Performance issues in local-area networks

by W. Bux

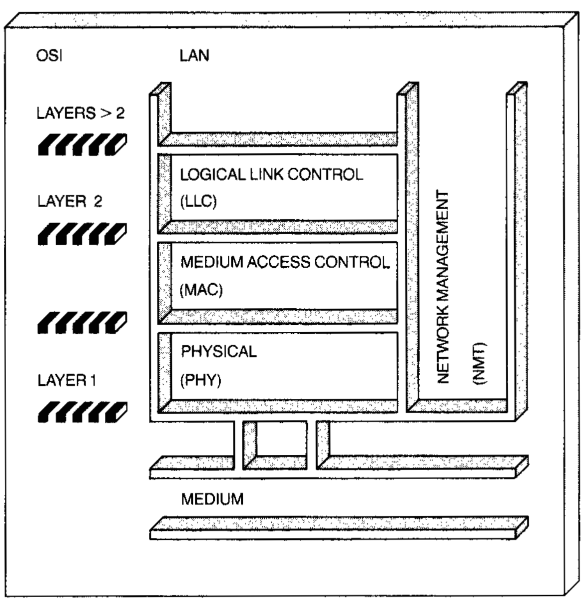
This paper discusses several important performance problems in the design of local-area networks. The questions discussed relate to various aspects of architecture, design, and implementation: (1) the delay-throughput characteristics of the medium access protocols, (2) the performance of local-area networks on which a file server provides file storage and retrieval services to intelligent workstations, and (3) timing problems in local-area network adapters. Since the paper does not primarily address the performance analyst, it is descriptive in nature; analytic details are omitted in favor of a more intuitive explanation of the relevant effects.

The performance evaluation of local-area networks (LANs) is a multifaceted problem because of the complex interaction among a potentially large number of system components. Therefore, modeling of LANs needs to be performed at various levels, similar to the hierarchical approaches in the analysis of equally complex systems, such as wide-area data networks, telephone networks, or computer systems. This paper summarizes some of the performance analysis work done at the IBM Zurich Research Laboratory in the context of a LAN research project. It discusses various aspects of LAN architecture, design, and implementation. The paper does not primarily address the performance-evaluation specialist; its intention is, instead, to provide a sound intuitive understanding of performance problems that are peculiar to LANs. Theoretical details are omitted, but an extensive list of references to the appropriate literature is given in which the interested reader can find additional detailed information. For an introduction to LANs in general, the reader is referred to Clark,[1] Dixon,[2] or Kuemmerle.[3]

A basic category of LAN performance questions is related to the properties of the medium access protocol, i.e., its throughput-delay characteristics. Investigations of the access protocol can provide valuable insight into the overall efficiency of the mechanism, its sensitivity to essential parameters (transmission rate, cable length, number of stations, etc.), and other important properties, e.g., fairness of access. The second section of this paper is devoted to an overview of the performance characteristics of important LAN medium access protocols.

Models of the above type are suitable to assess the performance characteristics of different access mechanisms (which usually imply a certain network topology) and thus are helpful in finding a good network design. Such models, however, are not appropriate for determining application-oriented performance measures. If one is interested, for example, in the quality of a file service, higher-level protocols, i.e., Logical Link Control, Network, Transport, and Session protocols have to be modeled. Moreover, implementation choices, such as the user-system-to-network interface or buffer management, may have an important effect on the quality of service seen by a user. In the third section, we describe a model of this category, i.e., a file server providing file storage and retrieval services over a LAN to a set of intelligent workstations.

Figure 1 LAN architecture reference model



In order to answer detailed questions about the performance of specific components of a system, modeling at a rather deep level of detail may be necessary. A typical example is discussed in the fourth section, where a model of an adapter to a LAN is described. This model was used to study the timing problems associated with the reception of a continuous stream of information.

With these three categories of models, we cover a rather broad spectrum of performance issues related to LANS; nevertheless, there are various important topics we do not address in the present paper. Examples are problems related to the interconnection of local subnetworks, an area where flow and congestion-control problems arise. Furthermore, specific applications, for example, transmission of voice and images, raise challenging performance problems. Additional important areas are traffic measurement methodologies and the traffic-related aspects of network management and configuration.

Delay-throughput characteristics of medium access protocols

The groups concerned with LAN standardization, the Institute of Electrical and Electronics Engineers (IEEE) Project 802 and the European Computer Manufacturers Association (ECMA) TC24, have adopted a LAN architecture model that describes the relationship of LAN architecture and the Open Systems Interconnection (OSI) Reference Model.[3-14] As shown in Figure 1, the OSI data-link layer is split into two sublayers, the medium-dependent "Medium Access Control" (MAC) sublayer and the medium-independent "Logical Link Control" (LLC) sublayer. Peculiarities of the various local-network techniques are thus restricted to the medium, the physical layer, and the MAC sublayer. Consequently, the quality of service at the MAC-to-LLC interface differs between different local-area networks. An important aspect of this service is the delay-throughput characteristic, which will be treated in this section.

We focus on the discussion of the three methods that have been standardized: Carrier-Sense Multiple-Access with Collision Detection (CSMA/CD), token ring, and token bus.[5-9,11-13]

CSMA/CD. Carrier-sense multiple-access with collision detection can be viewed as the offspring of CSMA methods developed for broadcast systems, mainly ground-radio packet-switching systems. Immediate detection of collisions is difficult in radio systems, whereas a rather simple collision-detection technique can be employed on bus systems, at least, if baseband transmission is used. Collision detection helps to improve performance in a short-delay environment. CSMA/CD was first described in Reference 15 as the access protocol of Ethernet.[16] In the meantime, ECMA and IEEE Project 802 have produced standards specifying a CSMA/CD-based local-area network.

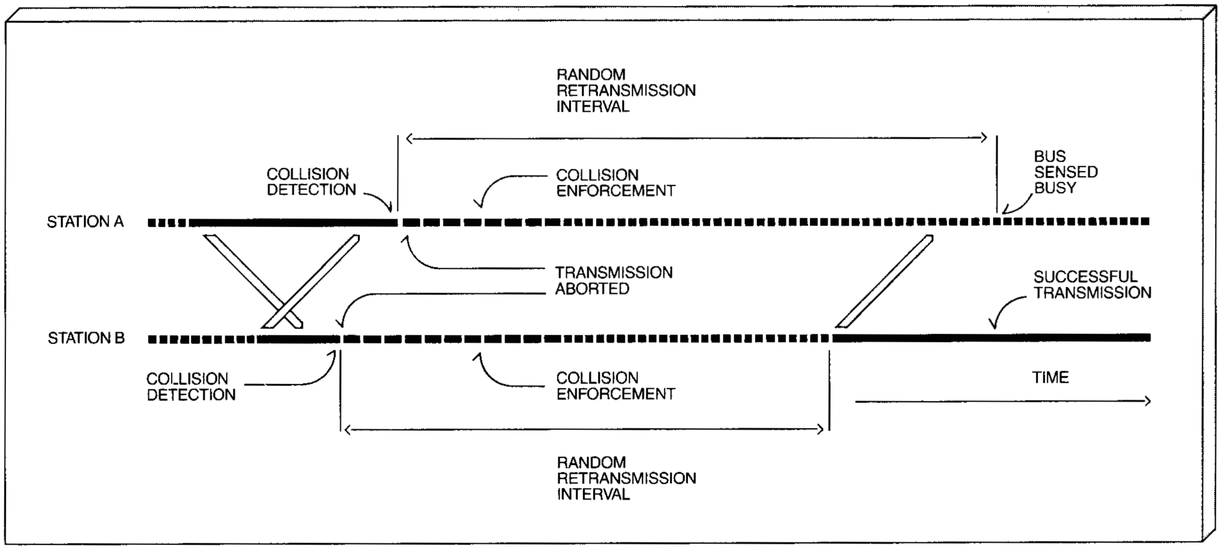
The following brief description of the CSMA/CD protocol follows the specification in the existing standards.[5-7,11]

Medium access protocol. The protocol can conceptually be divided into a transmission and a reception part.

In the transmission part, when a station has a frame ready for transmission, it monitors the cable to determine whether any transmissions take place. When the medium is found utilized, transmission is deferred. When the medium is clear, frame transmission is initiated (after a short interframe delay, e.g., 9.6 microseconds).

If multiple stations attempt to transmit at the same time, interference can occur (see Figure 2). Overlap of different transmissions is called a collision. In this case, each transmitting station enforces the collision

Figure 2 CSMA/CD: example of operation (from Ref. 21)
©1981 IEEE



by transmitting a bit sequence called the jam signal. This ensures that the duration of the collision is sufficient to be noticed by all other stations involved in the collision. Then stations schedule a retransmission attempt for a randomly selected time in the future. Retransmission is attempted repeatedly in case of subsequent collisions. Repeated collisions indicate a heavily utilized medium; therefore stations adjust their retransmission activity to the traffic load perceived. This is accomplished by expanding the mean of the random retransmission time on each retransmission attempt.

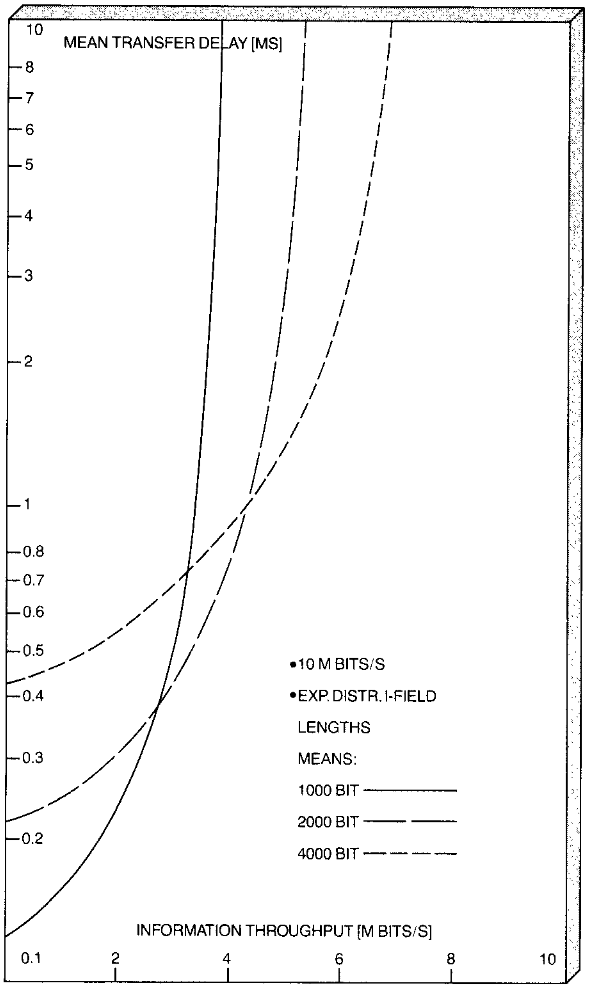
The scheduling of the retransmissions is determined by a process called "truncated binary exponential backoff." Retransmission times are an integral multiple of the so-called slot time. The slot time must be equal to or greater than the maximum round-trip signal propagation time of the system. For the 10-million-bit-per-second baseband CSMA/CD system, a slot time of 51.2 microseconds has been standardized. The number of slot times to be delayed before the nth retransmission attempt is taken from a discrete distribution that assumes all integer values between 0 and 2^n with equal probability. If ten retransmissions of the same frame fail, the attempt is abandoned, and an error is reported.

The CSMA/CD access mechanism requires transmission of frames of a minimum length. If the frame size is less than the minimum required, a transmitting station must append extra, so-called "pad" bits after the end of the LLC-supplied data. Standardized minimum frame length for a CSMA/CD system with a baseband of 10 million bits per second is 512 bits.

In the reception part, all active stations synchronize with the preamble of an incoming frame and then decode the received signal. The destination-address field of the frame is checked to decide whether the frame should be received by this station. If so, the relevant parts of the frame are copied. The station also checks the validity of the received frame by inspecting the frame check sequence and proper octet-boundary alignment.

Performance characteristics. The performance of CSMA and CSMA/CD systems has formed the subject of numerous studies. The groundwork for the understanding of the performance properties of CSMA was laid in References 17 and 18. CSMA/CD performance has been studied in References 15 and 19 through 23 for different variants of the access principle. To provide a basic understanding of the delay-throughput characteristic of the standard CSMA/CD protocol, an analysis based on the work by Lam[22] appears attractive because the approach is rather straightforward, the underlying assumptions are close to the standardized CSMA/CD protocol, and the results are simple to evaluate numerically.

Figure 3 CSMA/CD delay-throughput characteristic (exponentially distributed information-field lengths)



The assumptions underlying the analysis in Lam[22] are as follows. The traffic offered to the network is a Poisson process with a constant and state-independent arrival rate. Each station is allowed to store at most one frame at a time. The generation of a new frame is equivalent to increasing by one the number of stations ready to transmit a frame. Frame transmission times are generally distributed.

