CSMA-CD-DR: A NEW MULTI-ACCESS PROTOCOL FOR DISTRIBUTED SYSTEMS

W.M. Kiesel, P.J. Kuehn

University of Siegen, Department of Communications
Hoelderlinstrasse 3
D 5900 Siegen 21, W.-Germany

ABSTRACT

A new CSMA-CD-DR protocol (Carrier-Sensed-Multi-Access with Collision Detection and Deterministic Retransmission) for a distributed system with N stations and one common channel is described. In the silent state, the channel is operated in the contention mode. The channel arbitration after a completed transmission is resolved by staggered delays; these delays are dynamically changed for each station upon the broadcasted acknowledgements. This protocol limits the possibility of collisions strictly. The paper describes the protocol function and the principal software structures of each station by means of a SDL diagram. Extensions are discussed, especially with respect to overload control. The performance of the protocol is analyzed by simulation and analytic modeling. It is shown that the protocol combines the advantages of contention mode and fixed/demand assignment for low and high traffic, respectively. The results indicate that the protocol seems favourable for local networks and systems with distributed control.

1. INTRODUCTION

In distributed systems, the communication channel assignment plays an important role to provide low response times for a large number of competing users at a reasonable efficient utilization of the communications resources. With the development of broadband carriers for data communications, random access techniques have recently become considerable interest forming a powerful alternative to the well-known fixed/demand-assignment techniques [1].

Multi-access contention protocols, as the ALOHA or the basic CSMA (carrier-sensed-multi-access) and their derivatives, were originally developed for digital radio communications. Similar techniques become now important for local computer networks and communication systems with distributed control.

In the basic CSMA protocol, all stations (users, computers, control devices etc.) share one common channel. Contrary to ALOHA-type protocols, a station senses the channel and transmits a message only when the channel has been sensed idle. Nevertheless, collisions may still occur due to the propagation delay between the stations. Collisions may be detected either through a cyclical redundancy check upon reception of a message, or immediately at the collision instant through a bit-by-bit comparison of the transmitted stream and the stream observed on the channel (CSMA-CD: CSMA with collision detection). In either case, a proper schedule has to control the retransmission of the collided messages. Various schemes have been proposed which are based on randomly chosen retransmission delays [1], or fixed deterministic retransmission delays [2,3].

In this paper we propose a deterministic retransmission (DR) scheme which is based on the acknowledgement messages being broadcasted after each successful message transmission: making consequently use of the acknowledgement information, which is also available to all those stations other than the communicating ones, each station changes a deterministic retransmission delay such that no further collisions can occur during a "busy period" and that all stations are treated fairly. The algorithm can easily be changed to static priority schedules or dynamic priority schedules to, e.g., defeat temporary overload. This protocol limits the possibility of collisions strictly to a small time window following the first access attempt at the begin of a busy period of the common channel.

In Section 2 we define the new CSMA-CD-DR protocol in detail and also discuss some of its possible extensions. In Section 3 the performance of this protocol is analyzed by simulation as well as by approximate analytic means. Finally, we give some results of the throughput and delay performance in Section 4.

2. DEFINITION OF THE CSMA-CD-DR PROTOCOL

2.1 Network Configuration

We consider a local network type as shown in Fig. 1. An arbitrary number N of stations are connected to one common channel. Examples of such a structure are ETHERNET [4], SATNET [5] and in-house computer networks [6].

Each station interfaces its connected peripherals with the common channel and exists essentially of a Send-Queue, a Receive-Queue and a control C which performs the functions of channel sensing, sending, collision detection, address decoding, acknowledgement detection and processing as well as delay adjustment according to the following CSMA-CD-DR protocol.

Contribution to the "National Telecommunications Conference" (NTC), Nov. 29 - Dec. 3, 1981, New Orleans, La., USA

