $Q = k$ and the next cycle length is of type B_k

To calculate the headend state probability U_n^+ we consider again the two cases as above, taking into account the different interreserve interval and cycle types

i) normal case : $U_n^+ < L$

$$U_{n+1}^- = \max(0, U_n^+ - \Theta) \qquad \text{and} \qquad U_{n+1}^+ = U_{n+1}^- + B_\Theta \tag{4}$$

ii) backpressure case : $U_n^+ = k \geq L$

$$U_{n+1}^- = 0 \qquad \text{and} \qquad U_{n+1}^+ = B_k \tag{5}$$

To express eqns. (4) and (5) in terms of probability distributions we introduce the following auxiliary functions $u_n^{(1)}(i)$ and $u_n^{(2)}(i)$, corresponding to a separation of $u_n^+(i)$ due to the backpressure and normal cases:

$$u_n^{(1)}(i) = \begin{cases} 0 & i < L \\ u_n^+(i) & i \geq L \end{cases} \quad \text{and} \quad u_n^{(2)}(i) = \begin{cases} u_n^+(i) & i < L \\ 0 & i \geq L \end{cases} \tag{6}$$

The functional relationship between the distribution of the r.v. U_n^+, U_{n+1}^- and U_{n+1}^+ can be derived using the conditional probabilities:

i) $U_n^+ < L$

$$\begin{aligned} Pr\{U_{n+1}^+ = i \mid U_n^+ < L\} &= \frac{1}{\sum_{j=0}^{L-1} u_n^{(2)}(j)} \pi_0(u_n^{(2)}(i) * \delta(l+\Theta)) * b_\Theta(i) \\ &= \frac{1}{\sum_{j=0}^{L-1} u_n^{(2)}(j)} \pi_0(u_n^{(2)}(l+\Theta)) * b_\Theta(i) \end{aligned} \tag{7}$$

where the operator π_0 is defined as

$$v(i) = \pi_m(u(\cdot)) = \begin{cases} \sum_{\nu=-\infty}^{0} u(\nu) & i = 0 \\ u(i) & i > 0 \, , \end{cases} \tag{8}$$

the $*$-symbol denotes the discrete convolution operation and

$$\delta(m-n) = \begin{cases} 1 & m = n \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

ii) $U_n^+ = k \geq L$

$$Pr\{U_{n+1}^+ = i \mid U_n^+ = k \geq L\} = b_k(i) \tag{10}$$

Combining the two cases i) and ii) above we get

$$u_{n+1}^+(i) = \pi_0(u_n^{(2)}(l+\Theta)) * b_\Theta(i) + \sum_{j=L}^{\infty} u_n^{(1)}(j) \cdot b_j(i) \tag{11}$$

Eqn.(11) leads to an iterative algorithm to calculate the equilibrium distribution (mass function) $u^+(i) = \lim_{n\to\infty} u_n^+(i)$ of the headend state process observed immediately after the arrival of the *reserve*. We use standard procedures of the Fast Fourier Transform (FFT) to compute the convolution efficiently.

The service time B_Θ or B_k of the G/G/1 queueing system with control-feedback as presented is the length of the current reservation cycle. Per *reserve*, a station can make reservation for one packet of size X. The cycle length B_k depends directly on the length of the interreserve interval $Q = k$. A cycle length which corresponds to an interreserve interval of length $Q = k$ will be referred to as a cycle of type k.

Using an independent assumption the cycle length can be given as follows:

$$B_k = \sum_{j=1}^{N} B_k^{(j)} = \sum_{j=1}^{N} (1 - p_{o,k}^{(j)}) \cdot X \tag{12}$$

where $B_k^{(j)}$ represents the contribution of station j to the cycle and $p_{o,k}^{(j)}$ denotes the probability for station j having no packet to reserve in a cycle of type k:

$$p_{o,k}^{(j)} = 1 - \lambda_j \cdot k \cdot \Delta t \qquad \text{for} \qquad \lambda_j \cdot k \cdot \Delta t < 1 \tag{13}$$

where Δt denotes the slot duration. We obtain:

$$b_k^{(j)}(i) = p_{o,k}^{(j)} \cdot \delta(i) + (1 - p_{o,k}^{(j)}) \cdot x(i). \tag{14}$$

and finally the distribution of a cycle of type k

$$b_k(i) = b_k^{(1)}(i) * b_k^{(2)}(i) * \cdots * b_k^{(N)}(i) \tag{15}$$

where N is the number of active stations.