The following assumptions are made regarding the medium access protocol: (1) Following a successful transmission, all ready stations transmit within the next slot. (2) Following a collision, stations use an adaptive retransmission algorithm in such a way that the probability of a successful transmission within any of the slots subsequent to a collision is constant and equal to $1/e$ (= 0.368). For a large number of stations, this assumption is well justified.[15-22] (3) Operation is assumed to be slotted in time, i.e., transmission attempts are made only at the beginning of a slot.

Under the above assumptions, the mean queuing delay of the frames was determined in Lam.[22] The examples shown below have been computed with the aid of this solution; however, we modified the analysis in the following three points:

1. It has been assumed in Lam[22] that after every successful transmission, a time interval equal to the end-to-end signal propagation time expires before stations sense end-of-carrier and start to transmit. This represents a slightly pessimistic view of the operation as described in the previous subsubsection. For our results, we assumed that end of transmission is detected with zero delay by all stations.
2. A consequence of assuming a slotted channel is that, even if the channel utilization approaches zero, frames have to wait for half a slot length on the average. Such a delay, of course, does not occur on a nonslotted system, such as the one described in the last subsubsection. We therefore reduce the mean delay according to Lam[22] by half the slot length.
3. As pointed out above, the CSMA/CD access protocol requires a minimum frame length of at least the slot length measured in bits. This fact has not been taken into account in Lam.[22] However, it can be easily incorporated through an appropriate modification of the distribution function of the frame transmission times.

In Figures 3 to 5, we show basic results for delay and throughput of CSMA/CD systems. As parameters for these examples, the values standardized for the 10-million-bit-per-second baseband CSMA/CD system have been used.[5-7,11]

Figure 3 shows the mean transfer delay of the transmitted frames as a function of the information throughput. The frame transfer delay is the time from the generation of a frame until its successful reception at the receiver. The information throughput is defined as the number of bits contained in the LLC-Information field of the frames transmitted per unit time. An exponential distribution for the information-field lengths is assumed. As described above, padding bits are added when the frame length is shorter than the minimum required. We observe from the figure that the delay-throughput characteristic depends strongly on the mean length of the information field. The shorter the frame length, the smaller the delay at small throughput values, but also the smaller the maximum throughput and hence the steeper the increase of the delay curves. The reason for this behavior is that with a decreasing ratio of frame transmission time to slot length, the protocol overhead increases significantly in terms of the fraction of time lost for collisions and their resolution.

In Figure 4, the behavior of the same system, however, for constant information-field lengths of 1000, 2000, and 4000 bits is shown. It can be seen that the delays are smaller than for the exponential distribution. However, the general tendency is the same, in particular, the location of the vertical asymptotes; i.e., the maximum throughput is very insensitive to the information-field length distribution. Generally, the type of this distribution can have an impact on the maximum throughput because of the minimum frame-length requirement. As comparison of Figures 3 and 4 shows, this impact is relatively small for the information-field lengths considered.

How the maximum throughput depends on the transmission speed, given a slot length equal to the standardized value of 51.2 microseconds, is shown in Figure 5. For the different values of the mean information-field length, an area for the maximum throughput is indicated in this figure. The upper and lower boundaries of these areas are determined by two different considerations regarding the situation in the first slot following a successful transmission. In the above-described approach to determine the mean delays, a constant, state-independent frame arrival rate has been assumed. If, under this assumption, the traffic load reaches the system capacity, the probability of a collision after a successful transmission will approach one. As described above, the probability of a successful transmission in one of the subsequent slots is equal to $1/e$. Under this assumption, the lower bound of the maximum throughput regions in Figure 5 has been determined.

A more optimistic assumption is that in an overload case, the probability of a successful transmission in the first slot after a successful transmission is equal to $1/e$. Under this assumption, the average time between subsequent successful transmissions is one slot length shorter than for the more pessimistic assumption first described. The optimistic assump-

Figure 4 **CSMA/CD delay-throughput characteristic (constant information-field length)**

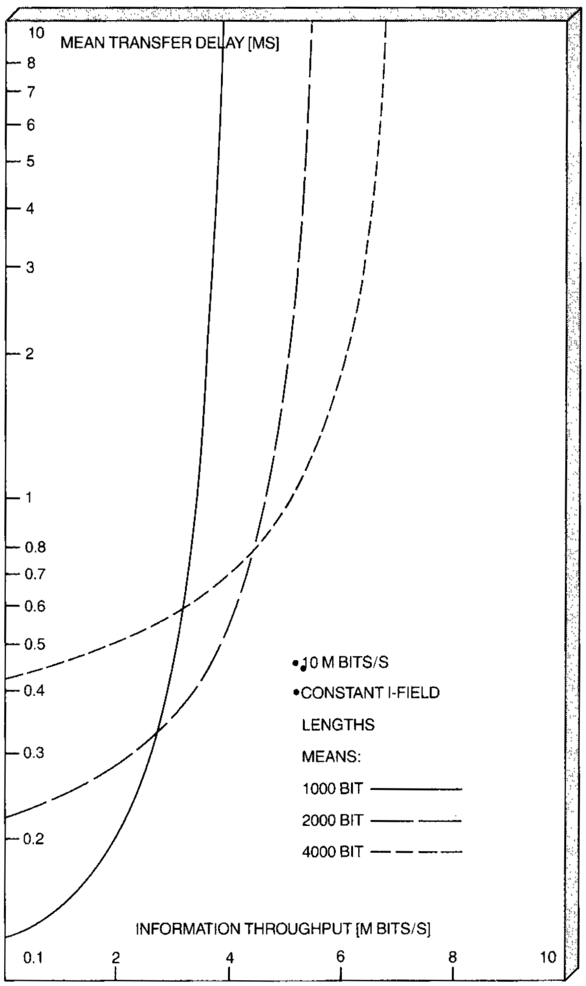


Figure 5 **CSMA/CD maximum information throughput versus transmission rate for 51.2-microsecond slot length**

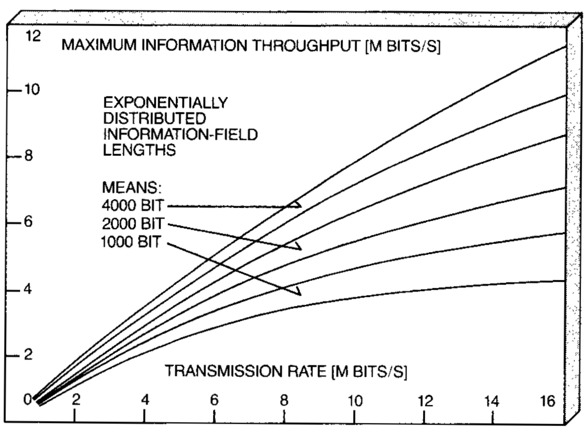
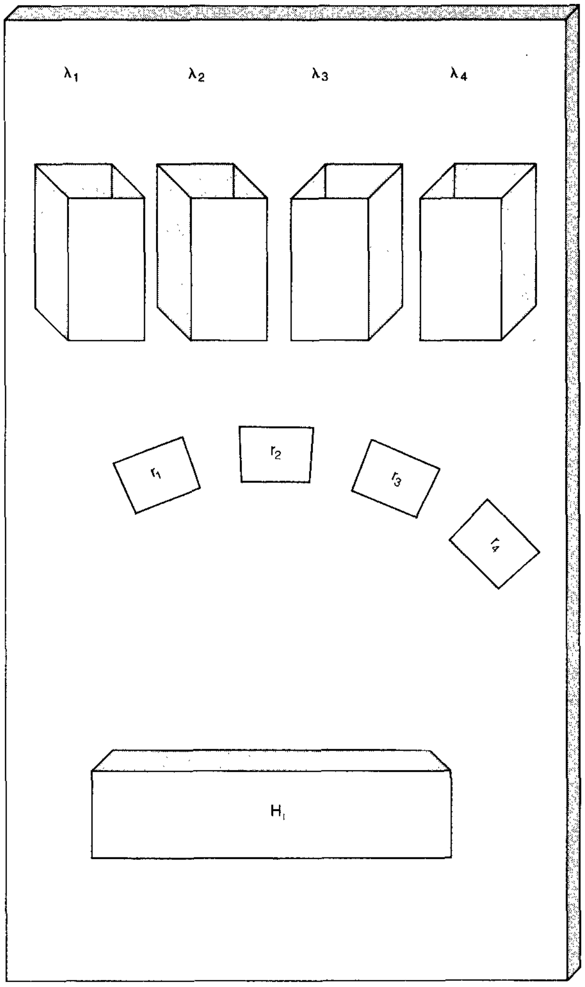


Figure 6 Token-ring queuing model (from Ref. 32, reprinted with permission of North-Holland Publishing Company)



tion underlies the analysis given by Metcalfe and Boggs.[15] This solution has been used to determine the upper bound of the shaded areas in Figure 5. It can be seen that, even under the optimistic assumption, the efficiency of the CSMA/CD protocol decreases significantly with increasing speed, especially in the case of a 1000-bit mean information-field length. Measurements performed on an Ethernet showed a mean frame length of 976 bits.[24]

Token ring. Compared to the other LAN techniques, token rings have a relatively long technical history. Experimental systems showed the feasibility of the ring technique long before alternative methods, e.g., CSMA/CD or token-passing bus systems, were considered.[25–27] However, because of a lack of applications, token rings were not implemented on a broad basis. With the advent of local-area networks, the token-ring principle was reconsidered and found to provide an attractive solution because of its favorable attributes regarding wiring, transmission technology, performance, and the potential for low-cost implementation.[1–3,27–30] Moreover, recent work showed that what had been considered a potential problem of token rings, namely, lack of reliability, can be overcome by a suitable access protocol and an appropriate wiring strategy.[28,29] The above arguments led the standards groups concerned with LAN standardization to consider token rings as one of the candidates for a LAN standard.

The following description is based on the specification of the token-ring operation given in the existing IEEE-802 and ECMA standards.[8,13]

Medium access protocol. A token ring consists of a set of stations serially connected by a transmission medium, e.g., twisted-pair cable. Information is transferred sequentially from one active station to the next. A given station (the one having access to the medium) transfers information onto the ring. All other stations repeat each bit received. The addressed destination station copies the information as it passes. Finally, the station which transmitted the information removes it from the ring.

A station gains the right to transmit when it detects a token passing on the medium. The token is a control signal comprised of a unique signaling sequence that circulates on the medium following each information transfer. Any station, upon detection of a token, may capture the token by modifying it to a start-of-frame sequence, and then appends appropriate control and address fields, the LLC-supplied data, the frame check sequence, and the frame-ending delimiter. On completion of its information transfer and after appropriate checking for proper operation, the station generates a new token which provides other stations the opportunity to gain access to the ring.

A token-holding timer controls the length of time a station may occupy the medium before passing the token.

Multiple levels of priority can be provided on a token ring through an efficient priority mechanism. This mechanism is based on the principle described in Bux et al.[28] whereby higher-priority stations can interrupt the progression of lower-priority tokens and frames by making "reservations" in passing frames. This scheme requires that stations do not issue a new token before having received back the header of their transmitted frame. This so-called "single-token" rule[21,28] also leads to improved reliability of the access protocol because each transmitting station can check the proper functioning of the ring at the beginning of its transmission.

Performance characteristics. The basic operation of a token ring can be described by a performance model as shown in Figure 6. The active stations are represented by their transmit queues. These queues are serviced in a cyclic manner symbolized by the rotating switch that stands for the token.

The time needed to pass the token from station i to station $(i + 1)$ is modeled by a constant delay r_i. On an actual ring, the delay r_i corresponds to the propagation delay of the signals between stations i and $(i + 1)$ (approximately five microseconds per km cable) plus the latency caused within station i by the repeater and by actions such as alteration of the token bit. The station latency is usually in the order of one bit time.

In token rings, the sender is responsible for removing the frames it transmitted from the ring. Therefore, the location of frame destinations on the ring relative to the location of the sender does not affect the ring performance.

Queuing models applicable to token rings have been extensively studied, primarily in the context of polling systems.[20,21,27,31–42] However, analytic results, and especially those that lend themselves to numerical evaluation, are scarce. This is particularly true for models in which the transmission time of a station per access opportunity is limited through a bound on either the token-holding time or the number of frames to be transmitted per token.

Subsequently, we discuss some fundamental results for the token-ring delay-throughput characteristic obtained through simulation and analysis (where applicable).

As pointed out above, a fundamental performance characteristic of any LAN medium access protocol is its sensitivity to transmission speed and distance. Figures 7 and 8 show how token rings perform for various speeds and distances. Figure 7 shows the mean frame transfer delay as a function of the information throughput for four-million-bit-per-second rings with one- and five-kilometer (km) cable lengths. It is assumed that all 100 stations generate the same amount of traffic; other traffic patterns lead to very similar results for the delay averaged over all stations. A further assumption is that frames are generated according to Poisson processes. Stations follow the single-token rule described in the last subsubsection; i.e., they wait until the header of their frame has returned before generating a new token. Only one frame per access opportunity can be transmitted. It can be seen that increasing the ring length from one to five km has virtually no impact on the delay-throughput characteristic.