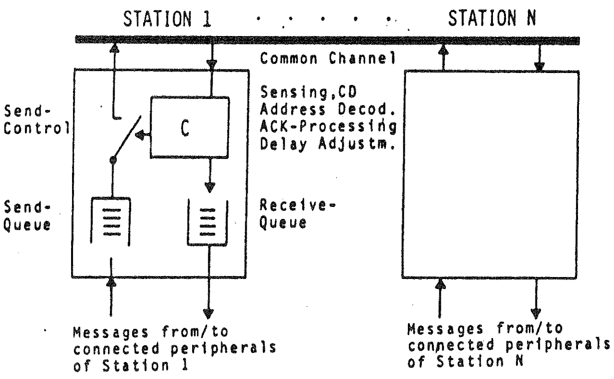

Fig. 1. Network structure for the CSMA-CD-DR-protocol

2.2 The Basic CSMA-CD-DR Protocol

The CSMA-CD-DR protocol is an extension of the CSMA-CD protocol using generally immediate acknowledgeing after a transmission. Contrary to the random transmission delays of the CSMA-CD protocol in case of contention, the CSMA-CD-DR protocol uses deterministic time delays to resolve the priority problem after a successful transmission or a collision. To provide for a fair chance to each of the connected stations, the deterministic time delays are cyclically changed upon reception of the positive acknowledgement signal being broadcasted after each successful transmission.

The channel can essentially be described by three states, see Fig. 2.

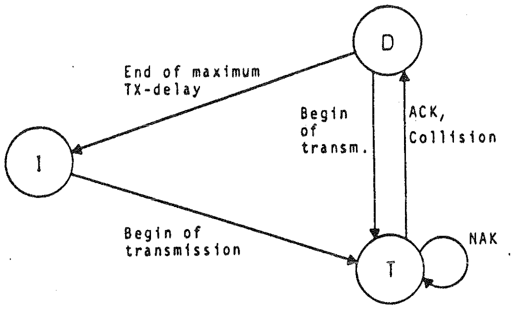

Fig. 2. State transition diagram of the common channel

Symbols: I Idle state (contention mode)
T Transmission state
D Delay state

The protocol being executed by each station can be defined as follows:

(1) Immediate transmission of an arriving message when the channel is sensed in the idle state (transition I→T).

(2) Collision detection during transmission. Under normal operation, collisions can only occur during a small time window following the transition I→T. Upon the occurence of a collision, the channel is considered to switch in the delay state D (transition T→D). All stations proceed now according to their transmission priority in the same manner as after reception of an ACK.

(3) Upon a successful message transmission, the receiving station broadcasts a positive acknowledgement ACK by which the channel is considered to change in the delay state D (transition T→D).

(4) At any time, each station owns an individual deterministic transmission delay time (TX-delay) $i \cdot t_0$, $i = 1,2,\ldots,N$. Upon detection of an ACK, each station updates its current TX-delay through cyclic incrementation by t_0, modulo N (cyclically changing transmission priority).

(5) A station with a message ready for transmission waits at least its current TX-delay after the preceding ACK. If another station transmits prior to this instant, the message waits further on until the next ACK where the same procedure is repeated. In case of no transmission up to this instant, the waiting message is immediately transmitted (transition D→T).

(6) Upon an unsuccessful (but complete) message transmission, the receiving station broadcasts a negative acknowledgement NAK; the sending station then immediately retransmits the message (transition T→T).

(7) If no transmission occurs until $(N+1)\cdot t_0$ after the preceding ACK, the channel is considered to change in the idle state (transition D→I).

The basic software structure of each station can now be described by means of a high level language. An example by the SDL language of CCITT (Specification and Description Language) is given in Fig. 3. The dimensioning of the basic delay time t_0 refers to the maximum two-way propagation delay between the spacially most distantly located stations, extended by a certain time for detection and processing of signals. Both parameters, t_0 and N, form the effective restriction with respect to the range of applicability of the protocol. According to current optical transmission and high-speed circuit technology, the protocol seems applicable to a wide range of local network topologies as well as distributed control or computing applications.

2.3 Extensions of the Protocol

The basic protocol CSMA-CD-DR of Section 2.2 can be extended in several directions :

a) Static priority schedules

In case of staggered transmission delays but without a cyclical change, the stations with the lower transmission delays have effectively a higher priority [2,3]. More general, stations of higher (nonpreemptive) priority can easily be formed through a fixed allocation of the lower transmission delay times and exclusion of those stations from the cyclically changing priority schedule.