2.3. Interreserve interval and prereservation delay

The distribution of the interreserve interval Q can be calculated out of the headend state probabilities $u^+(i)$ using the following equation:

$$q(k) \;=\; \begin{cases} u^+(k) & k \geq L \\ \sum_{i=0}^{L-1} u^+(i) & k = \Theta \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

Since we do not consider the propagation delay, we issues only the *reserve* commands, which will be confirmed. Thus we need not to deal with a *reject* or canceled *reserve.*

From the viewpoint of a station, the interreserve time Q is the time interval, during which one packet reservation can be made. We devote attention to the prereservation delay, which is defined as the delay of a packet in the sending station from generating instant until joining a *reserve* (T_{12}, see Fig. 3).

The queueing process describing the prereservation delay can be modelled by means of an M/G/1 queue with server vacation, where both the service time and the vacation time are identical with the interreserve time Q. The server with service time Q processes the local queue until it gets empty. When the server sees an empty queue at the end of a service time, it will take a vacation period of duration Q. The Laplace-Stieltjes-Transform (LST) of the prereservation delay T_{12} can thus be given as (cf. [8], [9])

$$T_{12}(s) = W_{M/G/1}(s) \cdot Q^r(s) \tag{17}$$

where $W_{M/G/1}(s)$ and $Q^r(s)$ denote the LST of the waiting time of the equivalent M/G/1 system without server vacation and the recurrence time of the interreserve interval, respectively. Since this implies an independency assumption, the following equations are approximations in the case of backpressure. Still the results are quite accurate as simulations show (cf. fig. 7).

Finally the prereservation delay in the station j is

$$T_{12}(s) = \frac{1 - \lambda_j EQ}{EQ} \cdot \frac{1 - Q(s)}{s - \lambda_j(1 - Q(s))}. \tag{18}$$

Some remarks should be made here concerning the numerical computation. To bypass the Laplace transform inversion, we have used the cepstrum-algorithm to analyse the discrete time M/G/1 system (cf. [10] , [6]). This results in the waiting time distribution $w_{M/G/1}(i)$. Further, the (discrete) recurrence time distribution $q^r(k)$ is computed directly out of $q(k)$ using the relationship:

$$\begin{aligned} q^r(0) &= \frac{1}{EQ}(1 - q(0)) \\ q^r(k) &= q^r(k-1) - \frac{1}{EQ} \cdot q(k), k = 1, 2, ... \end{aligned} \tag{19}$$

Finally, the prereservation delay $T_{12}(s)$, which is approximately represented by the discrete-time distribution $t_{12}(i)$, is determined by the convolution of $w_{M/G/1}(i)$ and $q^r(i)$.

2.4. Waiting time in the headend station

Recall that using the iteration according to eqn. (11) we obtained the equilibrium distribution of the headend state immediately after the arrival of a *reserve* with the analogous auxiliary function (cf. eqn. (6))

$$u_n^{(1)}(i) = \begin{cases} 0 & i < L \\ u_n^+(i) & i \geq L \end{cases} \quad \text{and} \quad u_n^{(2)}(i) = \begin{cases} u_n^+(i) & i < L \\ 0 & i \geq L \end{cases} . \tag{20}$$

Note that the probability that a backpressure case occurs is

$$P_B = \sum_{j=L}^{\infty} u^+(i) = \sum_{j=L}^{\infty} u^{(1)}(i) . \tag{21}$$

By means of the distribution $u^+(i)$ and the interreserve time distribution $q(i)$, the distribution of the headend waiting time $t_{34}(i) = Pr\{T_{34} = i \cdot \Delta t\}$ can be derived. In the following analysis we do not consider the *confirm* commands to get more simple equations. This approximation leads to difference of a single slot, if the headend is empty, when a *reserve* arrives.

Again, we observe a test packet generated in station i. The probability of the test packet to be imbedded in an interreserve interval of length $Q = k$ is

$$q^*(k) = Pr\{ \textit{ test packet in } Q = k\} = \frac{kq(k)}{EQ}. \tag{22}$$

Conditioning on a test packet being imbedded in a cycle with $Q = k$, the headend delay T_{34} can be given as follows

i) regular interreserve interval $Q = \Theta$

$$Pr\{T_{34} = i \cdot \Delta t \,|\, \textit{test packet in } Q = \Theta\} = \frac{1}{1 - P_B} \pi_0(u^{(2)}(l + \Theta)) \tag{23}$$

ii) interreserve interval with backpressure $Q = k \geq L$

$$Pr\{T_{34} = i \cdot \Delta t \,|\, \textit{test packet in } Q = k \geq L\} = \delta(i) = \begin{cases} 1 & \text{for } i = 0 \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

Finally, the headend delay distribution is:

$$t_{34}(i) = \sum_{k=0}^{\infty} Pr\{T_{34} = i \cdot \Delta t \mid \textit{test packet in } Q = k\} \cdot q^*(k) \tag{25}$$

or after some algebraic manipulations

$$t_{34}(i) = \pi_0(u^{(2)}(i + \Theta)) \frac{\Theta}{EQ} + \delta(i)(1 - \frac{\Theta(1 - P_B)}{EQ}). \tag{26}$$

2.5. The special case without backpressure

For the current choice of the backpressure threshold $L = \tau$, where τ is the bus latency, the backpressure mechanism plays only an important role for special cases, e.g. systems under heavy loads or systems with some stations under saturated load conditions. For normal traffic intensities the probability for the headend queue to exceed the threshold $L = \tau$ is very small, especially for long and high-speed bus systems such as in metropolitan area networks under consideration (see results in next section). For those cases, where the probability for the backpressure mode to occur is negligible, we can use a simpler analysis which will be briefly outlined in the following (cf. [3]).

