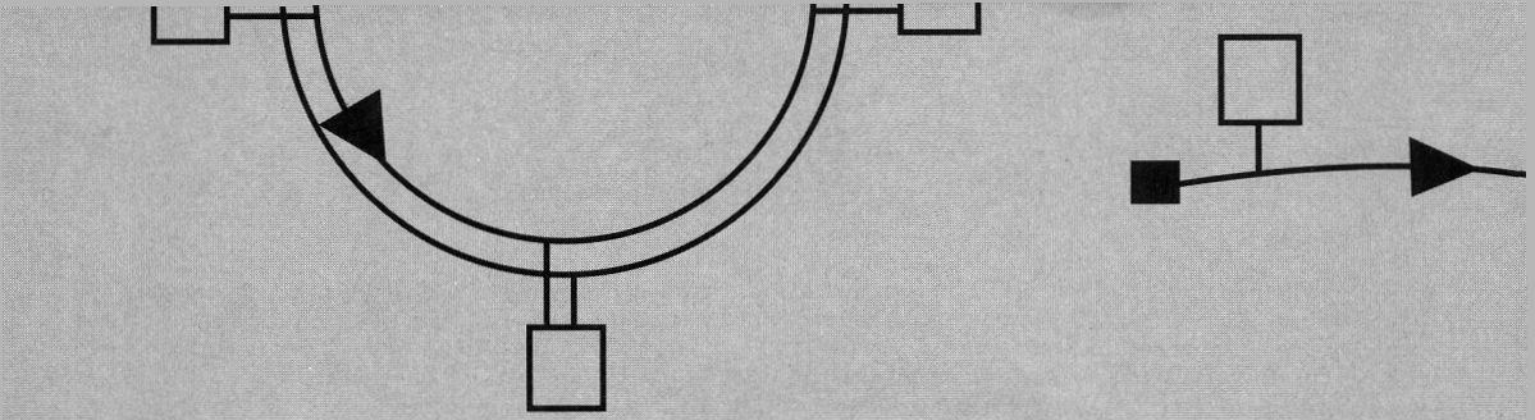
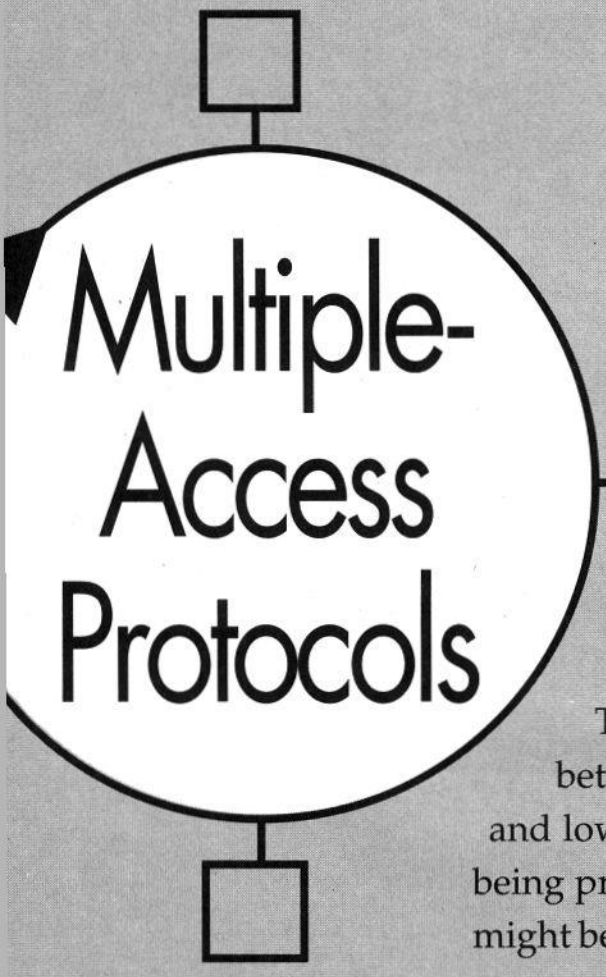
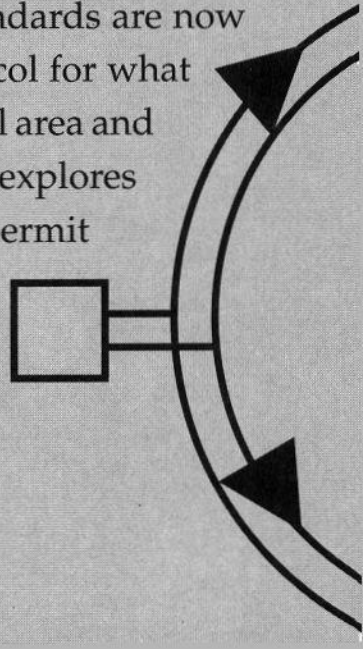


Multiple-Access Protocols for High-Speed Networks

BY
JOHN O. LIMB

Optical fibers have the potential to carry tens and even hundreds of Gbits/sec of information. The question of how to share these large capacities between many users through an efficient, flexible, and low cost network is a challenge. Standards are now being proposed that will decide the protocol for what might be called the third generation of local area and metropolitan area networks. This article explores the principal classes of protocols that permit efficient sharing of high-speed channels.