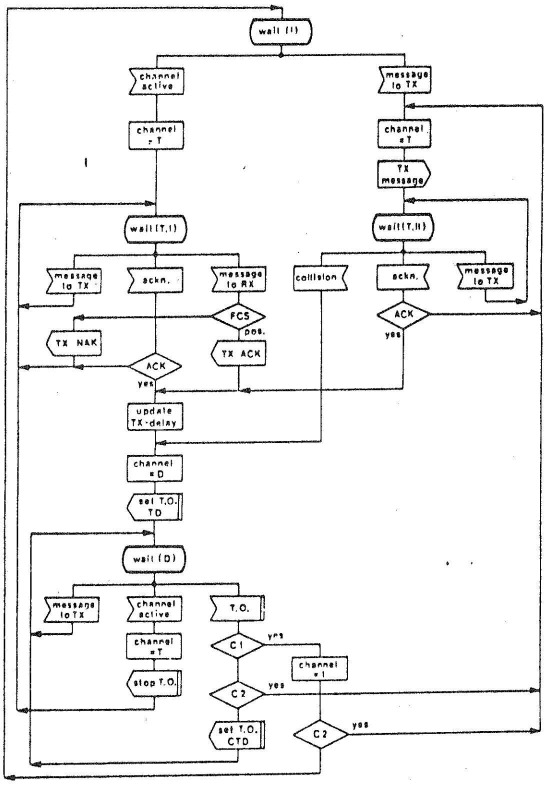

Fig. 3. Software structure in SDL

Symbols:		
	TX/RX	transmit/receive
	ACK/NAK	positive/negative acknowledgement
	FCS	frame check sequence
	T.O.	time out
	I,T,D	channel states (idle, transmission, delay)
	TD	actual TX-delay time
	CTD	complementary TX-delay time, CTD = $(N+1)\cdot t_0 - TD$
	C 1	maximum TX-delay $(N+1)\cdot t_0$
	C 2	message to transmit

Furthermore, priority classes can be formed by grouping of stations into delay classes where the cyclically changing priority schedule is restricted to each class. Again, the number of priority stations limits the applicability due to the ground transmission delays for the low-priority stations.

b) Dynamic priority schedules, overload control

Through a consequent use of the possibilities of a distributed system with broadcasting facilities, the transmission delays could be dynamically changed according to a commonly accepted algorithm taking the actual status (queue lengths, failures etc.) into account.

Such a scheme can easily be implemented using a status bit within the control field of the message headers. Once a status bit is set and broadcasted with a "regular" message transmission by an overloaded station, all other stations disable their transmission delays and the overloaded station proceeds transmitting constantly with the minimum delay t_0 until the status bit is reset upon which the regular schedule is followed again. Such an overload control strategy has the special advantage of operating at the lowest possible overhead, i.e. it defeats the overload with maximum transmission capacity. The status bit could be set, e.g., dependent on a 2-level queue length status.

Furthermore, a similar mechanism could be used to perform maintenance or administrative functions (e.g., extending the number of stations). An arbitrary station can be used as network control center; by setting the status bit, the system is monopolized by that station. All other stations are passive and can be supplied with the new informations.

3. PERFORMANCE OF THE CSMA-CD-DR PROTOCOL

Besides the costs being involved by hardware and software implementation, the usefulness of a protocol for distributed systems depends heavily on its throughput and delay performance. These characteristics are most sensitive to

- resource allocation schemes (operating system)
- overhead
- traffic statistics.

The quantitative qualification is subject to performance analysis.

3.1 Performance Evaluation by Simulation

The basic CSMA-CD-DR protocol has been modeled by a queuing system having one central server (the transmission channel) and N send queues, see Fig. 4.

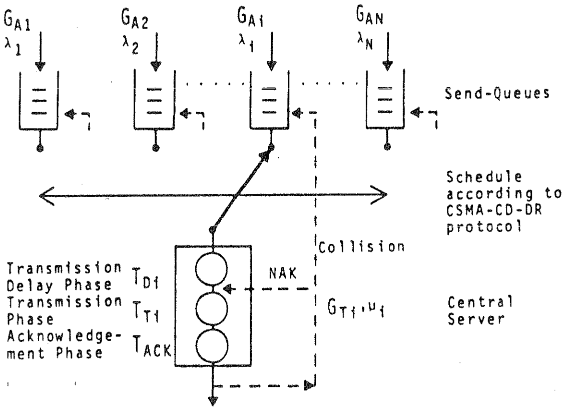

Fig. 4. Queuing model for the performance analysis

Symbols:		
	N	number of connected stations
	G_A	general message arrival process
	G_T	general message transmission process
	λ	arrival rate of messages
	μ	service rate of messages
	T_D	transmission delay time $(t_0,\dots,N\cdot t_0)$
	T_T	transmission time
	T_{ACK}	acknowledgement time
	i	index for station no. i, $i=1,2,\dots,N$

The transmission delay and acknowledgement phases are modeled as "service overhead" according to the actual status of the system. The dashed feedback loops indicate the retransmission after a collision or reception of a NAK. The collision is entirely determined by the arrival process and system state, whereas the negative acknowledgement is generated by a given probability standing for false message transmission.