The interreserve interval is constant for this case with $Q = \Theta\Delta t$. To calculate the pre-reservation delay T_{12} we simply employ a M/D/1 queue with server vacation, where both the service and vacation times have the constant value Θ slots. A simplified version of the discrete-time model discussed above can be used for computational purposes according to

$$T_{12}(s) = \frac{1-\lambda_j\Theta\Delta t}{\Theta\Delta t} \cdot \frac{1-e^{-s\Theta\Delta t}}{s-\lambda_j(1-e^{-s\Theta\Delta t})} \tag{27}$$

For the direct computation of T_{12} in time domain, we can also use the relationship for the delay distribution function given in [11] :

$$Pr(T_{12} \leq w\Theta\Delta t + t_v) = \frac{1-\lambda_j\Theta\Delta t}{\lambda_j\Theta\Delta t} \cdot \Big(\sum_{i=0}^{w} \frac{(-\lambda_j(i\Theta\Delta t + t_v))^{w-i}}{(w-i)!} e^{\lambda(i\Theta\Delta t + t_v)} - 1\Big) \tag{28}$$

where $\quad w = \lfloor t/\Theta \rfloor \quad$ and $\quad t_v \in [0, \Theta\Delta t)$.

For the basic access scheme considered here, the waiting time T_{34} in the headend station can be computed using a D/G/1 model without control-feedback (cf.[3]), where the interarrival time is identical to the interreserve interval Θ and the service time is the cycle length of type Θ as given in eqn. (15). The numerical computation of the waiting time of this system is done using the algorithm described in [10] and [6] involving the cepstrum transformation.

3. RESULTS

3.1. System parameters and configurations

We consider a network with $N = 16$ stations, which are equidistantly located on a folded bus system. The propagation velocity is assumed to be $200000 km/sec$. The slot length is chosen at 55 Bytes, consisting of 48 Bytes payload segment field, 5 Bytes wide area network header and 2 Bytes CRMA header. We assume further the packet length X to have a truncated Poisson distribution. The distribution is truncated by X_{MAX} and subsequently renormalized. The time axis in the Figures shown below is given in slots or slot transmission time.

The following configurations are taken into account:

a)	bus length :	10km	capacity :	140 Mbps	bus latency :	32 slots
b)	bus length :	10km	capacity :	1.2 Gbps	bus latency :	273 slots
c)	bus length :	100km	capacity :	140 Mbps	bus latency :	318 slots
d)	bus length :	100km	capacity :	1.2 Gbps	bus latency :	2727 slots

3.2. Influence of backpressure mechanism

Basically, the purpose of the reservation-cancelation backpressure mechanism is to protect the headend queue against overload. The main mechanism is to prolong the interreserve interval artificially in case of headend queue overload, i.e. if a threshold L is reached in the headend queue. Since a station is allowed to reserve only one packet per reservation command, this mechanism will reduce the total load offered to the headend under overload conditions. However, since L is chosen to be equal to the bus latency τ, the backpressure mechanism is expected to be active only under very high load levels or in the case where a few stations are sending packets under saturated conditions.

To investigate the influence of the backpressure mechanism and the capability of the system to survive a high load period where a number M out of N stations are in saturation, we consider first the CRMA protocol without backpressure, and try to cope with the question, under which circumstances and by which bus configuration the backpressure mechanism will be activated. We assume that a station in saturated condition has always packets to send, with the packet size X has a Poisson distribution truncated by X_max. For this purpose, we observe the headend state U^+ taken immediately after the arrival epoch of a reservation cycle. Additionally we devote attention to the complementary distribution function

$$Pr\{U^+ > k\} = U^{+,C}(k) = \sum_{i=k+1}^{\infty} u^+(i) \tag{29}$$

and the probability of reaching or exceeding the backpressure threshold:

$$P_B^* = Pr\{U^+ \geq L\} = u^{+,C}(L-1). \tag{30}$$

The complementary probability distribution function of the headend state after receiving a reserve command is depicted in Fig. 5, for different values of the mean packet length. The maximal packet size is chosen at $X_{MAX} = 80$ slots and the interreserve interval is 80. To illustrate the backpressure threshold according to the bus parameters, $L = \tau$ is marked in Fig. 5 for the configurations a), b) and c).