ETHERNET

Perhaps the most familiar multiple-access (or shared-access) network is Ethernet. Users share in the total transmission capacity of the network, but only one station can access the transmission medium at any one time; an access protocol is employed to provide orderly and fair access to the network. For small periods of time, an individual user can exploit the total capacity of the shared link. This is in contrast to a centralized switch in which each user is connected to the switch by a dedicated line.

In Ethernet (Fig. 1a), packets flow along the medium (coaxial cable) in both directions from the point at which the user connects to the medium. A user gains control of the medium and may transmit for a period of time before relinquishing the medium to another user. Control of the medium is established during a contention period that is related to the time it takes for a signal to propagate from one end of the bus to the other and back (round-trip delay).

As long as the contention period is short relative to the period of time that the station transmits, the protocol is reasonably efficient. This is the case with Ethernet, which was designed to run over a coaxial cable at a transmission rate of 10 Mbits/sec. Optical fibers, on the other hand, can transmit at speeds well in excess of a Gbit/sec, at which an Ethernet type protocol is no longer efficient. At such high speeds, typical messages (up to a maximum of about 2 kilobytes/sec) become short relative to the round-trip delay time required to resolve contentions. For example, consider a Gbit/sec network five kilometers in length—a packet of average length 2 kilobytes could be transmitted in approximately one-third of the round-trip delay time.

The solution to achieving more efficient use of the medium is to permit traffic to flow in only one direction on the medium. It is then possible to line up packets one behind the other with virtually no gap between them, thus permitting the medium to be efficiently used. If unidirectional transmission is used, two broad classes of network topology are possible: dual buses and rings. In a dual bus (Fig. 1b), the upper bus is used if a station transmits information to a station on its right, whereas the lower bus is used to transmit information to stations on the left. In a ring (Fig. 1c), information is passed from station to station in one direction around the ring. A second ring (Fig. 1d) transmitting information in the opposite direction can also be added to provide some level of redundancy, should the primary ring fail. Alternatively, the second ring can also carry traffic, so if one ring fails the total capacity of the system decreases to one half. Redundancy can also be provided for the dual bus by bending the buses into an open ring so that the ends are co-located at one station (Fig. 1e). Should the bus break, the ends are joined and the stations on either side of the break form the new ends.

This paper concentrates primarily on the dual bus,[1] although with just minor changes it is usually possible to adapt protocols developed for the bus to the ring. So let us look a little further at the dual bus (Fig. 2a). A user message produced at a station may be of any length and, for transmission, is usually broken up into a number of short, equal-length cells or frames. The stations at the ends

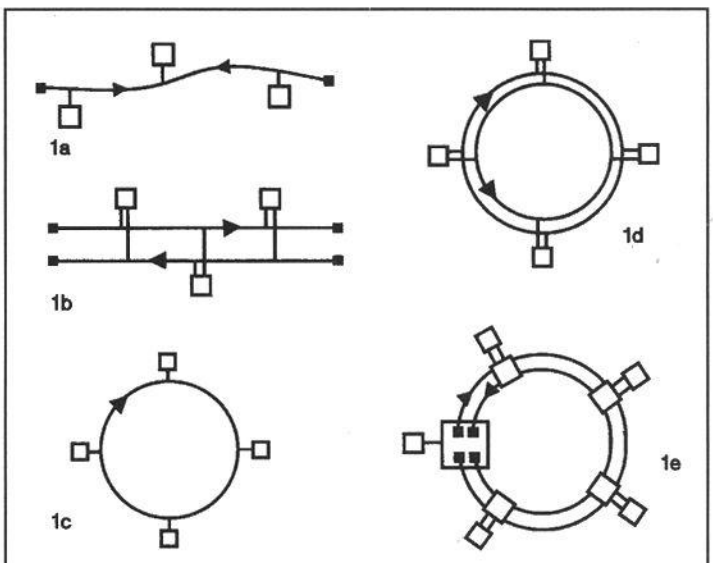


Figure 1. Various LAN/MAN topologies. (a) Ethernet, where energy flows along the bus in both directions. In both dual bus (b) and ring (c), energy only flows along a bus in one direction. A second ring may be added (d) to provide redundancy if a station or bus fails. Redundancy can also be added to the dual bus configuration (e).