A large variety of message arrival processes and message length distributions allows the simulation of almost all practical cases. Results of the traffic simulation are given in Section 4.

3.2 Analytic Performance Evaluation

An exact analytic performance analysis seems not feasible due to the complex protocol. We therefore restrict ourselves to approximate analysis methods in the case of Poisson arrivals. The following analysis holds for the practically interesting case when the system overhead is not dominating.

We proceed to model the system in case of symmetrical conditions by means of an M/G/1 queue. Dropping the very small probability for collisions and false transmissions, a resulting service time T per message transmission can be formed by

$$T = T_T + T_D(\underline{X}) + T_{ACK} \ . \tag{1}$$

In eq.(1), the TX-delay time $T_D(\underline{X})$ is a random variable; its actual value depends on the system state $\underline{X}$.

a) Evaluation of the TX-delay time $T_D(\underline{X})$

$T_D(\underline{X})$ can be determined exactly only in case of a detailed state description $\underline{X} = (x_1, x_2, \ldots, x_N; P)$, where x_i defines the number of waiting messages of station i and P indicates that station with the currently minimum TX-delay time. This detailed state description is approximately replaced by a single state variable x, i.e. $\underline{X} = x$, where x defines the number of waiting messages left behind by a departing message. Each of the x single messages is assumed to have a random position relative to that station currently owning the minimum TX-delay time. Thus, the TX-delay T_{Dj} of each of the x waiting messages occurs with probability density function (pdf)

$$f_j(t) = \frac{1}{N} \cdot \sum_{\nu=1}^{N} \delta(t - \nu t_0) \tag{2a}$$

or its probability distribution function (df)

$$F_j(t) = \frac{1}{N} \cdot \sum_{\nu=1}^{N} u(t - \nu t_0), \quad j = 1, 2, \ldots, x, \tag{2b}$$

where $\delta(t)$ and $u(t)$ are the impulse and unit step functions, respectively.

Assuming independence, the actually occuring TX-delay in the state x, $T_D(x)$, is the minimum of all individual TX-delays

$$T_D(x) = \min(T_{Dj}), \quad j = 1, 2, \ldots, x. \tag{3}$$

Thus, we find for its df, pdf, and ordinary moments

$$F(t|x) = 1 - \{1 - F_j(t)\}^x, \tag{4a}$$

$$f(t|x) = \sum_{\nu=1}^{N} \{(1 - \tfrac{\nu-1}{N})^x - (1 - \tfrac{\nu}{N})^x\} \cdot \delta(t - \nu t_0), \tag{4b}$$

$$E[T_D^k(x)] = \sum_{\nu=1}^{N} (\nu t_0)^k \cdot \{(1 - \tfrac{\nu-1}{N})^x - (1 - \tfrac{\nu}{N})^x\}, \quad k = 1, 2, \ldots \tag{4c}$$

b) State-dependent queuing analysis

With $T_D(x)$ according to eqs.(3,4), an M/G/1 analysis can be performed on the basis of an imbedded Markov chain. This analysis will not further be pursued here.

c) State-independent queuing analysis

A simpler analysis can be performed when using average values. As mentioned above, x defines the number of waiting messages left behind by a departing message. According to a general theorem for GI/G/1 queues[7], the distribution of x is identical to the distribution of messages met by an arriving message which is, in case of M/G/1, the same as the state distribution p(x) for an arbitrary time instant. To further simplify the averaging, we use the state distribution for the M/M/1 instead of the M/G/1. We find for the pdf f(t) of the (unconditioned) TX-delay time T_D

$$f(t) = \sum_{x=0}^{\infty} p(x) \cdot f(t|x) = \sum_{x=0}^{\infty} (1-\rho)\rho^x \cdot f(t|x), \tag{5}$$

where ρ defines the utilization (load) factor of the queuing system. From eq.(5) and eqs.(4b,c) finally follow the first and second ordinary moments of the (unconditioned) TX-delay T_D

$$E[T_D] = (1-\rho) t_0 \cdot \sum_{\nu=0}^{N-1} 1/\{1 - \rho(1 - \tfrac{\nu}{N})\}, \tag{6a}$$

$$E[T_D^2] = (1-\rho) t_0^2 \cdot \sum_{\nu=0}^{N-1} (2\nu+1)/\{1 - \rho(1 - \tfrac{\nu}{N})\}. \tag{6b}$$