Figure 7 Token-ring delay-throughput characteristic (four-million-bit-per-second transmission rate; symmetrical traffic pattern)

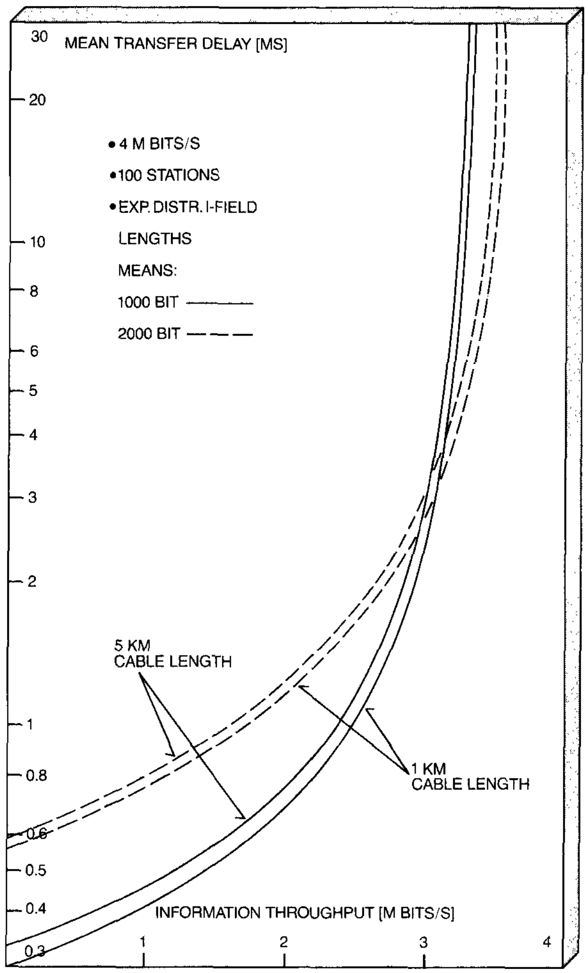
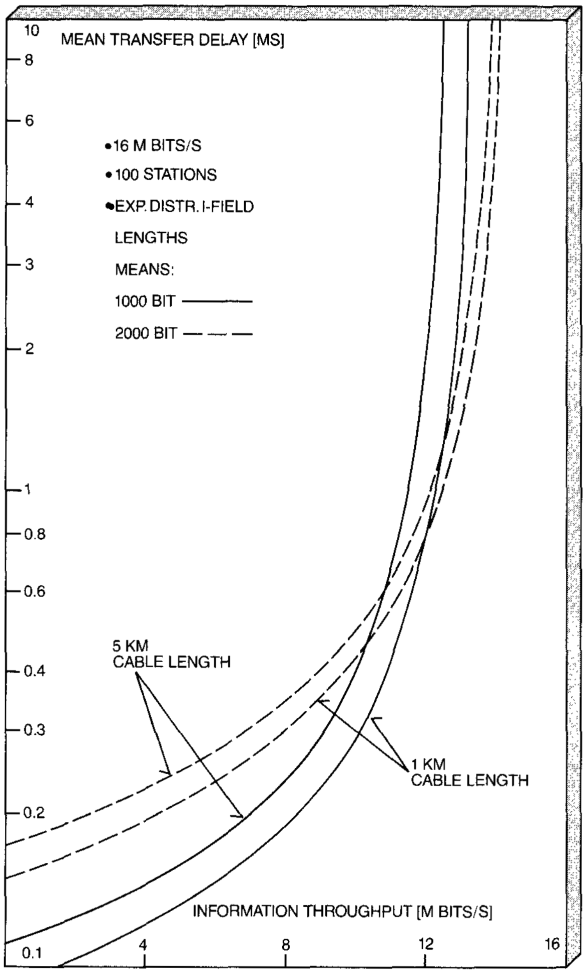


Figure 8 Token-ring delay-throughput characteristic (16-million-bit-per-second transmission rate; symmetrical traffic pattern)

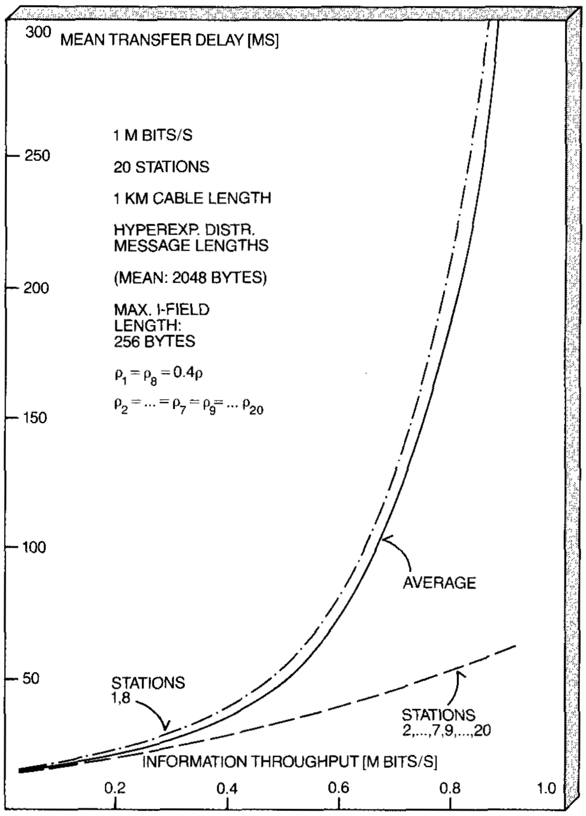


Under the same assumptions, except for a transmission rate of 16 million bits per second, Figure 8 shows the same performance measures as the previous one. Increasing the cable length from one to five km leads to more noticeable differences here, primarily because of the single-token rule; however, the overall effect is still minor.

Overall efficiency of an access protocol is the most basic performance property; a further important criterion is the quality of service given to individual stations, especially in case of unbalanced traffic situations. This service can differ significantly, depending on the rule defining the time a station is allowed to transmit per access opportunity. As mentioned in the last subsubsection, the standards specify the use of a token-holding timer that limits the time a station is allowed to transmit continuously. To demonstrate the impact of this timer, we subsequently consider two extreme cases, a very short timer, such that stations can only transmit one frame per token (Figure 9), and a very long timer, such that stations can always empty their transmit queues completely on each transmission opportunity (Figure 10).

For both examples, Poisson arrival processes have been assumed. However, the arriving data units are not single frames but entire messages, the lengths of which are distributed according to a hyperexponential distribution with a coefficient of variation equal to two. In cases where a message is longer than the maximum information-field length of a frame (256

Figure 9 Token-ring delay-throughput characteristic (one-million-bit-per-second transmission rate, asymmetrical traffic pattern, short token-holding time-out) (from W. Bux, F. Closs, K. Kuemmerle, H. Keller, and H. R. Mueller, "Architecture and Design of a Reliable Token-Ring Network," *IEEE Selected Areas in Communications* SAC-1, No. 5, 756–765 (November 1983). ©1983 IEEE

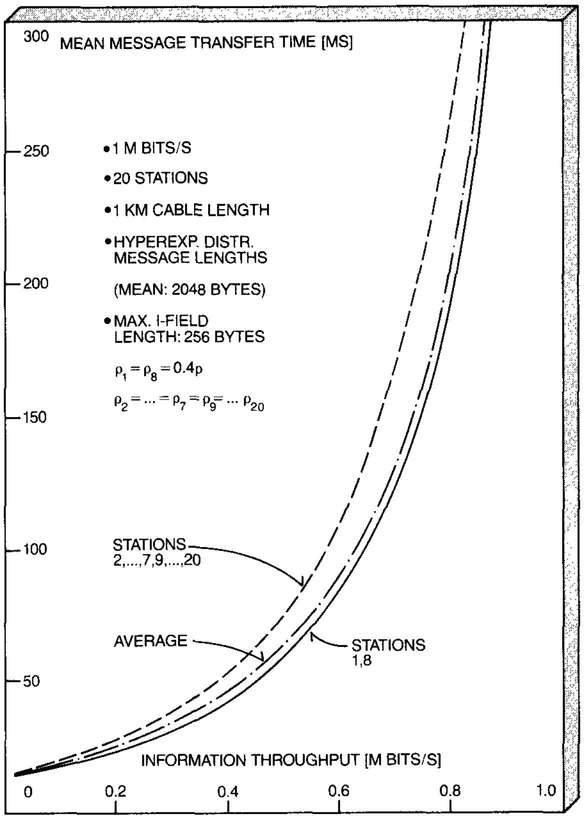


bytes), the message is segmented. In both examples, the assumed traffic pattern is very unbalanced: two of the 20 stations (Nos. 1 and 8) each generate 40 percent of the total traffic; each of the other 18 stations generates only 1.1 percent of the total traffic.

For the single-frame-per-token operation, Figure 9 shows the mean transfer delay of the messages (not frames!) as a function of the total information throughput. Of course, the delay averaged over all stations increases with increasing ring throughput. The same is true for the delay of the messages transmitted by the heavy-traffic stations 1 and 8. However, the delay experienced by the light-traffic stations remains rather small even for very high utilizations. In this sense, the token-passing protocol combined with a single-frame-per-token operation provides fair access to all users.

From Figure 10, it can be seen that the relationship of the delay experienced by light and heavy users is reversed when the token-holding time is long. Here, the mean message transfer delay of light-traffic stations is even higher than the one of heavy-traffic stations. This is due to the fact that messages generated at a heavy-traffic station have a relatively good chance that their station is holding the token and, in this case, are transmitted before frames waiting in other stations. These two examples demonstrate that the token-holding timer can be used to control the station-specific quality of service.

Figure 10 **Token-ring delay-throughput characteristic (one-million-bit-per-second transmission rate, asymmetrical traffic pattern, long token-holding time-out)**



Token bus. The token-bus technique is the third method being considered by the LAN standards bodies. The intention behind developing this technique has been to combine attractive features of a bus topology (e.g., use of broadband transmission) with those of a controlled medium access protocol (e.g., good efficiency under high traffic load, speed-distance insensitivity, and fairness of access).

The subsequent description follows the specification of the token bus given in References 9 and 12.

Medium access protocol. The essence of the token-bus access method can be characterized as follows. A token controls the right to access the medium; the station that holds the token has momentary control over the medium. The token is passed among the active stations attached to the bus. As the token is passed, a logical ring is formed (see Figure 11). Since the bus topology does not impose any sequential ordering of the stations, the logical ring is defined by a sequence of station addresses.

Steady-state operation simply requires the sending of the token to a specific successor station when a station has finished transmitting. A more difficult task is establishing and maintaining the ring (initialization, station insertion in, or removal from, the logical ring). Each participating station knows the addresses of its predecessor and its successor. After a station has completed transmitting data frames, it passes the token to its successor by sending a special MAC control frame, called an "explicit token." The maximum transmission time of any station is controlled by a token-holding timer.

After having sent the token, the station monitors the bus to make sure that its successor has received the token and is active. If the sender detects a valid frame following the token, it will assume that its successor has the token and is transmitting. If the sender does not sense a valid frame from its successor, it must assess the state of the network and, if necessary, take appropriate recovery actions to re-establish the logical ring. Details about establishment and re-establishment of the logical ring are specified in References 9 and 12.

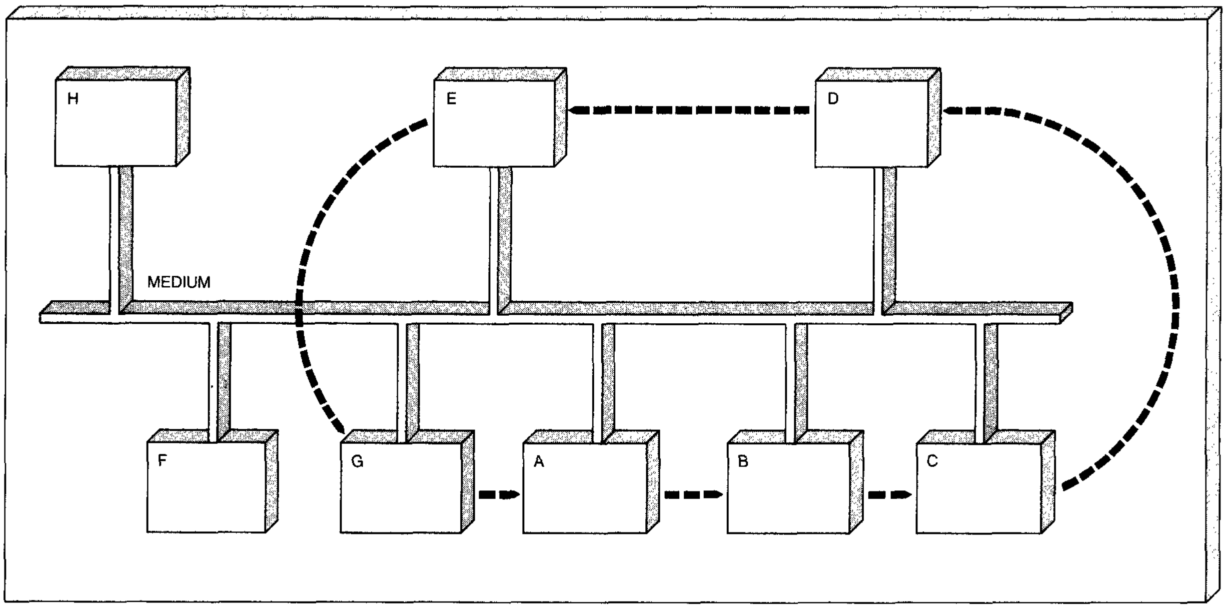
Conceptually, token passing on buses and rings is very similar.