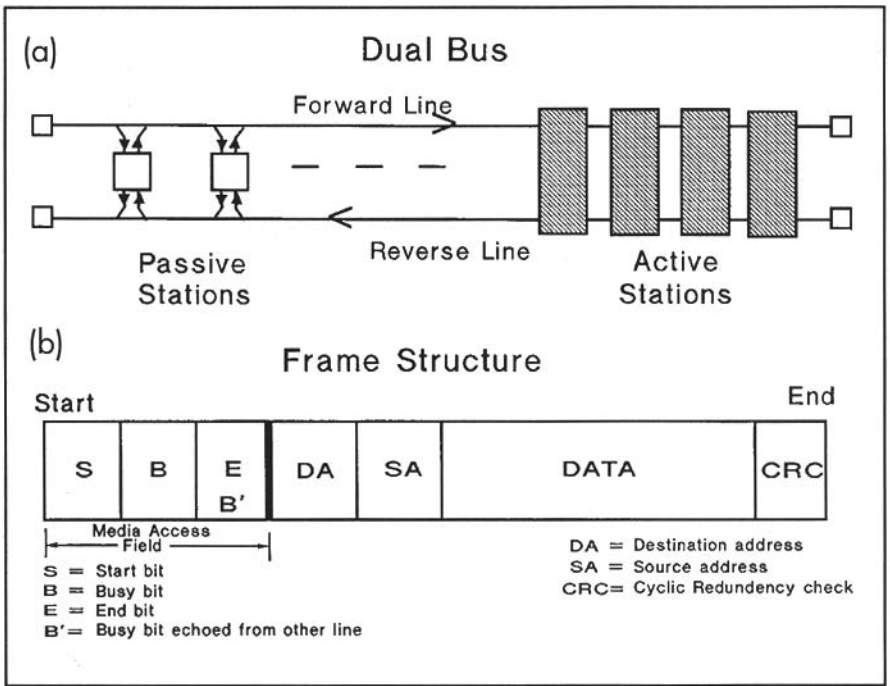


Figure 2. (a) The dual bus may have stations connect passively as shown on the left end of the bus (via directional couples) or actively, as shown on the right end. (b) Simplified structure of a frame or cell showing the media access field.

of the bus have an additional housekeeping function to perform. They send timing information to the individual stations to indicate the start of a fixed-length slot in which cells or frames may be transmitted. In addition, they may be involved in other functions required to implement a particular multiple-access protocol. If we consider optical transmission media, a station may attach passively to the buses (as shown in stations to the left of the line in Fig. 2a), in which case a small amount of power is tapped from the bus to read the incoming packet and power is added to the bus to write a new frame. Alternatively, the station may make an active attachment (stations to the right of the line in Fig. 2a), in which case a bus segment is terminated at the station and the signal is regenerated and transmitted on the next bus segment.

Note that if each station transmits an equal amount of traffic to all other stations, a station at the end of the line will transmit all of its traffic on one line and receive traffic on the other, whereas a station in the middle of the line would transmit equally on both lines. Thus, if we consider one line only, under this assumption the traffic produced by a station decreases linearly from one end to the other. Since the operation of each bus on a dual bus network is identical, we will consider the action of only one of the buses in what follows. We refer to the bus carrying the data as the forward line and the other bus, which may carry control information for the forward line, as the reverse line.

Let us first consider the desirable properties we would expect of a multiple-access protocol and discuss ways to measure its performance. We will then consider a number of protocols, starting with some very simple ones, to develop insight into what makes a good protocol. We will conclude with a comparison of the protocols.

CRITERIA AND PERFORMANCE

The typical cell or frame structure shown in Figure 2b consists of a media-access field having a bit that indicates whether a slot is occupied or not (busy bit) and other fields that control access to the network. In addition, there are source and destination address fields, then a data field or payload, and finally an error detection field to indicate whether or not the frame has been corrupted. The frame header is overhead as far as the user is concerned, but we will ignore this in calculating the network utilization or load. Thus, for a slotted ring or dual bus we will define the utilization on the system as the fraction of all slots that are filled with traffic. Most protocols will perform well under light load; that is, frames submitted to the network will be transmitted with little delay. It is the performance under heavy load and overload that determines how well a protocol performs.