Note, that both limits $E[T_D] \rightarrow N \cdot t_0$ for $\rho \rightarrow 0$ and $E[T_D] \rightarrow t_0$ for $\rho \rightarrow 1$ follow correctly from this approximation.

The utilization (load) factor ρ of the M/G/1 queue is

$$\rho = \lambda \cdot E[T] = \lambda \cdot (E[T_T] + E[T_D] + E[T_{ACK}]) . \tag{7}$$

For stationarity, ρ may not exceed 1; this results in the maximum throughput rate the system can handle

$$\lambda_{max} = 1/(E[T_T] + t_0 + E[T_{ACK}]) . \tag{8}$$

Since $E[T_D]$ itself is a function of ρ, see eq.(6a), the load factor ρ has to be iterated properly according to eqs.(6a) and (7) to reach consistency.

Finally, with the consistent ρ we find all the interesting values from the well-known M/G/1 formulas. The mean waiting time $E[T_W]$ of an arbitrary message is

$$E[T_W] = \rho \cdot \frac{1+c^2}{2(1-\rho)} \cdot E[T] , \tag{9a}$$

where

$$c^2 = E[T^2]/E[T]^2 - 1 \tag{9b}$$

is the squared coefficient of variation of T and

$$E[T^2] = E[T_T^2] - E[T_T]^2 + E[T_D^2] - E[T_D]^2 + E[T]^2. \tag{9c}$$

d) Analysis of collisions

Although collisions have been neglected in the derivation of the M/G/1 model with state-dependent service times, we can estimate the collision probability approximately. Let P_C be the probability that an arbitrary arriving message collides (note, that in *one* collision event at least *two* messages observe collisions).

In the real system, collisions may only occur within the small time window ("collision window") of length t_O following immediately after the transition $I \rightarrow T$ of the common channel (in reality, there are less collisions, since t_O is the upper bound accounting for the maximum propagation delay and safety margins against time tolerances). Candidate messages for collisions are:

(1) messages arriving *after* the expiration of the current TX-delay of the respective station when the channel stays silent until the end of the maximum delay time $(N+1) \cdot t_O$,

(2) messages arriving *within* the collision window itself.

Contrary to the simplified M/G/1 model, in the real system several "busy periods" my follow each other immediately due to the aforementioned effect (1). In the substitute M/G/1 model busy periods (BP) and idle periods (IP) interchange each other. Within one BP we have one collision window at the begin and a geometrically distributed number of collision windows at the end. Within these windows at the begin and end of a BP there are in average

$$\alpha = \lambda t_O$$

$$\beta = \sum_{i=1}^{N} (N+1-i) t_O \cdot \frac{\lambda}{N} + \lambda t_O = \frac{N+3}{2} \cdot \lambda t_O$$

candidate messages for collisions, respectively. Since the Poisson process is additive, the number of candidate messages for collisions follow Poisson distributions a(x) and b(x) in each of the collision windows with parameters α and β, respectively. Thus, the average number of message collisions during one complete BP is

$$n_C = \sum_{x=2}^{\infty} x \cdot a(x) + \sum_{x=2}^{\infty} x \cdot b(x) \,/\, (1 - \sum_{x=2}^{\infty} b(x)).$$

The average number of messages served in a BP is $1/(1-\rho)$.[7] Thus, the wanted collision probability P_C equals

$$P_C = (1-\rho) \cdot n_C = (1-\rho) \cdot \{\alpha(1-e^{-\alpha}) + \frac{\beta(1-e^{-\beta})}{(1+\beta)e^{-\beta}}\}. \tag{10a}$$

For small α and β we find by expansion

$$P_C \approx (1-\rho) \cdot (\alpha^2 + \beta^2). \tag{10b}$$

4. PERFORMANCE RESULTS

The principal performance of the CSMA-CD-DR protocol will be shown by means of throughput and delay curves. Simulation as well as analytic results are given; the latter ones are validated against simulation in case of the low-overhead approximation of Section 3.2.