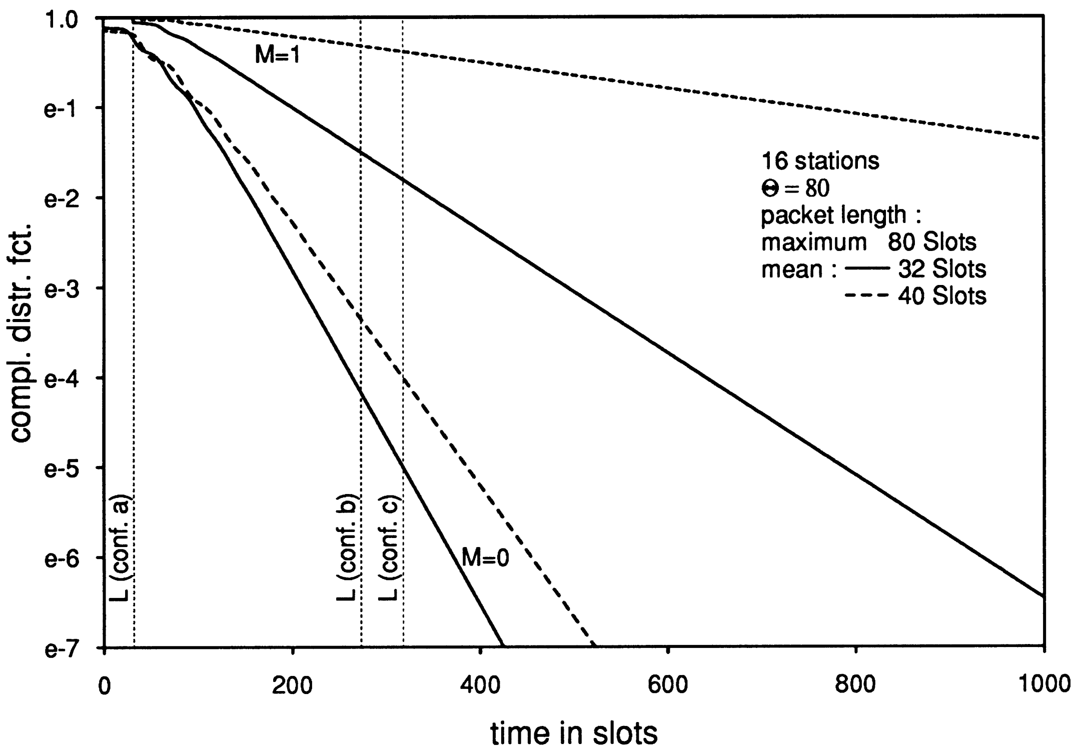

Figure 5: *CRMA without backpressure: probability of exceeding backpressure threshold*

It can clearly be seen in Fig. 5 that for the configuration a) (short bus, lower capacity), the probability of reaching or exceeding the backpressure threshold is high. For configurations b) and c) and the case of $M = 0$, i.e. all stations with symmetrical non-saturated traffic (bus utilization $\rho = 0.5$ for both packet sizes), the probability P_B^* is around 10^{-4}. This probability is higher, as expected, for higher mean packet size, as shown in Fig. 5. For this case, due to small backpressure probabilities, the simpler analysis for the CRMA protocol without backpressure may give already sufficiently accurate approximate results.

Keeping $N - M$ stations on the same traffic level, with one station ($M = 1$) switching over to saturated conditions, Fig. 5 shows a dramatic increase of P_B^*. It should be mentioned here that due the consideration of $M = 1$, where the saturated station submits one packet per reserve command, the bus utilization is $\rho = 0.87$ for $EX = 32$ slots and $\rho = 0.97$ for $EX = 40$ slots. The necessity of the backpressure mechanism is obvious for the case $M = 1$ in this figure.

3.3. CRMA access delay with backpressure mechanism

The next step is to investigate how the backpressure mechanism affects the prereservation delay, the headend delay and their distribution functions. Figs. 6, 7 and 8 show a comparison of CRMA with and without the reservation-cancelation backpressure mechanism, where the bus configuration b) with a bus latency of 273 slots is chosen. A normal load level $\rho = 0.5$ is chosen for Figs. 6 and 7, while an overload case ($\rho = 0.9$) is considered in

Fig. 8. The Figs. 6 and 7 show also some simulation results to proof the accuracy of the analytical results.