The capacity of a network is the maximum utilization that the network can achieve. The capacity is often given as a measure of performance; however, in some systems the capacity may approach "1" arbitrarily closely, though the delay that a frame experiences in accessing the network could be very large. A more useful measure is the delay that a frame experiences as a function of network utilization. A typical curve of delay as a function of utilization is shown in Figure 3a; we will return to this later. Also important is the behavior of the network under overload conditions, since the offered load on a typical network may frequently exceed one (*i.e.*, overload situation) for periods of time. It is desirable that all stations continue to obtain their share of the capacity of the network under this condition. Ideally, a small number of users generating a large amount of traffic should be able to gain a significant fraction of the total capacity of the network, but at the same time, a heavily loaded station should not freeze out other stations with less demand.

A fair network would be one in which each station receives the same quality of service. This is rarely the case with a dual bus. Most protocols will give better service to stations nearer the start of the line. Fairness is not usually a criterion of importance to a user. It is more important that a network provide the level of service guaranteed by the supplier. This could be defined in terms of the average access delay a station experiences. Thus, instead of attempting to design a fair network, it is more useful to design a network that minimizes the average delay experienced by the station receiving the *poorest* performance. The particular model of traffic used to test a protocol is also

important. Smooth traffic is much more easily handled than traffic that arrives in bursts.

Let us look at a couple of cases to clarify the above points. The graph in Figure 3a was obtained by simulating the operation of a dual-bus network. The details of the protocol are not important at this point. The average time a frame waits in a queue before being transmitted is shown as a function of the network utilization. Each point in the figure is the result of a simulation. Each simulation was run for 300,000 frames with the statistics being gathered after the first 100,000 frames. The lower curve shows the queuing delay averaged over all stations. As can be seen, the resulting curve is quite smooth. The upper curve shows the average queuing delay for the station with the greatest queuing delay. The maximum average delay is the more useful measurement as it defines a guaranteed level of service available to any station. At the same time, it accounts for any unfairness in the system. A problem with using this measure is that, since it is an extremal statistic, the results obtained are much less smooth. Furthermore, the maximum average delay (or worst-case average delay) will change with the number of the samples (*i.e.*, the length of the simulation); equal length simulations should be used in comparing results.

The effects of different types of traffic are illustrated in Figure 3b. If messages are assumed to consist of one fixed-length segment and they arrive according to a Poisson distribution, the worst-case delay is given by the lower curve. The performance of the protocol with this type of traffic is extremely good; if a station could tolerate a delay of 200 slots (corresponding to 200 microseconds in this example), the network could be operated with a load up to 0.94. If, however, the traffic arrives in bursts, the upper curve results. In this case, the traffic model assumes that messages are still arriving with a Poisson distribution, but that each message is either one frame or 16 frames in length. We assume that short messages are four times more probable than long messages. While this model of computer traffic is very simple, it is a reasonable starting point.[2] With the same maximum tolerable delay of 200 frames, it would now be possible to load the network to 0.80. Thus, it can be seen that the particular model of traffic assumed will greatly influence the amount of traffic that a network can support. To make meaningful comparisons between different protocols, it is important to adopt realistic simulation parameters and to use the same parameters in each case.