4.1 Throughput Performance

Fig.5 shows results on the throughput Y (successfully transmitted messages) versus the offered load $A = \lambda \cdot E[T_T]$. The curve for the CSMA-CD-DR protocol is compared with other contention protocols[1]. Due to the low collision rate, the new protocol reveals an excellent overload behavior; the maximum throughput rate is given by eq.(8).

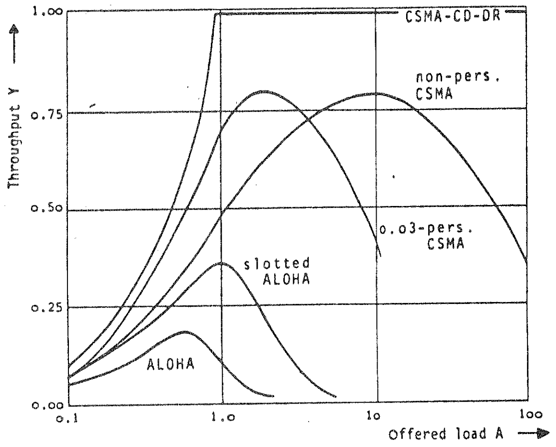

Fig. 5. Throughput of various contention protocols

Parameters: $t_O/E[T_T] = 0.01$ $E[T_{ACK}] = 0.$

4.2 Delay Performance

In Fig.6, the relative average message delay $E[T_W]/E[T_T]$ is plotted versus the offered load A for a system with N = 100 stations under balanced traffic, constant message lengths and various overhead values. The simulation results reveal a peculiar behavior for higher overhead (i.e. increasing t_O, likewise for increasing N or both, N and t_O): the dependence of the TX-delay on the load can be such that the average delay can even decrease with increasing load ! The CSMA-CD-DR protocol behaves in the low traffic region like the other contention protocols. For heavy traffic, it approaches the cyclic polling protocol [8,9,10]: In fact, each station has messages to transmit in the limit; then always that station with the minimum TX-delay t_O transmits whereby the served queues with the minimum TX-delay change in cyclical order.

4.3 Validation of Analytic Results

To validate the approximate analytic results of Section 3.2 against simulation, Fig. 7 gives some results of the delay performance and the collision probability P_C.

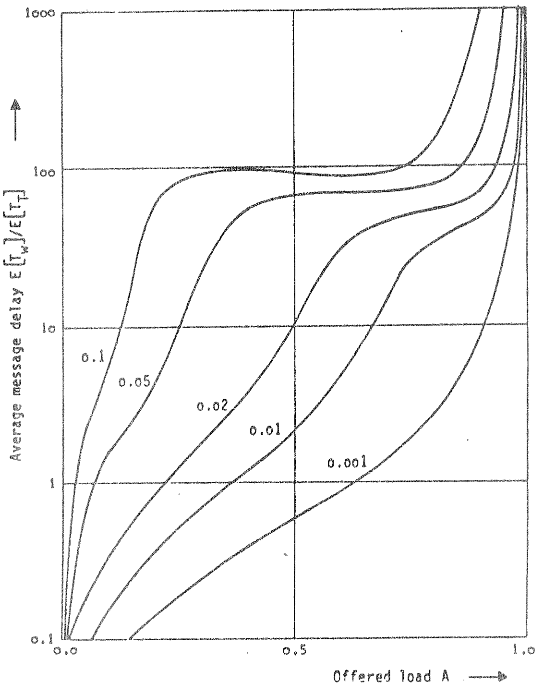

Fig. 6. Average message delays versus offered load (simulation)

Parameters: $N = 100$, constant message lengths
$t_O/E[T_T] = 0.001 \ldots 0.1$, $E[T_{ACK}] = 0$