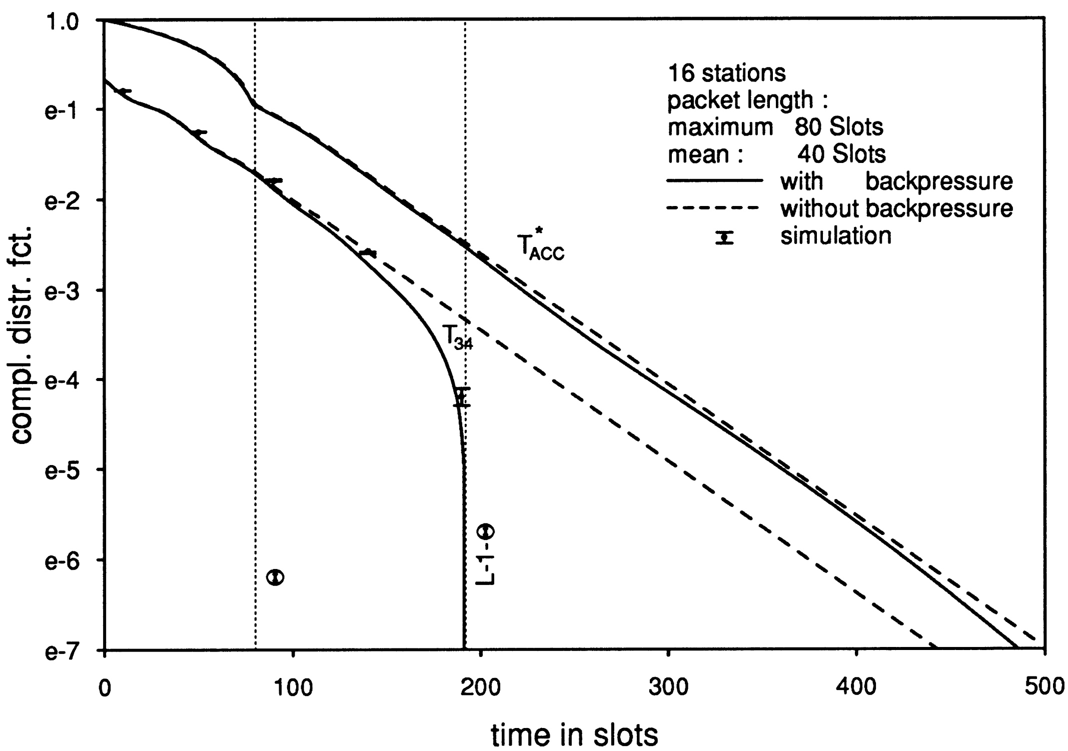

Figure 6: *Comparison of headend delay with and without backpressure*

In Fig. 6 the complementary distribution functions of the headend delay T_{34} and the medium access delay T^*_{ACC}, defined as the medium access delay without bus latency (cf. eqn (3)) are depicted. The load-limiting function of the backpressure scheme can clearly be recognized. The distribution function of the headend delay is bounded by $L - 1 - \Theta$. This value corresponds to the case, where after processing the last reserve command, exactly $L - 1$ slots are in the headend queue.

While the headend delay is shorter with the backpressure mechanism, the prereservation delay T_{12} becomes longer, as shown in Fig. 7. This effect corresponds to the objective of the backpressure mechanism. The headend delay, which is a global delay for all stations, is kept smaller. The influence of local overload situations in a few stations to normal-load stations is reduced by the backpressure mechanism. Due to the enlargement of the interreservation intervals under overload conditions, loads coming from high-load stations are throttled, since each station can reserve at most one packet per interreserve interval.

It should be noted here that the bow-like characteristics of the prereservation delay result from the interreserve recurrence time part of T_{12} as stated in eqn. (17).

It can be seen in Figs. 6 and 7 that, irrespective of the differences of the headend delays and the prereservation delays, the difference between medium access delays T^*_{ACC}