The token-bus access method also allows defining of a priority mechanism, which is not further discussed here.

Performance characteristics. Conceptually, token passing on buses and rings is very similar; hence, the same type of performance model can be used to describe the two techniques. It is obvious, however, that the model parameters are rather different; this is particularly the case for the token-passing overhead. In a token ring, the time to pass the token from one station to the next consists of the signal propagation time between the two stations (approximately five microseconds per km cable) plus the delay caused within a station. As pointed out in the previous subsubsection, the latter delay can be kept as small as one bit time. In contrast to this, on a token bus, passing the token from a station to its successor requires the transmission of an explicit-token frame, which in the standard for the token bus is 152 bits long. To this the signal propagation delay between the two stations has to be added. The third component of the token-passing overhead is the reaction time of the station, i.e., the time a station needs from reception of a token until it has prepared either a token or a data frame for transmission.

In Figure 12, we show the delay-throughput characteristic of one-million-bit-per-second token-bus systems with 100 and 200 stations attached. Further assumptions are: two km cable length, exponentially distributed information-field lengths with means of 1000 and 2000 bits, and zero reaction (processing) delay in the stations. For this example, the traffic is assumed to be completely symmetrical; i.e., all stations generate the same amount of traffic. Furthermore, for this and the following example, a token-holding time is assumed that is sufficiently long for stations always to be able to completely empty their

Figure 11 Token bus: logical ring on physical bus



transmit queues at each transmission opportunity. The figure shows that the mean transfer delays are remarkably high compared with one and two milliseconds, respectively, the time it takes to transmit an information field of average length. This is due to the relatively large token-passing overhead of the token-bus technique.

As the next figure demonstrates, the token-passing overhead is reduced in case of asymmetric traffic. The parameters assumed for Figure 13 are a rate of five million bits per second (which is another one of the standardized speeds), a two-km cable length, 100 stations, and exponentially distributed information-field lengths with a mean of 1000 bits. Three different

Figure 12 Token-bus delay-throughput characteristic (one-million-bit-per-second transmission rate, symmetrical traffic pattern)

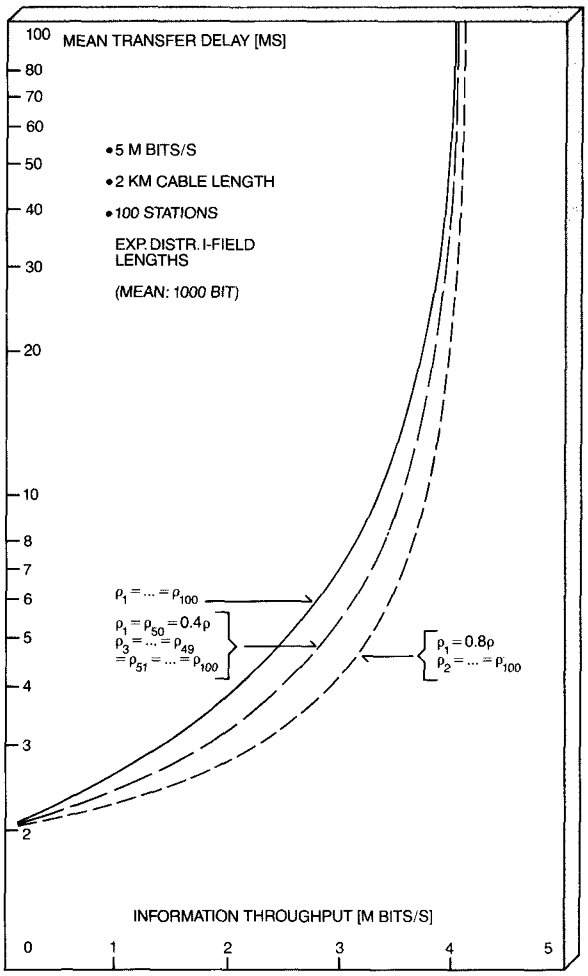
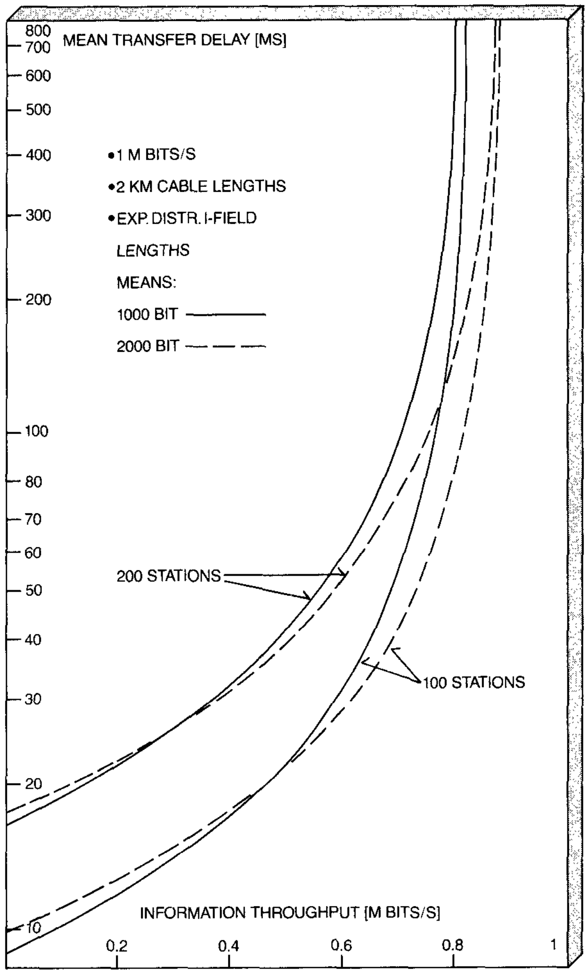


Figure 13 Token-bus delay-throughput characteristic (mean transfer delay averaged over all stations for *different symmetrical and asymmetrical traffic* patterns)



traffic patterns are assumed: (1) a totally symmetrical situation, (2) a situation where two stations each generate 40 percent of the total traffic, the rest being generated by the other stations in equal amounts, and (3) a situation with one station generating 80 percent of the traffic while again the rest is generated by the other stations. For each of these traffic patterns, the figure shows the mean frame transfer delay averaged over all stations as a function of the total ring information throughput. We observe that with increasing asymmetry of the traffic, the average delay decreases slightly, because—per frame transmission—the overhead to forward the token is smaller. It should be noted, however, that this is only true when the token-holding time-out is sufficiently long; for a short token-holding time-out, the effect is reversed.

Other local network techniques. In addition to the three "standard" approaches discussed previously, various alternative LAN techniques have been developed and used. Among the most attractive techniques are slotted rings,[27,43] buffer-insertion rings,[27,44,45] buses with controlled-type access,[37,46–49] or buses employing a combination of random and controlled access.[37,49–53] Because of a lack of space, we cannot discuss these methods in detail; for the interested reader, we subsequently list references in which performance questions related to the above systems are discussed.

The performance of slotted rings is discussed in References 21, 31, 54 and 55. Analyses of buffer-insertion rings can be found in References 56 through 58. Controlled and hybrid-access schemes for buses have been analyzed in References 21, 49, and 59 through 64.

End-to-end flow and error control

Introduction. Models of the type discussed in the previous section are useful in understanding the quality of service provided by the local network at the MAC-to-LLC interface. Apparently, the performance characteristics seen at this interface are not the ones experienced by a user at the application level. To determine application-oriented performance measures, additional levels of architecture need to be modeled, such as an end-system-to-end-system protocol providing means for flow and error control.

The need for flow control arises in cases of a speed mismatch between the communicating partners, limited buffer sizes in the end systems and/or network adapters, and applications where, e.g., one station provides a certain service simultaneously to multiple workstations.

Means to detect and recover from errors are needed for various reasons: (1) Data units can be corrupted by transmission errors; (2) Frames may be lost because of buffer overflow in the receiving end system and/or its adapter; (3) Timing problems in the receivers may cause loss of frames (see the following major section on LAN adapter design).

Protocols providing the functionality needed for flow and error control in LANs are, for example, Class 4 of the ISO/ECMA Transport Protocol,[65,66] or the Type 2 Logical Link Control protocol defined by the IEEE Project 802.[10] Depending on this choice, end-to-end flow and error control is performed in layers corresponding to either layers 4 or 2 of the OSI reference model.

In the next subsection, a scenario consisting of a file server and workstations attached to a local-area ring network is described. The subsection after that describes a model developed to study performance issues of such systems.[67] Results of this study are summarized in the subsequent subsection.

Flow control is implemented by a window mechanism.

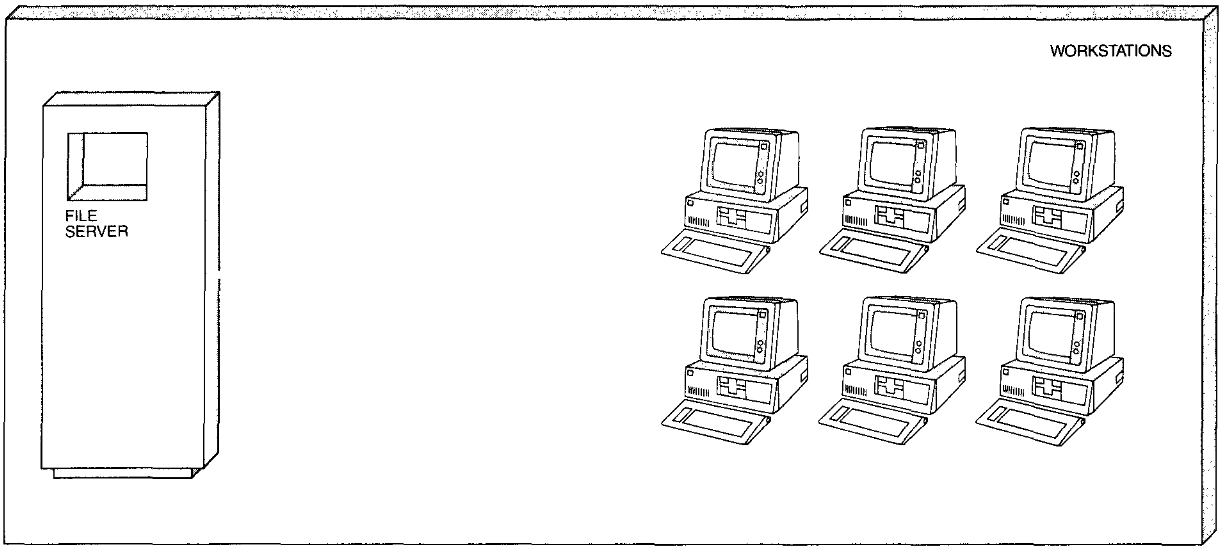
Network operation. The configuration of the local-area network under consideration is shown in Figure 14. It consists of user systems (file server and workstations) attached to a token-ring network through ring adapters. Each adapter has a number of transmit/receive buffers. It also contains a processor whose major tasks are to control the data transfer between the ring and the transmit/receive buffers, to manage these buffers, and to control the interface to the user system.

File transfer is performed over a logical connection between the file server and the workstation. The file server can manage multiple connections simultaneously. The protocol under consideration is a subset of the IEEE 802.2 Type 2 Logical Link Control protocol.[10] It provides procedures for connection establishment, connection termination, flow control, and error recovery.

A file is transmitted as a series of Information (I-) frames. For the file transfer environment, information flow on a given connection is unidirectional; i.e., on one connection, I-frames are either sent from the file server to a workstation, or vice versa.

Flow control. Flow control is implemented by a window mechanism; i.e., a sender is permitted to

Figure 14 File-service scenario



transmit up to W (the window size) I-frames without having to wait for an acknowledgment. The receiver uses Receive Ready (RR-) frames to acknowledge correctly received I-frames and to indicate to the sender that more I-frames can be transmitted.