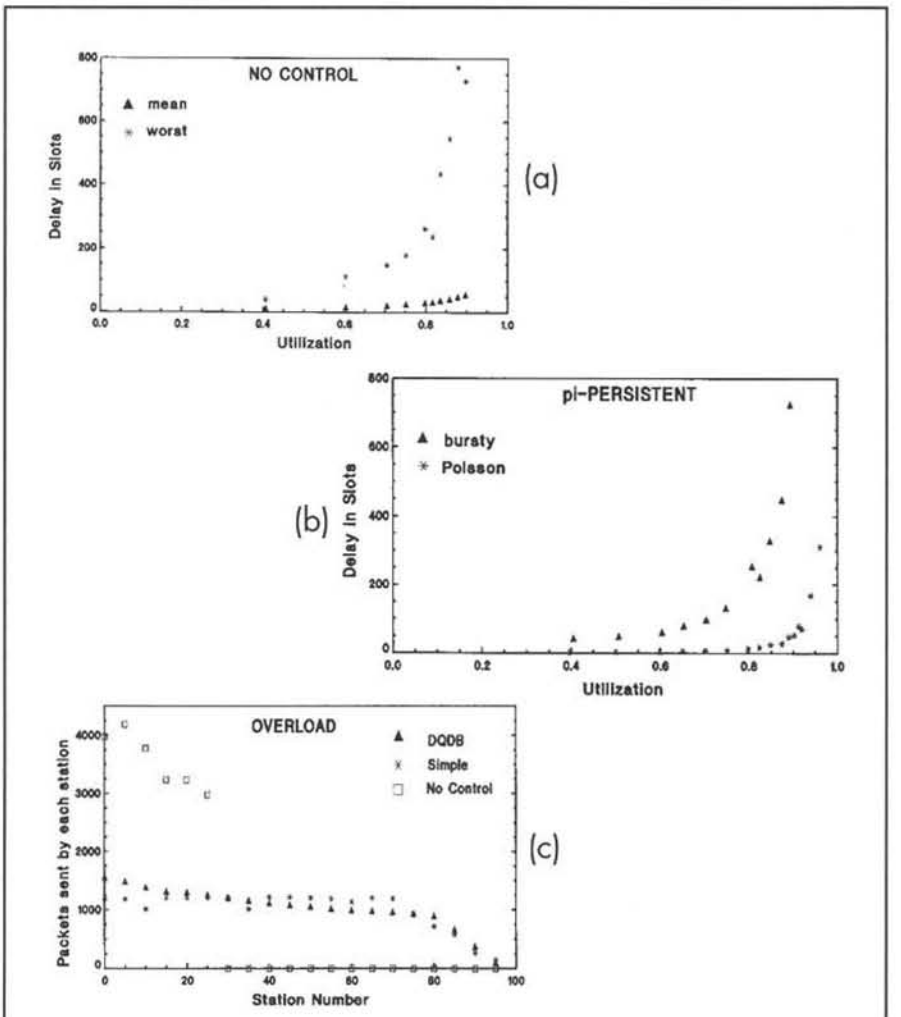


Figure 3. (a) Queuing delay vs. network utilization for the dual bus with bursty traffic where each station seizes the first empty slot ("no control"). The mean delay experienced by all stations is small and the results obtained are smooth compared to the results obtained by plotting the greatest mean delay experienced by any station (worst). (b) Queuing delay vs. network utilization for the dual bus with both bursty and Poisson traffic for the p-persistent protocol. The type of traffic greatly affects the performance. (c) Overload performance of three protocols. The offered load is twice as great as the network can carry. For DQDB and the "Simple" protocol, all stations gain good access to the network. With "no control," stations 30-99 are denied access.

Four protocols

We now look at four protocols in order of increasing complexity and simulate their performance under identical conditions. As before, let us assume the network is 5 kilometers from end to end (10 kilometers round trip), that it operates at 1 Gbit/sec, and that slots are 1000 bits long. Thus, expressed in slot-lengths the network is 50 slots long (assuming that signals travel at a speed of 0.2 km per microsecond). We also assume that the traffic is bursty, as described above, and that stations are randomly distributed along the length of the network.