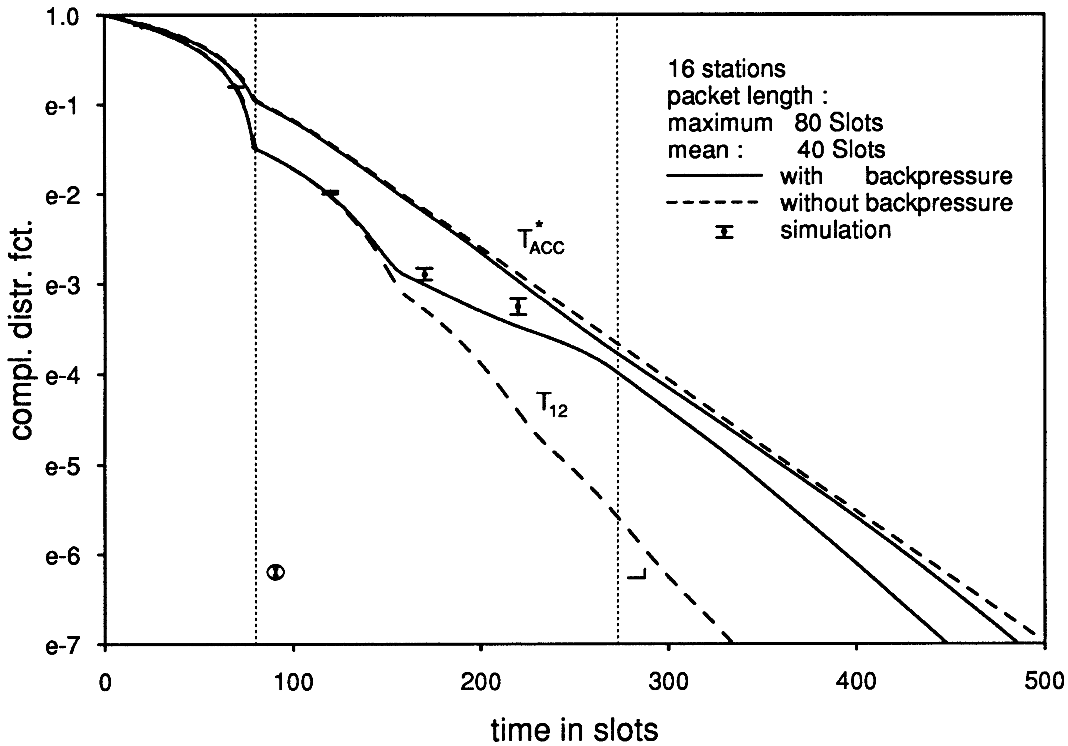

Figure 7: *Prereservation delay with and without backpressure*

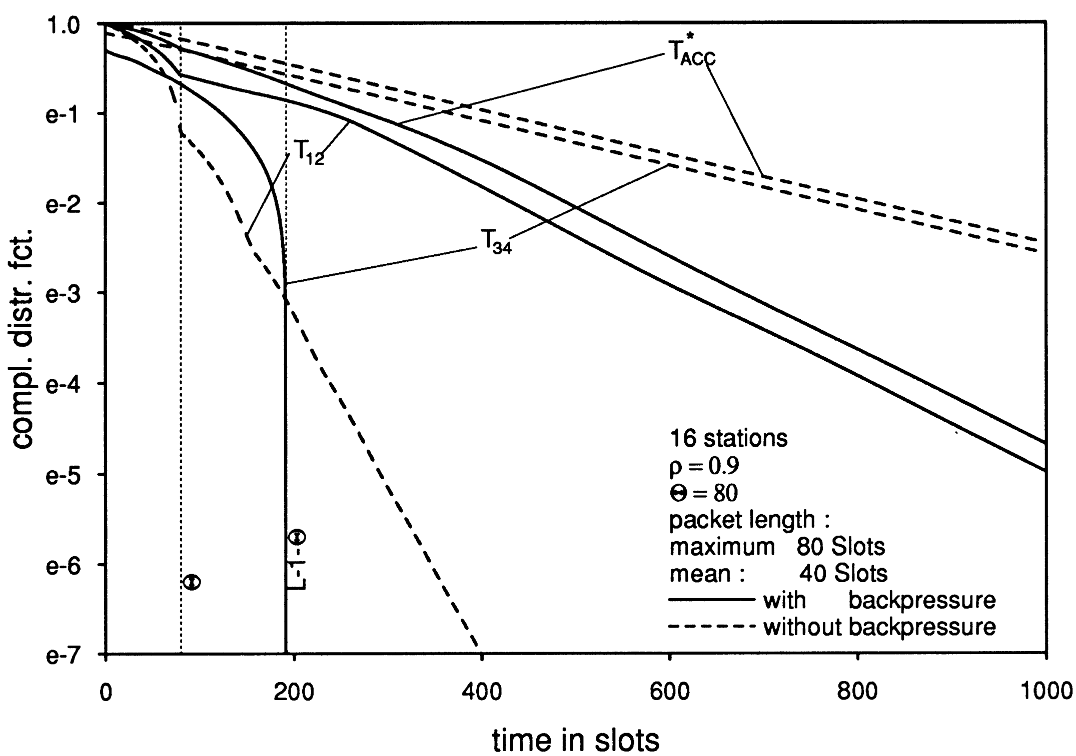

Figure 8: *Overload control capability of the backpressure mechanism*

for CRMA with and without backpressure is small at lower load level ($\rho = 0.5$). This difference is larger for systems under overload conditions ($\rho = 0.9$) as shown in Fig. 8, indicating the efficiency of the backpressure scheme. The access delay T^*_{ACC} is shorter using the backpressure scheme. The backpressure mechanism diminishes obviously the influence of a local overload situation, e.g. when a station is heavily overloaded or in saturation, to the global system performance.