Error recovery. Any I-frame received with an incorrect Frame Check Sequence (FCS) is discarded. If a received I-frame has a correct FCS, but its send sequence number is not equal to the one expected by the receiver, the receiver will return a Reject (REJ-) frame. The receiver then discards all I-frames until the expected I-frame has been correctly received. The sender, upon receiving a REJ-frame, retransmits I-frames starting with the sequence number received within the REJ-frame.

In addition to REJECT recovery, a time-out mechanism is used. At the instant of transmission of an I-frame, a timer will be started if it is not running already. When the sender receives an RR-frame, it restarts the timer if there are still unacknowledged I-frames outstanding.

When the timer expires, the station performs a "checkpointing" function by transmitting an RR-frame with a dedicated bit (the "P-bit") set to one. The receiver, upon receiving this frame, must return an RR-frame with the "F-bit" set to one. When this RR-frame has been received by the sender, it either proceeds with transmitting new I-frames or retransmits previous I-frames depending on the sequence number contained in the RR-frame received.

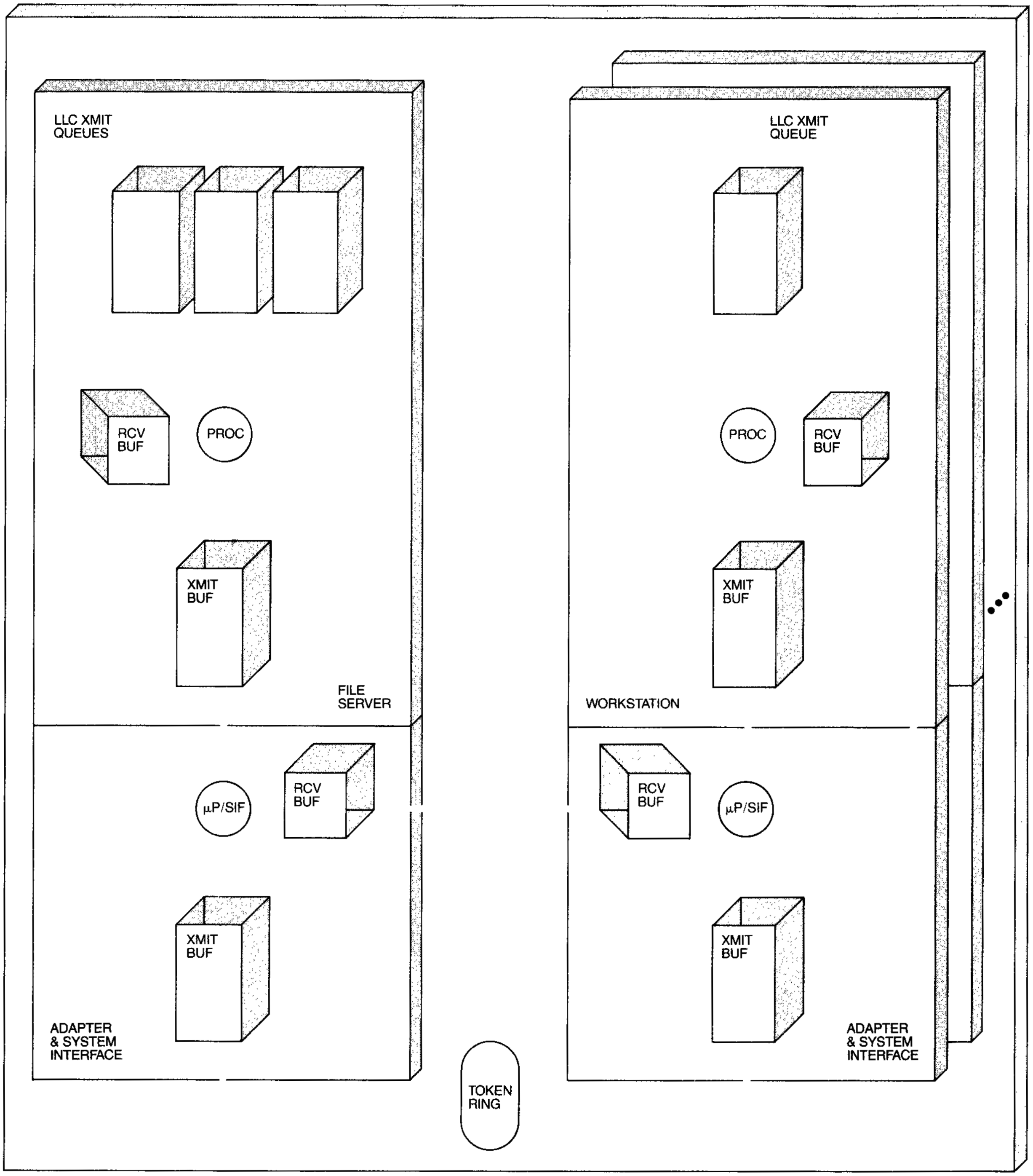
Simulation model. A simulation model employed to study the above scenario is illustrated in Figure 15 and subsequently described.[67]

Medium access protocol. The token-ring protocol for medium access is modeled by a multiqueue, single-server submodel with cyclic service (cf. the earlier section on performance characteristics of the token ring). A queue in this submodel represents the frames waiting in the transmit buffers of an adapter.

Ring adapter and system interface. The adapter transmit buffers contain frames to be transmitted onto the ring. The adapter receive buffers temporarily hold frames received from the ring until they can be transferred to the user system. When upon arrival of a frame no receive buffer in the adapter is available, the frame is lost and has to be recovered through the LLC protocol. Transmission errors are assumed to have negligible effect and are not included in the model. Furthermore, it is assumed that timing problems associated with the receive operation, such as the ones described in the next major section, do not exist here.

The adapter processor, together with the system interface, is modeled by a single server with two queues: the receive buffer queue in the adapter and

Figure 15 File-service performance model



the transmit buffer queue in the user system (see Figure 15). The service time corresponds to the sum of the time to set up a transfer by the adapter processor and the data-transfer time across the system interface. The adapter processor handles frames in its receive buffer with nonpreemptive priority over those in the transmit buffer.

File server. The processor in the file server is modeled by a multiqueue, single-server model. One of the queues is the receive-buffer queue in the file server; the others are for the various connections, containing frames to be prepared for transmission to the workstations. The receive-buffer queue is given nonpreemptive priority over the transmit queues. Among the transmit queues, service is cyclic. Received I-frames are copied to the mass storage of the file server.

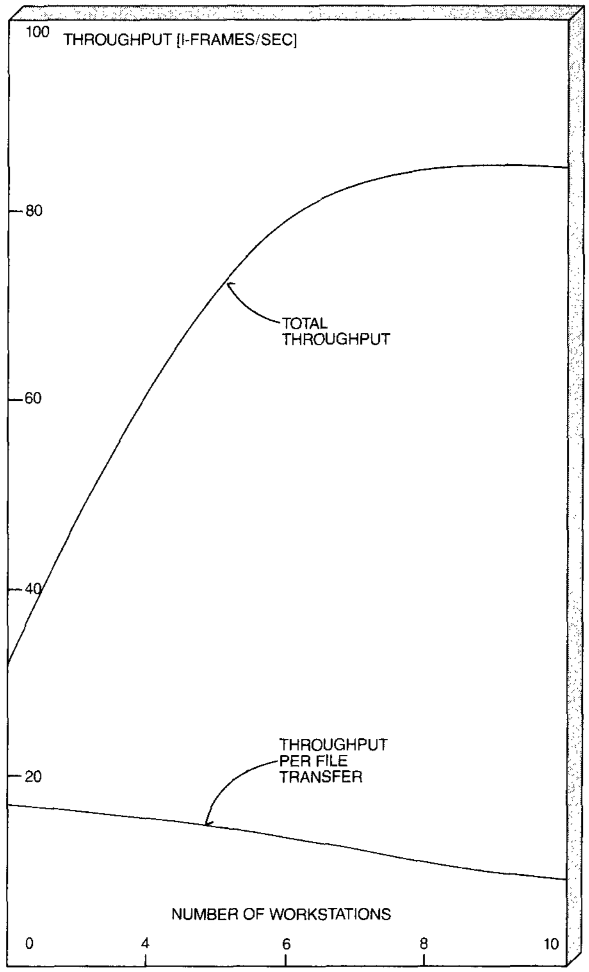
Workstations. From the modeling viewpoint, a workstation appears as a special case of a file server with only one file transfer.

It is assumed that both file server and workstations are always able to accept I-frames (i.e., remove them from the LLC receive buffers) and that all traffic sources always have a backlog of I-frames to be transmitted.

Frame lengths and buffer management. The length of I-frames is assumed to be constant and equal to the maximum frame length. This is motivated by our assumption of a permanent backlog of frames at the sources. Each frame is assumed to occupy a complete buffer in the user system or adapter. In the file server, separate sets of buffers are dedicated to the transmit and receive directions; both buffer sets are shared by all logical connections. Similarly, each adapter has two separate sets of transmit and receive buffers.

Results. The results subsequently presented are based on the following selection of parameter values: one million bits per second ring speed, two million bits per second effective system interface speed, 500 bytes constant I-frame length, and 20 microseconds set-up time at adapter processor. We shall refer to a logical connection for file transfer from file server to workstation as a "get-file transfer," and that from workstation to file server as a "put-file transfer." The scenario considered consists of a file server handling an equal number of get-file and put-file transfers. The mean time to process an I-frame (RR- or REJ-frame) at the file server is assumed to be 10 milliseconds (2 milliseconds). The corresponding values for a workstation are 50 and 10 milliseconds. Each adapter/user system has the same number of transmit and receive buffers.

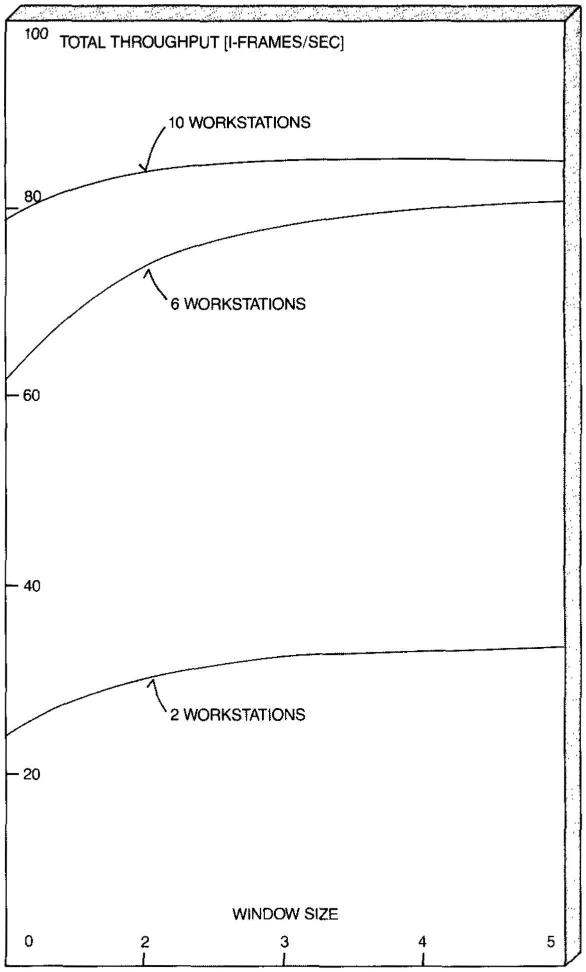
Figure 16 **Total throughput and throughput per file transfer versus number of workstations**



In Figure 16, we show the throughput per file transfer and the total throughput versus N, the number of workstations. The assumed window size is four. Each adapter has four send and four receive buffers. The number of send and receive buffers in the file server is equal to the product of window size and number of workstations; those in the workstations are equal to the window size.

The figure shows that for a small number of workstations, the total throughput increases roughly linearly with N. The reason is that, as long as N is small, the workstations are the bottleneck, and the addition of a workstation does not cause much interference at the file server. When N is large, the bottleneck is shifted from the workstations to the file server. The

Figure 17 Total throughput versus window size



processor of the file server is working at close to full capacity; increasing N does not result in an improvement in total throughput. Since the file-server processor is shared by the various file transfers, the throughput per file transfer is a decreasing function of N.