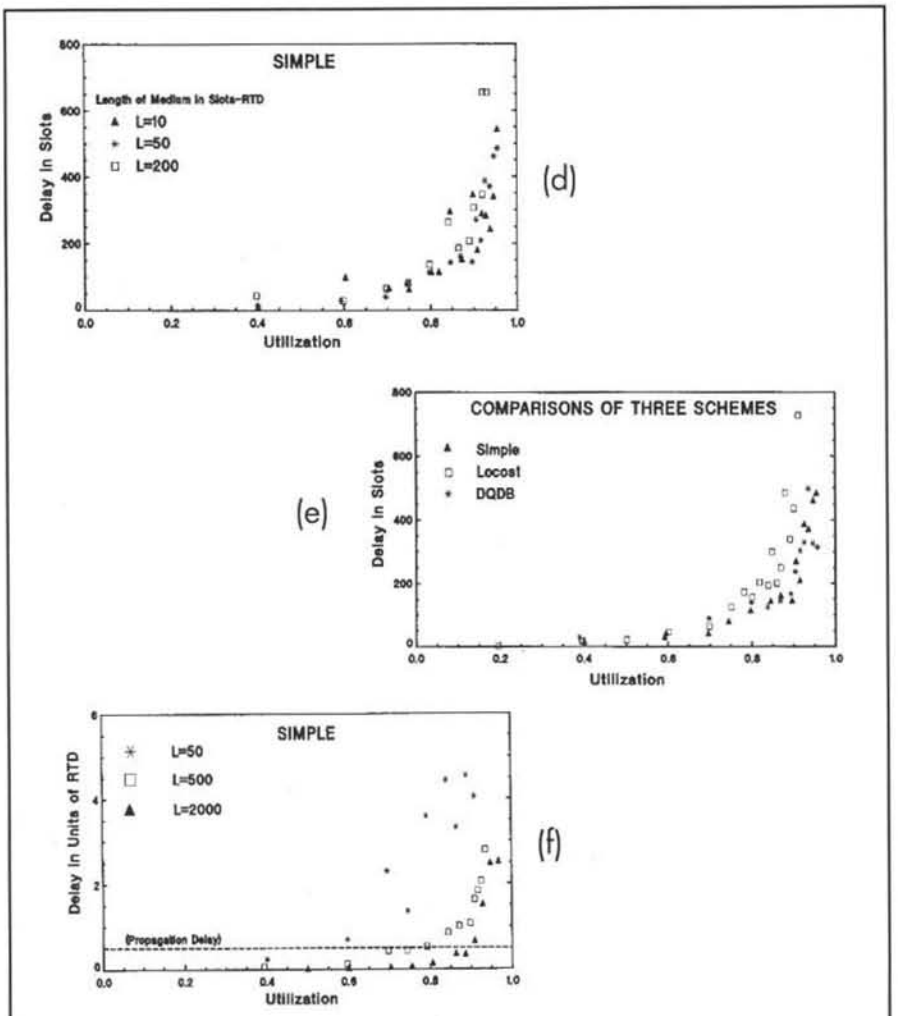


Figure 3 (cont.). (d) Queuing delay vs. network utilization for the dual bus with bursty traffic for the "Simple" protocol. Results are shown for three different lengths: 10, 50, and 200 slots round-trip delay. Performance is slightly worse for a length of 200 slots than for 10 and 50 slots. (e) Comparison of queuing delay vs. network utilization for the dual bus with bursty traffic for three different protocols. The performance of "Simple" and DQDB are similar. LOCOST is a little worse. (f) Queuing delay vs. network utilization for the dual bus with bursty traffic for "Simple." The queuing delay is expressed in units of round-trip delay. For very long networks, the delay is small relative to the propagation delay except at very high loads.

No control

Let us assume that the protocol consists simply of allowing each station with a frame awaiting transmission to transmit it in the next arriving empty slot.[3] Such a protocol favors stations situated toward the start of a line since these stations will have the first opportunity to access a slot. However, as shown in Figure 3a, the protocol performs well. The problem is that downstream stations will be prevented from transmitting if the offered load exceeds unity for any length of time. Since practical networks will almost surely suffer periods of overload, this protocol must be judged inadequate. The number of frames transmitted by a station in overload (offered load = 2.0) is shown in Figure 3c; stations after number 29 are frozen out. Note, however, that the protocol has the nice property of being distance-independent, meaning that the performance of the protocol does not change with the length of the line.

The p_i-persistent protocol

This protocol is also simple to describe:[4] a station with a frame to transmit will transmit in an empty slot with a probability p_i, where the subscript denotes the number of the station. It is possible to calculate p_i such that the average queuing delays experienced at each station are equalized.[4] The consequence is that all stations (except the second to last and the last) allow some empty slots to pass, giving downstream stations the opportunity to transmit. The queuing delay under steady-state conditions is shown in Figure 3b. The queuing delay is very similar to the results obtained with the "no control" protocol and, in overload conditions, stations will be able to transmit frames in proportion to their generated traffic.

The calculation of the optimum p_i is simple, but it does require knowledge of the load generated by other stations, and, of course, loads will change dynamically. The p_i can be calculated dynamically by observing the traffic from each station on the reverse line,[5] but for the dual-bus topology this will require more than just reflecting the busy-bit on the reverse line (refer to the next protocol). The address of the source station would also be required. This protocol is also distance independent.

The "Simple" protocol

On a global level this protocol operates in cycles:[1] when a cycle starts, stations transmit as many frames as they have queued, up to a maximum, and then a restart mechanism operates to begin a new cycle. This protocol requires the addition of another control bit in the access field besides the busy-bit (B). This additional bit is referred to as the busy prime bit (B′) and is generated by the last station by copying the value of B in an arriving frame on the forward line into the B′ field on the next departing frame on the reverse line (see Fig. 2b). Thus, an end-of-cycle is indicated by an empty slot arriving at the end station (B = 0). This indication is propagated back to all stations by means of the reverse line, using the B′ = 0.

The protocol for a station consists of monitoring the reverse line for the first occurrence of a B′ that is set to zero. The station then sets a counter (the "P" counter) to a value of P_{max}, the maximum number of frames a station

is permitted to send in a cycle. It then continues to transmit frames until P_{max} frames have been transmitted or another $B' = 0$ is encountered. In this case, the counter is reset to P_{max} and the cycle starts again. The best value of P_{max} will depend on the length of the line and the number of stations, but only weakly. For the conditions assumed for our simulations, a value of $P_{max} = 16$ is appropriate. Values of 8 or 32 reduce performance only marginally.