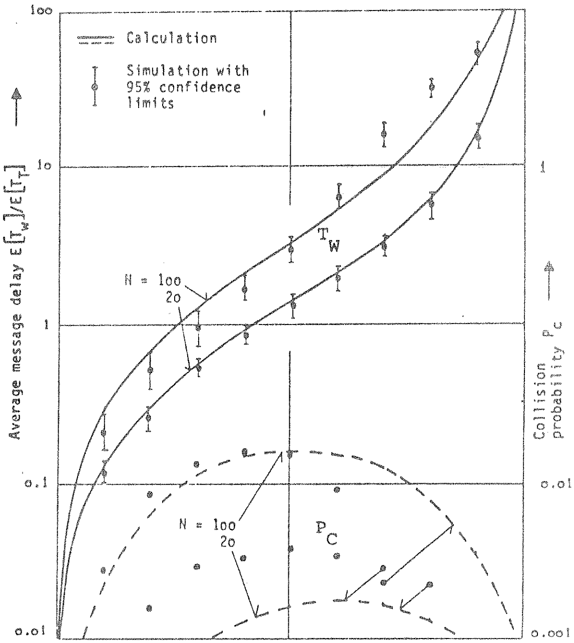

Fig. 7. Validation of approximate analytic results

Parameters: $N = 20, 100$
$t_O/E[T_T] = 0.01$ $E[T_{ACK}] = 0$
exponential message lengths

For small overhead (small t_O or small N), the approximation generally fits very well. In case of higher overhead, the approximation overestimates for low traffic and underestimates for high traffic.Both effects can be explained. In case of low traffic, most transmissions occur in the contention mode; the service time inflation (which is only effective within the BP's) is too large. In case of higher traffic, the assumption of equally probable distances in eq. (2a) does not hold: during the longer BP's, the "curser" station with the minimum TX-delay moves slower than the actual stations being served; thus, the positions with the larger distances relative to the "curser" station will dominate. For the heavy overhead case, the analytic theory must still be improved.

Finally, the collision probability result depends on ρ and, therefore, also on the accuracy of the average delay approximation. However, the simple formulas eq. (10a,b) reflect the principal behavior. Since the confidence intervals for P_c in Fig. 7 are large, they have been omitted for clearity.

CONCLUSION

A new CSMA-type protocol for a common channel with completely distributed stations has been presented. The protocol uses carrier sensing and dynamically staggered transmission delays. The software structure for each of the connected stations has been developed by means of a SDL diagram. Performance results have been given by simulation as well as analytic modeling. The results show that the protocol combines the advantages of random access and fixed/demand assignment protocols at low and high traffic, respectively. The protocol can easily be extended, especially with respect to overload control. The still relatively simple structure of the stations and the results of the performance evaluation allow the conclusion that this protocol seems favourable for local networks and systems with distributed control.

This work is supported by the Deutsche Forschungsgemeinschaft (DFG).

REFERENCES

(1) Tobagi,F.A.,"Multiaccess protocols in packet communication systems". IEEE Trans. Comm., Vol. COM-28, April 1980, pp. 468 - 488.

(2) Spaniol,O., "Analysis and performance evaluation of HYPERchannel access protocols".Performance Evaluation, Vol.1, 1981, pp. 170-179.

(3) Franta,W.R., Bilodeau,M.B., "Analysis of a prioritized CSMA protocol based on staggered delays". Acta Informatica, Vol. 13, 1980, pp. 299 - 324.

(4) Metcalfe,R.M., Boggs,D.R., "ETHERNET: Distributed packet switching for local computer networks". Comm. ACM, Vol.19, July 1976, pp. 395 - 403.

(5) Jacobs,I.M., Binder,R., Hoversten,E.V., "General purpose packet satellite networks". Proc. IEEE, Vol. 66, Nov.1978,pp.1051 - 1068.

(6) Clark,D.D., Pogran,K.T.,Reed,D.P., "An introduction to local area networks". Proc. IEEE, Vol. 66, Nov. 1978, pp. 1497 - 1517.

(7) Kleinrock,L., "Queueing systems. Volume 1: Theory". J. Wiley & Sons, Inc., New York/London/Sydney/Toronto, 1975.

(8) Kuehn,P.J., "Performance of ARQ-protocols for HDX-transmission in hierarchical polling systems". Performance Evaluation, Vol. 1, 1981, pp. 19 - 30.

(9) Kuehn,P.J., "Multiqueue systems with nonexhaustive cyclic service". Bell System Techn. J., Vol. 58, March 1979, pp. 671 - 698.

(10) Arthurs,E., Stuck,B.W., "A theoretical performance analysis of polling and carrier sense collision detection communication systems". Int. Conf. on Performance of Data Communication Systems and Their Applications, Paris, Sept. 14-16, 1981.