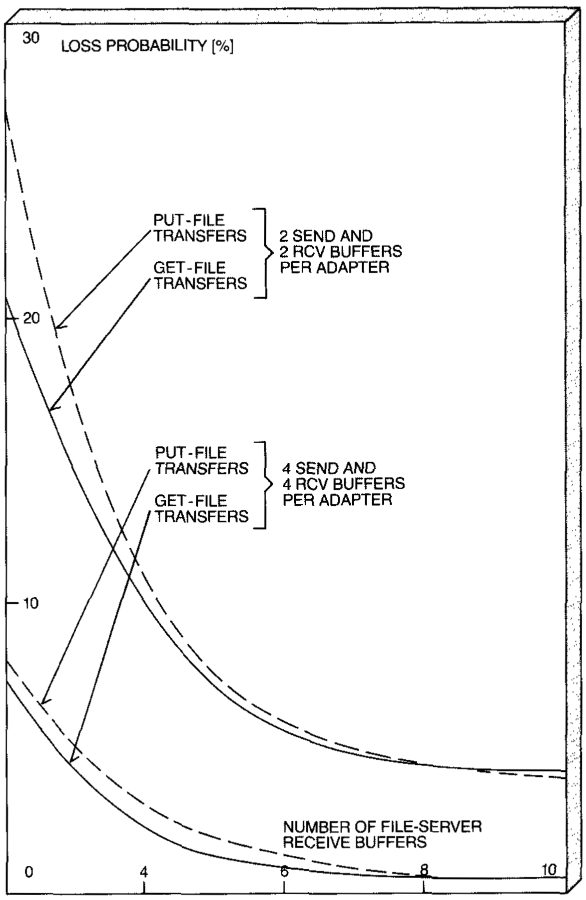
The ring is not heavily utilized; its utilization increases from 14 percent when $N = 2$ to 34 percent when $N = 10$. Also, the loss probability due to buffer shortage at the file-server adapter is less than 0.2 percent for all cases.

We next study the effect of window size W on the total throughput. In Figure 17, we show the total throughput for different values of W. Three cases are considered: $N = 2$, 6, and 10 workstations. The assumptions regarding the buffer sizes are identical to the ones underlying Figure 16.

For the case of small W, both the workstation and the file-server processors are not busy all the time. An increase in W (e.g., from one to two) therefore results in a noticeable improvement in total throughput. However, when W is larger, either the file server or the workstation processor is busy almost all the time; hence, increasing the window size does not cause an increase in total throughput.

Consider now the effect of file-server buffer size on performance. In Figure 18, we plot the loss probabilities at the file-server adapter versus the number of file-server receive buffers for a configuration with $N = 10$ workstations. The window size is four. Results are shown separately for put-file transfers (loss

Figure 18 Loss probability at file-server adapter versus number of file-server receive buffers

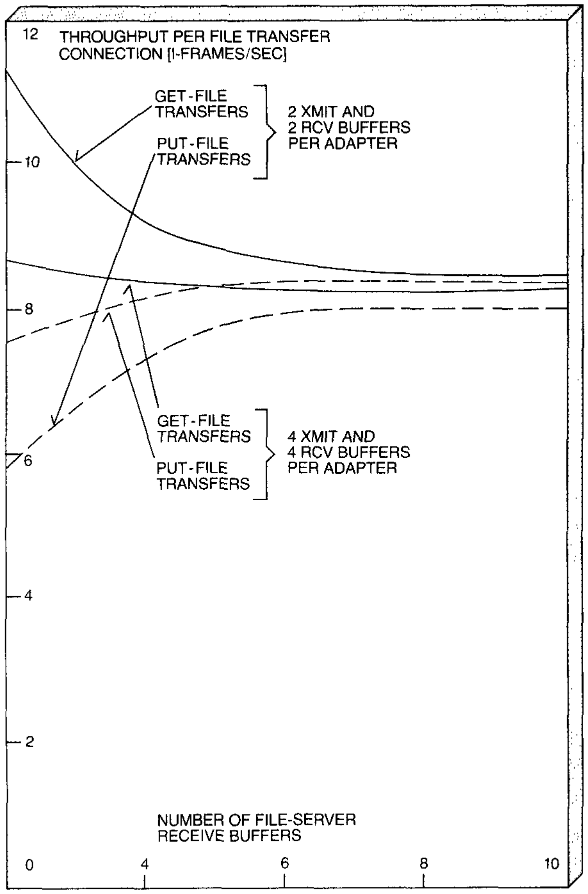


of I-frames) and get-file transfers (loss of RR- and REJ-frames). The results show that a significant fraction of frames is lost when the number of file-server receive buffers is small, but the loss probability decreases quickly with this number. The loss probability also decreases with an increasing number of buffers in the adapter, because in an overload situation, the adapter receive buffers function as an "extension" of the file-server receive buffers.

In view of the fact that the five workstations involved in put-file transfers may have a total number of 20 I-frames simultaneously outstanding, it is surprising that the loss probabilities are rather small already for six file-server receive buffers. This result can be explained as follows. In the implementation of the LLC protocol, each I-frame is separately acknowledged by an RR-frame. Since the file-server processor gives priority to frames received from its adapter, the preparation of frames for transmission is delayed. When the processor is ready to prepare an RR-frame for a put-file transfer, a number of I-frames for this connection may have been received, but only one of them is acknowledged. This RR-frame authorizes the workstation to transmit one I-frame only. It follows that the windows of the workstations and hence the arrival rate of I-frames to the file server are self-regulated.

Generally, high loss probabilities are an indication of insufficient receive buffers at the file server and its adapter. In other words, these receive buffers may have been over-sold to the various logical connections. Under this condition, it is of interest to study the effect of frame losses on throughput. Figure 19 shows the throughput per file transfer as a function of the number of file-server receive buffers for the same scenario as for the previous figure. The put-file transfers suffer significant degradation in throughput when the number of file-server receive buffers is small and hence the loss probability is high (cf. Figure 18). This is due to the fact that both REJECT and time-out recovery result in a delay period before an I-frame with the correct sequence number is retransmitted by the workstation. For lost I-frames and I-frames received out of sequence, no acknowledgments have to be generated (except for a REJ-frame generated when the first out-of-sequence I-frame has been received). Furthermore, out-of-sequence I-frames are not copied, and hence less time is required for processing. This results in more processing resources available to the get-file transfers, which therefore experience an improvement in throughput. Consequently, the total throughput is very insensitive to frame losses because the throughput degradation of the put-file transfers is compensated by the throughput increase of get-file transfers.

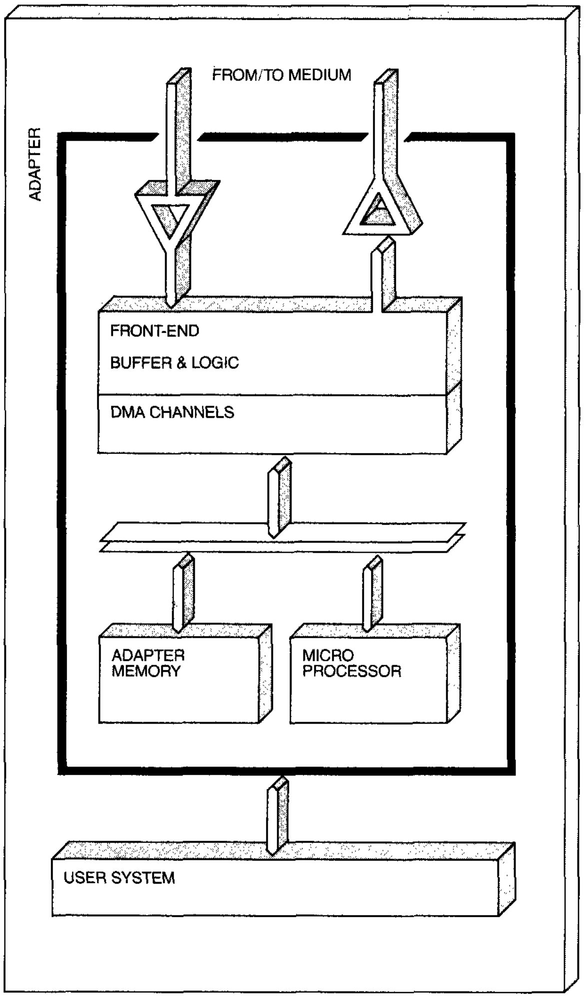
Figure 19 Throughput per file transfer versus number of file-server receive buffers



Local-area network adapter design

In a local-area network, user systems are attached to the transmission medium through network adapters, also called network controllers. An essential feature of an adapter is that it is able to receive frames arriving with no or very small gaps between them. If adapters were frequently unable to receive such frames, the performance of the local network—as seen by the user—would be unacceptable. Subsequently, we describe a study, the goal of which was to understand the timing problem associated with the reception of back-to-back frames.[68,69]

Figure 20 **Structure of local-area network adapter (from Ref. 68) ©1982 IEEE**



Adapter operation. The structure of the network adapter under consideration is shown in Figure 20. It contains the circuitry necessary to transmit data onto and receive data from the transmission medium and memory for buffering both outgoing and incoming frames. It also has one or more direct memory access (DMA) channels for data transfer between the transmission medium and the adapter memory. Furthermore, the adapter contains a processor that manages the frame buffers and DMA channel(s), and controls the interface to the user system.

When a frame is received, its destination address is compared with the address of the adapter to determine whether the frame is to be copied. If so, the frame is transferred into the adapter memory, provided a DMA channel has previously been set up by the processor. Arriving frames can get lost if no receive buffer is available or if a buffer is available but the processor was unable to set up a DMA early enough. At the end of each DMA transfer, an interrupt to the processor is generated. When servicing this interrupt, the processor searches for a free receive buffer and then sets up the DMA channel to receive into the acquired buffer.

The design goal is that a DMA channel is enabled when the first bit of a frame is received. Obviously, the chance of achieving this goal is higher, the smaller the DMA set-up time and the more DMA channels provided. In addition, one may employ a FIFO buffer at the adapter front-end to temporarily store incoming data in case no DMA channel is enabled. A further possibility to achieve zero (or very small) frame-loss probability is to define the medium access protocol in such a way that a minimum gap is guaranteed between subsequent frames. In this paper, we do not consider the latter possibility, although the analysis can be modified to cover this case.[69]

Data flow on transmit operations is essentially the reverse of the receive operation described above. Since our study concentrates on the most critical part of the adapter operation, namely, frame reception, we do not elaborate on details of the transmit operation.

An alternative to the adapter structure under consideration is a design where received frames are transferred (by DMA) directly into the user system memory without being buffered in the adapter. This, of course, places more constraints on the architecture and performance of the attaching station; an advantage is, however, that intermediate buffering is not needed in the adapter. It should be noted that in such a system, basically the same problem has to be solved regarding the reception of back-to-back frames. Again, a DMA channel must be enabled when the first bit of a frame needs to be buffered.

The model subsequently developed is oriented towards the adapter structure shown in Figure 20. However, the basic mechanism modeled is general enough for the analysis of this model also to be applicable to other adapter structures, e.g., one without buffers.

Performance model. The major assumptions underlying this study are as follows. Frame losses due to shortage of receive buffers are negligible, either because sufficient receive buffers are provided or because the frames received can be moved very rapidly to the user system. A fixed number of DMA channels is always dedicated to the receive direction. We shall restrict our discussion to the situation where a series of frames arrives back-to-back at an adapter. Upon arrival of the first frame, all DMA channels are assumed to be enabled.

The structure of our model is shown in Figure 21; its operation can be described as follows. When the first bit of a frame is to be copied, the state of the front-end buffer is checked; if the front-end buffer is not empty, the frame will be lost. Otherwise, two different situations may occur:

- At least one DMA channel is enabled: In this case, the frame is transferred via one of the enabled