The protocol becomes less efficient as the length of the line increases because of the time taken for an empty frame to propagate down one line and return as a $B' = 0$ frame on the reverse line. For stations close to the downstream end of the line, this period is short; for stations close to the start of the line, the period approaches the round-trip delay time. Some frames will be transmitted during this period by newly arriving traffic at stations that have not exhausted their frame allocation for that cycle. Figure 3d shows the performance of the "simple" protocol for batch traffic under three different round-trip delays: 20 slots, 50 slots, and 200 slots (corresponding to a round-trip length of 40 km). As would be expected, the performance at 200 slots round-trip is slightly worse than at 20 or 50 slots round-trip. In overload, all stations continue to gain access to the network (Fig. 3c) and, as the overload increases, the amount of traffic transmitted by each station approaches equality. Should the number of users suddenly drop to a few, they will not succeed in exploiting all the available capacity. The protocol has been modified to overcome this problem.[6]

The distributed queue dual-bus (DQDB) protocol

The DQDB protocol[7] also requires an additional one bit field in the access field of the frame and is referred to as the "request bit." Consider a frame of a message that has arrived at the head of a queue and is ready to be transmitted. The station sets the request bit to one in the access field of the next arriving slot on the reverse line of the dual-bus. This request travels upstream and increments a counter in each station (referred to as the "request counter"). At the same time, this counter is being decremented every time an empty slot passes a station on the forward line. Thus, the request counter can be thought of as measuring the number of downstream stations with one or more frames to transmit for which an empty slot on the forward line has not been allocated (the request counter can not be less than zero). As the station sets the request bit on the reverse line, the count on the request counter is transmitted to a count-down counter and the request counter is then reset and begins again counting newly arriving requests. The station then allows a number of empty packets to pass the station equal to the value of the count-down counter. The frame can then be transmitted and the cycle is repeated for the next frame.

If it were not for the delay experienced by frames traveling on the network, the above described mechanism would dictate that frames arriving at all stations are transmitted in the order in which they arrive at the head of queue—thus, the notion of a distributed queue. The protocol works efficiently under most conditions (Fig. 3e). However, when there are a small number of users, such as two or three, the capacity seized by each user can vary greatly depending upon the length of line between the users and the timing of the service requests by the users. A technique for overcoming this deficiency has been proposed, called "bandwidth balancing."[8] (Since it is similar to the protocol in the next section, it will not be described.) Another problem with the protocol is that a single station with a very large load can severely degrade the performance of other stations.[9] The protocol works well under overload conditions behaving similarly to the "Simple" (Fig. 3c). This protocol is also distance independent.

Load controlled scheduling of traffic (LOCOST)[11]

As with the "Simple," the busy-bit on the forward line is echoed on the reverse line as B'. This enables a station anywhere on the line to measure the total traffic on the forward line by observing B' on the reverse line. Each station then adjusts the rate at which it transmits traffic on the forward line, based on the measured utilization, so as to hold the utilization at some target value. The algorithm used to adjust the transmission rate in response to the measured utilization should respond quickly to load changes, but at the same time be stable. The protocol requires some unused capacity to operate. In practice, a target utilization of 0.95 is achievable and the target can be held within a couple of percent rms if the offered load is large enough to consistently exceed the target utilization. The performance is shown in Figure 3e and is very similar to the other protocols. It behaves well under overload conditions. When only one or two stations are active, the algorithm will adapt to permit them to seize most of the available capacity.

The speed with which the algorithm can adapt to a rapid change in load is important. Simulations show that a new quiescent state is achieved in a worst-case time of 10-15 round-trip delays. The control algorithm can be adjusted to distribute the total capacity between stations in an arbitrary manner. In addition, capacity can be allocated between different classes of traffic (*e.g.*, data, video, voice) if a separate code is used in the access field for each class. Thus, the capacity of a link can be allocated between different types of traffic or classes of use in a flexible manner.[10]

PERSPECTIVE ON THE PROTOCOLS

We have described four rather different multiple-access protocols for high-speed networks. Let us compare and contrast their performance.