Some hints to scale the basic interreserve interval Θ can be found in Fig. 9, where the mean prereservation time and the mean headend delay are depicted as functions of the basic interreserve interval Θ. A choice of a short Θ will lead to shorter prereservation time and longer headend delay. On the other hand, Θ must be chosen large enough to avoid hogging of reservations. If one station should be able to use the entire bus, a basic interreserve interval of $\Theta = X_{MAX}$ seems to be a good practical choice [1].

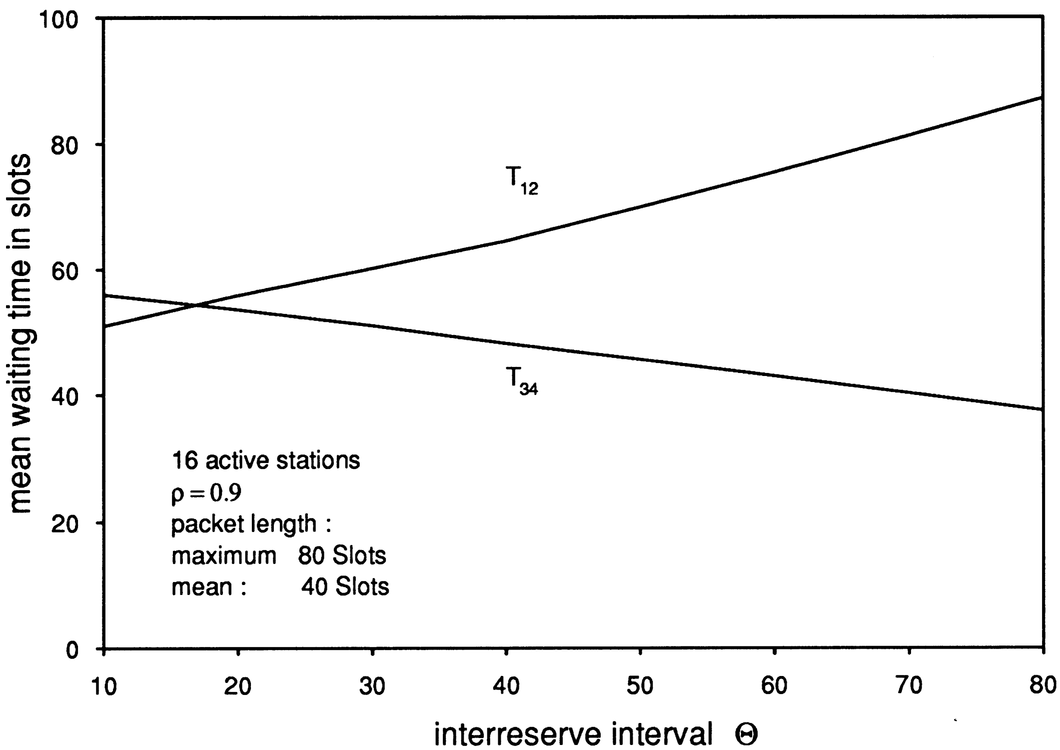

Figure 9: *Influence of interreserve interval on mean delay (with backpressure)*

Fig. 10 shows in summary the prereservation delay, the headend delay and the medium access delay of station 1, 8, and 16, where the complementary probability distribution functions are depicted. The position-dependency of the access delay can be recognized, which is indicated by the differences of the access delays of the considered stations.