Figure 21 Adapter performance model (from Ref. 68) ©1982 IEEE

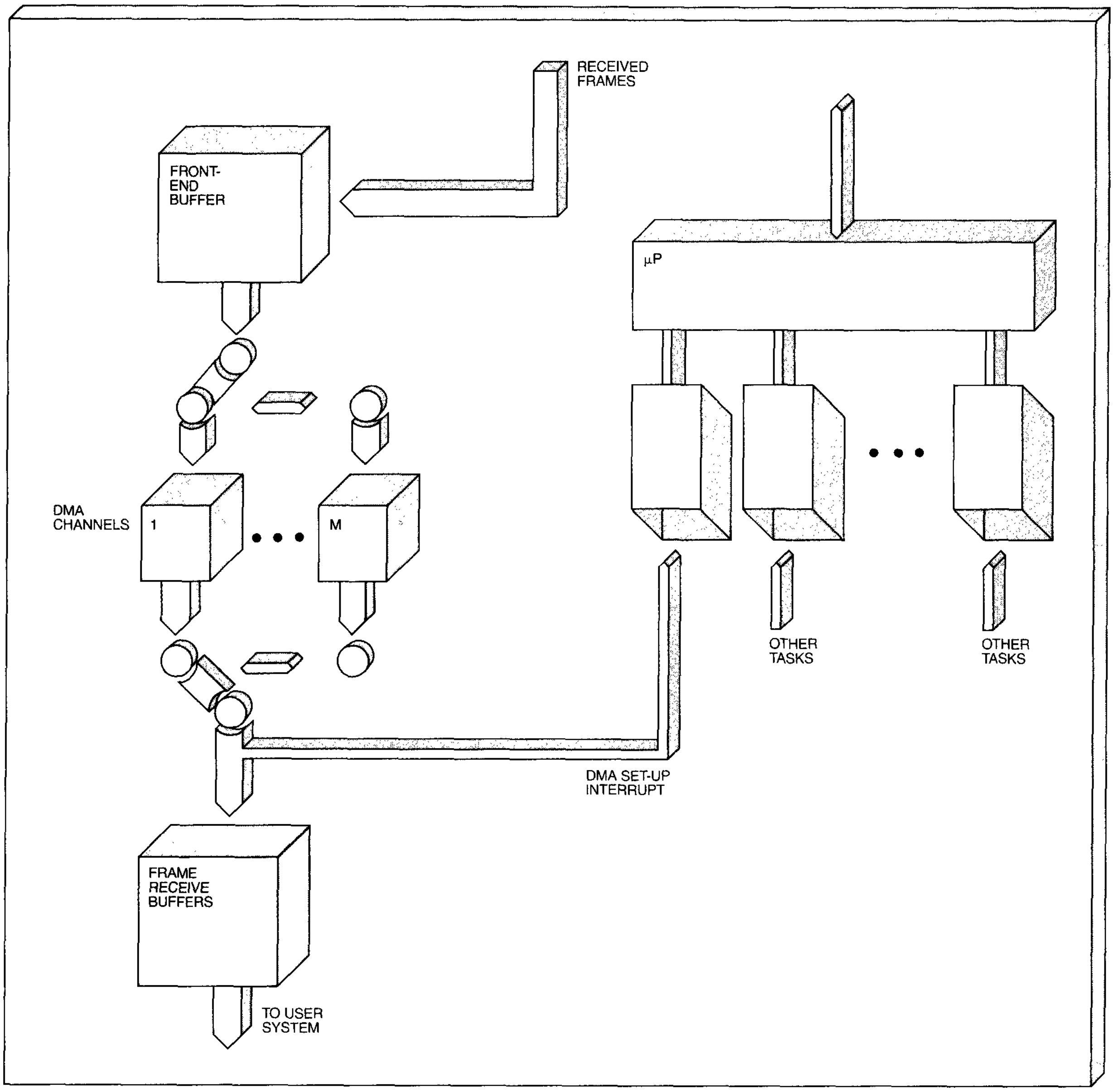


Table 1 Probabilities PB(n) (in percent) that nth back-to-back frame has been lost (one DMA channel; B is in bytes)

	Speed (million bits per second) 4		8		16	
	$B = 0$	$B = 10$	$B = 0$	$B = 20$	$B = 0$	$B = 40$
PB (1)	0	0	0	0	0	0
PB (2)	100	0	100	0	100	0
PB (3)	0	0	0	15.0	32.6	65.4
PB (4)	100	0	100	12.7	67.4	27.2
PB (5)	0	0	0	10.8	54.6	17.4
PB (6)	100	0	100	11.4	56.1	41.0
PB (7)	0	0	0	11.6	58.8	32.2
PB (8)	100	0	100	11.5	55.7	26.1

DMA channels into a receive buffer at medium transmission speed.

- No DMA channel is enabled: At medium transmission speed, the frame is written into the front-end buffer. If the front-end buffer is filled before a DMA channel is enabled, the frame will be lost. Otherwise, the newly enabled DMA channel will transfer the contents of the front-end buffer into a receive buffer at DMA channel speed, which is higher than the medium transmission speed. Once the front-end buffer has been emptied, the remainder of the frame is, of course, transferred at medium transmission speed.

An interrupt to the processor is generated at the end of the DMA operation. When servicing this interrupt, the processor acquires a free receive buffer and sets up the DMA channel with the starting address of this buffer. It is assumed that processing of the interrupt takes a constant time and that this interrupt has preemptive priority over the other processing tasks. Our model takes into account that, at interrupt generation time, the processor may still be busy processing an earlier interrupt of the same type, or—in an even worse situation—that previously generated interrupt requests from other DMA channels may still be waiting to be processed.

Analysis. It is relatively straightforward to determine conditions under which back-to-back frames are always successfully received (see Wong and Bux[68,69]). In practice, these conditions may not be met for reasons of hardware/software constraints or cost. If this is the case, knowledge of the probabilities of (a) losing the nth back-to-back frame and (b) being able to receive n back-to-back frames successfully will be very useful in designing an adapter. We subsequently outline how analytic results for these probabilities can be obtained.

The basic approach is to study the time-dependent behavior of a two-dimensional stochastic process $(i(t), j(t))$ defined as follows: $i(t)$ measures the occupancy of the front-end buffer at time t, expressed in terms of the time it takes to transfer the buffered data to the adapter memory at DMA speed; $j(t)$ is the total amount of unfinished work of the adapter processor (relevant to the DMA set-up task). Since the DMA set-up time is constant, the number of enabled/disabled DMA channels can be simply deduced from $j(t)$ at any point in time. Figure 22 shows a sample path of this process and the corresponding states of the DMA channels.

Define an "observation instant" to be a point in time immediately after the last bit of a frame has been copied, under the condition that a DMA channel is available. It is not difficult to see that the process $(i(t), j(t))$ possesses the Markov property[70] at these observation instants. Furthermore, one can determine whether or not a frame has been lost from the state of the process at the previous observation instant. We can therefore obtain answers to our basic performance questions if the state probabilities at the observation instants are known.

Details of the analysis are given in Reference 68; Reference 69 also describes efficient numerical algorithms to compute the relevant performance measures.

Results. For the subsequent results, the transmission speeds considered are 4, 8, and 16 million bits per second. The DMA channel speed D is assumed to be 32 million bits per second. Frame lengths are distributed according to the discrete analog of a truncated hyperexponential distribution;[69] the mean and coefficient of variation of this distribution are 100 bytes and 1.2, respectively. The choice of these values is motivated by the measurement data reported in Shoch and Hupp.[24]

Adapter model with one DMA channel. Consider first the case of one DMA channel. In Table 1, we show the probabilities PB(n) that the nth frame in a sequence of back-to-back frames is lost for a DMA set-up time of 20 microseconds and different medium transmission speeds. The front-end buffer size B is either zero or equal to the product of DMA set-up time T and ring transmission speed R.

Figure 22 Sample path of front-end buffer occupancy, adapter processor and DMA channel activities (from Ref. 68) ©1982 IEEE

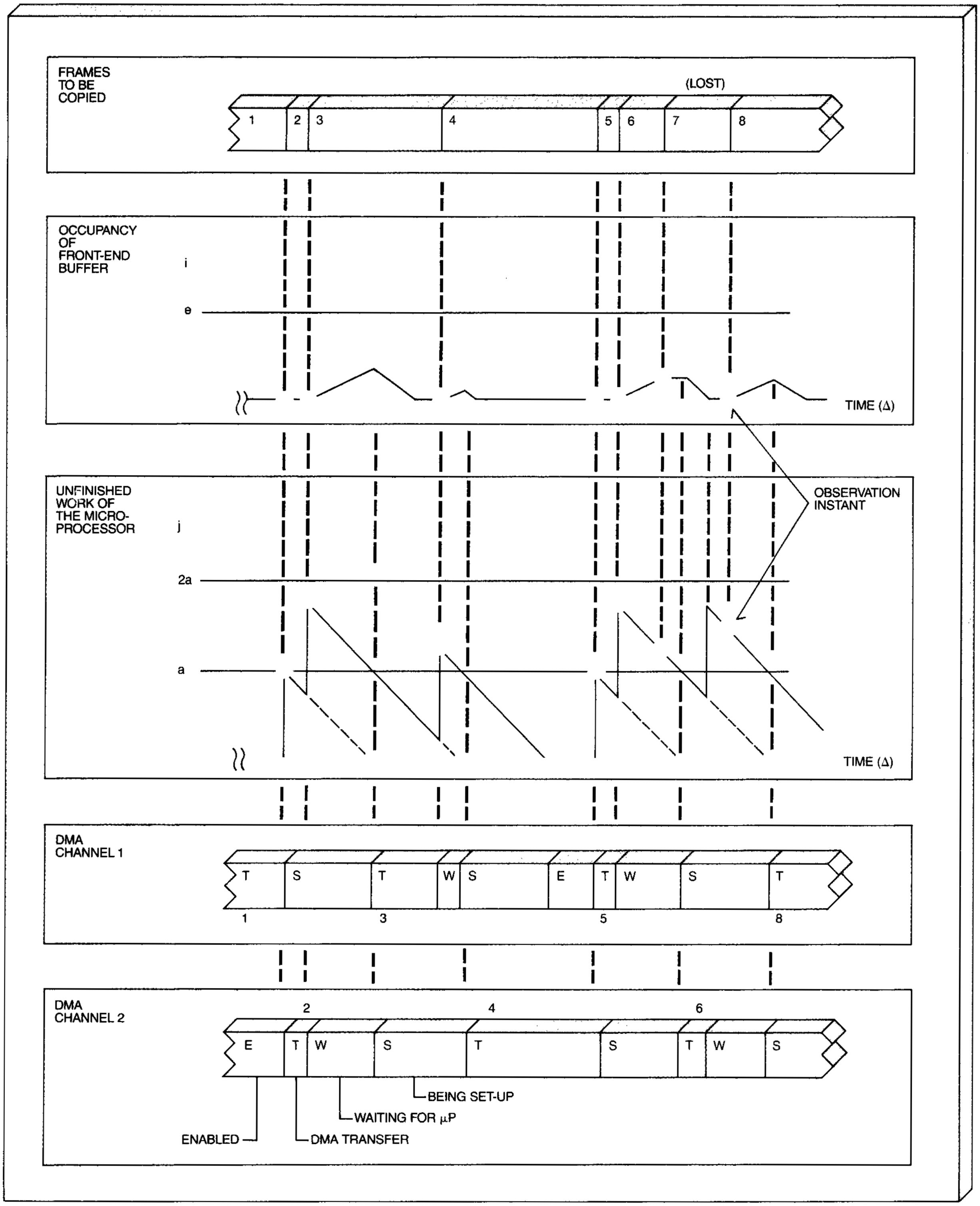


Table 2 Probability NB(n) (in percent) that first n back-to-back frames have been successfully received (one DMA channel; 16-million-bit-per-second transmission rate)

	DMA Setup Time (microseconds)				
	4	8	12	16	20
NB (1)	100.0	100.0	100.0	100.0	100.0
NB (2)	100.0	100.0	100.0	100.0	100.0
NB (3)	100.0	78.6	58.2	44.2	34.6
NB (4)	100.0	61.6	33.8	19.2	12.0
NB (5)	100.0	48.5	19.7	8.7	4.2
NB (6)	100.0	38.1	11.5	3.8	1.4
NB (7)	100.0	20.0	6.7	1.7	0.5
NB (8)	100.0	23.5	3.9	0.8	0.2

Table 3 Probability NB(n) (in percent) that first n back-to-back frames have been succcessfully received (16-million-bit-per-second transmission rate)

	DMA Channels 2	2	3
	No F/E buffer	40-byte F/E buffer	No F/E buffer
NB (1)	100	100	100
NB (2)	100	100	100
NB (3)	67.4	100	100
NB (4)	45.4	95.4	100
NB (5)	30.6	90.1	98.7
NB (6)	20.6	85.1	97.1
NB (7)	13.9	80.4	95.5
NB (8)	9.4	75.9	94.0

Apparently, the adapter performs poorly when front-end buffering is not employed. Every other frame is lost for medium speeds of 4 and 8 million bits per second. The results for the case of 16 million bits per second are different because more than one frame may arrive during DMA set-up.

If front-end buffering is employed, all frames are successfully received at the four-million-bit-per-second transmission speed. However, frame loss is observed when the speed is doubled. For the case of 16 million bits per second, the use of front-end buffering results in only a slight improvement in the loss probability. This is an indication that the DMA set-up process is too slow and the processor is the system bottleneck. This observation leads us to subsequently study the adapter performance as a function of the DMA set-up time.

In Table 2, we show the results for NB(n), the probability that the first n frames in a sequence of back-to-back frames have been successfully received for different values of the DMA set-up time T. The medium transmission speed is 16 million bits per second. The front-end buffer size is equal to $T * R$. All frames have been successfully received when $T = 4$ microseconds. For other values of T, frames may be lost, and the results in Table 2 show the performance degradation when T is increased.

Adapter model with two or more DMA channels. Finally, we consider the effectiveness of using more than one DMA channel to prevent loss of frames. For the case of a transmission speed of 16 million bits per second, Table 3 shows results for three different designs: (1) two DMA channels, no front-end buffer; (2) two DMA channels, 40-byte front-end buffer; (3) three DMA channels, no front-end buffer. The DMA set-up time is 20 microseconds.

We observe substantial improvements offered by the use of front-end buffering or the use of an additional DMA channel. The addition of an extra DMA channel is slightly more effective than front-end buffering in alleviating the timing problem associated with DMA setup. Any further increase in the number of DMA channels is not expected to improve performance significantly.