1. All schemes have similar and very adequate delay performance. The p_i-persistent and LOCOST protocols are marginally worse than the simple and DQDB.
2. Of the four protocols, the Simple is the only one that has distance-dependent performance. Up to a length of about 200 slots, even this protocol shows little change. As the line length increases beyond 200 slots, the propagation delay starts to exceed the queuing delay so that queuing delay becomes less and less important. This is illustrated in Figure 3f, where delay in units of round-trip delay is plotted against utilization. For a length of 2000 slots (400 km) and for a utilization of

up to 0.8, eliminating access delay altogether would have little effect as it is small relative to the round-trip delay.
3. There is some advantage in having bursts of frames or cells from one message transmitted in a batch—reassembly of the packet is somewhat simplified. None of the four protocols can guarantee that the frames are transmitted contiguously. One way to achieve this is to use a cycling protocol like Fasnet[1], but this protocol is more distance-dependent. For example, for a round-trip delay of 50 slots the maximum load would be 0.7 if a maximum delay of 200 slots were tolerated.
4. At very high speeds, there is an added incentive to keep the protocol simple to simplify the implementation. A simple protocol can also lead to simplified management and control. None of the protocols described is complex. Simple and DQDB can be easily implemented in dedicated logic. Both pi-persistent and LOCOST require periodic calculations, but this need only be performed in microseconds rather than nanoseconds.
5. Provisions for high priority or isochronous traffic can be built into all of the above protocols (*e.g.*, see Ref. 1). Priority classes are perhaps most easily accommodated in LOCOST because stations are measuring and controlling the transmitted traffic directly.

In this short article it is not possible to cover all classes of protocols. Two other classes are central-reservation schemes and buffer-insertion schemes. Central reservation provides high utilization and a lot of flexibility. The downside is that the protocols are more complex and greater latency occurs due to the reservation and central scheduling process.

In the schemes described above, once slots are written into they are not reused. Utilization can be significantly increased by reusing slots after they have reached their destination. For a uniform distribution of traffic, this technique typically doubles the capacity of the line; with a bidirectional ring, the capacity can be quadrupled. Buffer-insertion schemes typically employ reuse of a slot or packet and consequently they are very efficient. They also permit slots from a station to be transmitted in a burst, or variable-length messages can be transmitted. They are more expensive to implement and control since each station must now be prepared to buffer a frame or a maximum-length packet.

Toward third generation networks

First generation LANs working in the 10 Mbit/sec range are now widely deployed and we are on the threshold of the second generation of LAN and WAN evolution in the form of FDDI and DQDB operating in the 100 Mbit/sec range. The third generation of LANs and WANs, operating in the Gbit/sec range, is now being contemplated by standards bodies, and protocols like those described in this paper form the basis of these deliberations. At these speeds, widespread deployment of fiber for the delivery of the service will be mandatory. Many factors outside of the protocol will determine the appropriate system. Such factors include reliability, ease of fault location and isolation, ability to support multi-media services, scalability from low to high speeds, scalability in number of users, and cost. These factors will be blended with performance considerations in deciding the next and future generations of LANs and WANs.

John O. Limb *is a laboratory director with Hewlett-Packard Laboratories, Palo Alto, Calif.*

References

1. J.O. Limb and C. Flores, "Description of Fasnet—a unidirectional local area communication network," BSTJ **61**, 1982, 1413-1440.
2. J.F. Shoch and J.A. Hupp, "Measured performance of an Ethernet local network," Commun. Assoc. Comput. Mach. **23**, 1980, 711-721.
3. W. Dobosiewicz and P. Gburzynski, "On the apparent unfairness of a capacity-1 protocol for very fast local-area networks," Third IEE Conference on Telecommunications, Edinburgh, Scotland, March 1991.
4. B. Mukherjee and J.S. Meditch, "The pi-persistent protocol for unidirectional broadcast bus networks, IEEE Trans. on Commun. **36**, 1988, 1277-1295.
5. B. Mukherjee *et al.*, "Dynamic control and accuracy of the pi-persistent protocol using channel feedback," IEEE Transactions on Communications **39**, 1991, 887-897.
6. S. Tohme and G. Watson, "A performance analysis of S++: A MAC protocol for high speed networks," 3rd IFIP WG6.1/6.4 Workshop on Protocols for High-Speed Networks, Stockholm, Sweden, May 1992.
7. Z.L. Budrikis *et al.*, "QPSX: A queue packet and synchronous circuit exchange," Proc, 8th Int. Conf. on Comp. Comm., Munich, FRG, Sept. 15-19, 1986, 288-293.
8. E.L. Hahne *et al.*, "Improving the fairness of DQDB networks," Proc., IEEE INFOCOM '90, San Francisco, Calif., June 1990, 175-184.
9. J.O. Limb, "A simple multiple access protocol for metropolitan area networks," Proc. ACM SIGCOMM '90, Philadelphia, Pa, **20**, 1990, 69-78.
10. J.O. Limb, "Load-controlled scheduling of traffic on high-speed metropolitan area networks," IEEE Trans. on Commun. **37**, 1989, 1144-1150.