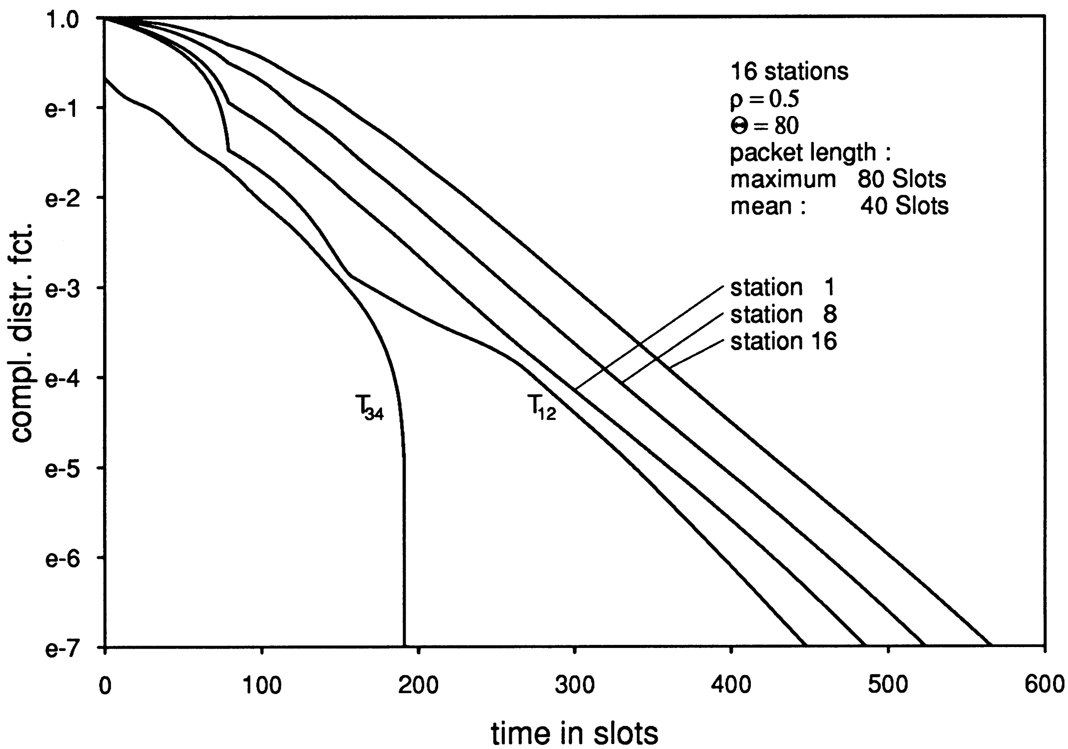

Figure 10: *Delays for several stations*

4. CONCLUSION AND OUTLOOK

An analysis of the CRMA (Cyclic reservation Multiple Access) medium access protocol, proposed for high-speed LANs and MANs, has been presented in this study. The aim is to obtain the distribution functions of the medium access delay and its parts, i.e. the prereservation delay and the waiting time in the headend station, where an approximate computational method is derived. Both versions, the CRMA basic access and the CRMA with reservation-cancelation backpressure mechanism have been considered in the model. The analysis is done using a model decomposition for the medium access delay. Two submodels arose and have been analysed; a G/G/1 queue with state-dependent feedback and a M/G/1 queue with server vacation. The models are observed and the analysis is done in discrete-time domain. Using some numerical results, the necessity and the influence of the backpressure scheme to the access delay have been investigated. The results shown that the backpressure mechanism is seldomly activated in the case of symmetrical non-saturated traffic and moderate load levels. In contrast, the backpressure mechanism is necessary to protect the headend queue in particular and the overall system in general against i) global overload, i.e. in systems under extreme high-load conditions and
ii) local overload, i.e. in systems with saturated-traffic stations. The scaling issues of protocol parameters like the interreserve interval under various load conditions and system configurations have been investigated.

Acknowledgement

This study arose from a research cooperation with IBM Zurich Research Laboratory. The authors would like to thank Mehdi Nassehi and Pitro Zafiropulo for stimulating discussions on CRMA issues and Steve Fuhrmann for valuable hints on queueing systems with server vacations. Interesting discussions with Thomas Stock and programming supports of Manfred Mittler are greatly appreciated.

References

[1] Nassehi M.M., "CRMA: an access scheme for high-speed LANs and MANs", Proc. SUPERCOMM/ICC'90, Atlanta, GA, April 90.

[2] Nassehi M.M., "Cyclic-reservation multiple-access scheme for Gbit/s LANs and MANs based on dual-bus configuration", Proc. EFOC/LAN'90, Munich, June 90.

[3] Boxma O.J., Levy H., Yechiali U., "Cyclic reservation schemes for efficient operation of multiple-queue single-server systems", Preprint 1990.

[4] Marsan M.A., Neri F., Taricco G., "A simulation study of the CRMA protocol", Preprint 1990.

[5] Müller H.R., Nassehi M.M., Wong J.W., Zurfluh E., Bux W., Zafiropulo P., "DQMA and CRMA: New access schemes for Gbit/s LANs and MANs", Proc. IEEE INFOCOM'90, San Francisco, CA, June 90.

[6] Tran-Gia P., "Discrete-time analysis for the interdeparture distribution of GI/G/1 queues", Proc. Semin. Teletraffic Analysis and Comp. Perform. Eval., Amsterdam, The Netherlands, 1986.

[7] Tran-Gia P., "Analysis of a load-driven overload control mechanism in discrete-time domain", Proc. 12th International Teletraffic Congress, Torino 1988.

[8] Servi L.D., "D/G/1 queues with vacations", Operations Research 34(1986) 619-629.

[9] Fuhrmann S.W., "A note on the M/G/1 queue with server vacations", Operations Research 31(1981) 1368-1373.

[10] Ackroyd M.H., "Computing the waiting time distribution for the G/G/1 queue by signal processing methods", IEEE Trans. Commun. COM-28(1980) 52-58.

[11] Langenbach-Belz M., "Sampled queueing systems", Proc. Symp. on Comp. Comm. Networks and Teletraffic, Pol. Inst. of Brooklyn, NewYork 1972, pp 157-176.