Acknowledgment

The last two major sections of this paper are based on joint work by J. W. Wong and the author. The token-ring simulation results in the second major section were produced by H. L. Truong. Both contributions are gratefully acknowledged. The author would also like to thank K. Kuemmerle for many helpful discussions and for reviewing the manuscript. Much of this paper also appears in *Lecture Notes in Computer Science* (Editors: D. Hutchinson, D. Shepherd, and J. Mariani), published by Springer-Verlag, Heidelberg, Germany, and is reprinted with their permission.

Cited references

1. D. D. Clark, K. T. Pogran, and D. P. Reed, "An introduction to local area networks," *Proceedings of the IEEE* **66**, 1497–1517 (1978).
2. R. C. Dixon, N. C. Strole, and J. D. Markov, "A token-ring network for local data communications," *IBM Systems Journal* **22**, Nos. 1 & 2, 47–62 (1983).
3. K. Kuemmerle and M. Reiser, "Local-area communication networks—An overview," *Journal of Telecommunication Networks* **1**, 349–370 (1982).
4. *ECMA TR/14: Local Area Networks Layers 1 to 4 Architecture and Protocols* (September 1982).

5. *Standard ECMA-80: Local Area Networks (CSMA/CD Baseband) Coaxial Cable System* (September 1982).
6. *Standard ECMA-81: Local Area Networks (CSMA/CD Baseband) Physical Layer* (September 1982).
7. *Standard ECMA-82: Local Area Networks (CSMA/CD Baseband) Link Layer* (September 1982).
8. *Standard ECMA-89: Local Area Networks Token Ring* (September 1983).
9. *Standard ECMA-90: Local Area Networks Token Bus (Broadband)* (September 1983).
10. *IEEE Standard 802.2, Logical Link Control.*
11. *IEEE Standard 802.3, CSMA/CD Access Method and Physical Layer Specifications.*
12. *IEEE Standard 802.4, Token-Passing Bus Access Method and Physical Layer Specifications.*
13. IEEE Project 802 Local Area Network Standards, *Draft IEEE Standard 802.5, Token Ring Access Method and Physical Layer Specifications,* Working Draft (February 1984).
14. International Organization for Standardization, *Data Processing—Open Systems Interconnection—Basic Reference Model,* Draft Proposal ISO/DP 7498 (February 1982).
15. R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed packet switching for local computer networks," *Communications of the ACM* **19**, 395–404 (1976).
16. *The Ethernet—A Local Area Network Data Link Layer and Physical Layer Specifications,* Version 1.0, Digital Equipment, Intel, and Xerox Corporations (September 30, 1980); available from Digital Equipment Corporation, Maynard, MA, from Intel Corporation, Santa Clara, CA, and from Xerox Corporation, Stamford, CT.
17. L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: Part I—Carrier sense multiple access modes and their throughput-delay characteristics," *IEEE Transactions on Communications* **COM-23**, 1400–1416 (1975).
18. F. A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part IV—Stability considerations and dynamic control in carrier sense multiple access modes," *IEEE Transactions on Communications* **COM-25**, 1103–1120 (1977).
19. G. T. Almes and E. D. Lazowska, "The behavior of Ethernet-like computer communications networks," *Proceedings of the 7th Symposium on Operating Systems Principles*, Asilomar Grounds, CA (1979), pp. 66–81.
20. E. Arthurs and B. W. Stuck, "A theoretical performance analysis of polling and carrier sense collision detection systems," *Proceedings of the IFIP Conference on Local Computer Networks*, North-Holland Publishing Company, Amsterdam (1982), pp. 415–438.
21. W. Bux, "Local-area subnetworks: A performance comparison," *IEEE Transactions on Communications* **COM-29**, No. 10, 1465–1473 (1981).
22. S. S. Lam, "A carrier sense multiple access protocol for local networks," *Computer Networks* **4**, 21–32 (1980).
23. F. A. Tobagi and V. B. Hunt, "Performance analysis of carrier sense multiple access with collision detection," *Computer Networks* **4**, 245–259 (1980).
24. J. F. Shoch and J. A. Hupp, "Measured performance of an Ethernet local network," *Communications of the ACM* **23**, 711–721 (1980).
25. D. J. Farber, J. Feldman, F. R. Heinrich, M. D. Hopwood, D. C. Loomis, and A. Rowe, "The Distributed Computer System," *Proceedings of the 7th IEEE Computer Society International Conference,* IEEE, Piscataway, NJ (1973), pp. 31–34.
26. W. D. Farmer and E. E. Newhall, "An experimental distributed switching system to handle bursty computer traffic," *Proceedings of the ACM Symposium on Problems in the Optimization of Data Communications,* Pine Mountain, GA (1963), pp. 31–34.
27. B. K. Penney and A. A. Baghdadi, "Survey of computer communications loop networks: Parts 1 and 2," *Computer Communications* **2**, 165–180 and 224–241 (1979).
28. W. Bux, F. Closs, P. A. Janson, K. Kuemmerle, and H. R. Mueller, "A reliable token-ring system for local-area communication," *Conference Record of NTC '81,* IEEE, Piscataway, NJ (1981), pp. A2.2.1–A2.2.6.
29. W. Bux, F. Closs, P. A. Janson, K. Kuemmerle, H. R. Mueller, and E. H. Rothauser, "A local-area communication network based on a reliable token-ring system," *Proceedings of the International Symposium on Local Computer Networks,* North-Holland Publishing Co., Amsterdam (1982), pp. 69–82.
30. J. H. Saltzer and K. T. Pogran, "A star-shaped ring network with high maintainability," *Computer Networks* **4**, 239–244 (1980).
31. A. K. Agrawala, J. R. Agre, and K. D. Gordon, "The slotted vs. the token-controlled ring: A comparative evaluation," *Proceedings of COMPSAC 1978*, Chicago (1978), pp. 674–679.
32. W. Bux and H. L. Truong, "Token-ring performance: Mean-delay approximation," *Proceedings of the 10th International Teletraffic Congress* (Montreal, Canada), North-Holland Publishing Company, Amsterdam (1983), pp. 3.1-3/1–3.1-3/6.
33. W. Bux and H. L. Truong, "Mean-delay approximation for cyclic-service queueing systems," *Performance Evaluation* **3**, 187–196 (August 1983).
34. R. B. Cooper, "Queues served in cyclic order: Waiting times," *Bell System Technical Journal* **49**, 399–413 (1970).
35. M. Eisenberg, "Queues with periodic service and changeover times," *Operations Research* **20**, 440–451 (1972).
36. O. Hashida, "Analysis of multiqueue," *Review of Electrical Communications Laboratories* **20**, 189–199 (1972).
37. J. F. Hayes, "Local distribution in computer communications," *IEEE Communications Magazine,* pp. 6–14 (March 1981).
38. A. R. Kaye, "Analysis of a distributed control loop for data transmission," *Proceedings of the Computer Communications, Networks, and Teletraffic Symposium,* Brooklyn Polytechnic, New York (1982), pp. 47–58.
39. A. G. Konheim and B. Meister, "Waiting lines and times in a system with polling," *Journal of the ACM* **21**, 470–490 (1974).
40. P. J. Kuehn, "Multiqueue systems with nonexhaustive cyclic service," *Bell System Technical Journal* **58**, 671–698 (1979).
41. I. Rubin and L. F. DeMoraes, "Polling schemes for local communication networks," *Conference Record ICC '81,* IEEE, Piscataway, NJ (1981), pp. 33.5.1–33.5.7.
42. G. B. Swartz, "Polling in a loop system," *Journal of the ACM* **27**, 42–59 (1980).
43. M. V. Wilkes and D. J. Wheeler, "The Cambridge Digital Communication Ring," *Proceedings of the Symposium on Local Area Communication Networks,* Boston (1979), pp. 47–61.
44. E. R. Hafner, Z. Nenadal, and M. Tschanz, "A digital loop communications system," *IEEE Transactions on Communications* **COM-22**, 877–881 (1974).
45. C. C. Reames and M. T. Liu, "A loop network for simultaneous transmission of variable length messages," *Proceedings of the 2nd Annual Symposium on Computer Architecture,* Houston, TX (1975), pp. 7–12.
46. L. Fratta, F. Borgonovo, and F. A. Tobagi, "The Express-net: A local area communication network integrating voice and data," *Proceedings of the International Conference on the Performance of Data Communication Systems and their Applications,* North-Holland Publishing Co., Amsterdam (1981), pp. 77–88.

47. J. O. Limb and C. Flores, "Description of Fasnet, a unidirectional local area communications network," *Bell System Technical Journal* **61**, 1413–1440 (September 1982).
48. E. H. Rothauser and D. Wild, "MLMA: A collision-free multiaccess method," *Proceedings of IFIP Congress 77*, North-Holland Publishing Co., Amsterdam (1977), pp. 431-436.
49. F. A. Tobagi, "Multiaccess protocols in packet communication systems," *IEEE Transactions on Communications* **COM-28**, 468–488 (1980).
50. J. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Transactions on Information Theory* **IT-25**, 505–515 (1979).
51. J. Capetanakis, "Generalized TDMA: The multi-accessing tree protocol," *IEEE Transactions on Communications* **COM-275**, 1476–1484 (1979).
52. J. F. Hayes, "An adaptive technique for local distribution," *IEEE Transactions on Communications* **COM-26**, 1178–1186 (1978).
53. W. M. Kiesel and P. J. Kuehn, "CSMA-CD-DR: A new multiaccess protocol for distributed systems," *Conference Record NTC '81*, IEEE, Piscataway, NJ (1981).
54. G. S. Blair, "A performance study of the Cambridge Ring," *Computer Networks* **6**, 13–20 (1982).
55. G. S. Blair and D. Shepherd, "A performance comparison of Ethernet and the Cambridge Digital Communication Ring," *Computer Networks* **6**, 105–113 (1982).
56. W. Bux and M. Schlatter, "An approximate method for the performance analysis of buffer insertion rings," *IEEE Transactions on Communications* **COM-31**, 50–55 (1983).
57. M. T. Liu, G. Babic, and R. Pardo, "Traffic analysis of the distributed loop computer network (DLCN)," *Proceedings of the National Telecommunications Conference*, IEEE, Piscataway, NJ (1977), pp. 31:5-1–31:5-7.
58. A. Thomasian and H. Kanakia, "Performance study of loop networks using buffer insertion," *Computer Networks* **3**, 419–425 (1979).
59. W. Bux, "Analysis of a local-area bus system with controlled access," *IEEE Transactions on Computers* **C-32**, 760–763 (1983).
60. M. Fine and F. A. Tobagi, "Performance of round-robin schemes in unidirectional broadcast local networks," *Proceedings of the ICC '82*, IEEE, Piscataway, NJ (1982), pp. 1C.5.1–1C.5.6.
61. V. C. Hamacher and G. S. Shedler, "Access response on a collision-free local bus network," *Computer Networks* **6**, 93–103 (1982).
62. O. Spaniol, "Modelling of local computer networks," *Computer Networks* **3**, 315–326 (1979).
63. O. Spaniol, "Analysis and performance evaluation of HYPERCHANNEL access protocols," *Proceedings of the 2nd International Conference on Distributed Computing Systems*, IEEE, Piscataway, NJ (1981), pp. 247–255.
64. F. A. Tobagi and R. Rom, "Efficient round-robin and priority schemes for unidirectional broadcast systems," *Proceedings of IFIP WG 6.4 International Workshop on Local Area Networks*, Boston (1979), pp. 47–61.
65. *Standard ECMA-72: Transport Protocol*, 2nd Revision (September 1982).
66. International Organization for Standardization, *Data Processing—Open Systems Interconnection—Connection Oriented Transport Protocol Specification*, Draft Proposal ISO/DP 8073 (June 1982).
67. J. W. Wong and W. Bux, *Performance Evaluation of Logical Link Control in a Local-Area Ring Network*, to appear as an IBM Research Report (1984).
68. J. W. Wong and W. Bux, "Analytic modeling of an adapter to local-area networks," *Proceedings of GLOBECOM 82*, Miami, IEEE, Piscataway, NJ (1982), pp. C.2.3.1–C.2.3.6.
69. J. W. Wong and W. Bux, *Analytic Modeling of an Adapter to Local-Area Networks*, Research Report RZ-1175, IBM Corporation, Research Division, P.O. Box 218, Yorktown Heights, NY 10598.
70. L. Kleinrock, *Queueing Systems*, Vols. I and II, John Wiley & Sons, Inc., New York (1976).

Reprint Order No. G321-5230.

Werner Bux *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Dr. Bux joined the IBM Zurich Research Laboratory as a Research Staff Member in 1979. He is currently on assignment to the Research Center in Yorktown Heights. His work for the past several years has been in the area of architecture and performance evaluation of local-area networks. He has also been involved in local network standardization through his work in TC24 of the European Computer Manufacturers Association (ECMA). Dr. Bux received the M.S. and Ph.D. degrees in electrical engineering from the University of Stuttgart, Stuttgart, West Germany, in 1974 and 1980, respectively.