Tokenless Protocols for Fiber Optic Local Area Networks

PAULO RODRIGUES, LUIGI FRATTA, MEMBER, IEEE, AND MARIO GERLA, MEMBER, IEEE

Abstract—A family of LAN (Local Area Network) protocols is presented. The LAN consists of a pair of unidirectional fiber optic buses to which stations are connected via passive taps. The protocols provide round-robin bounded delay access to all stations. Contrary to most round-robin access schemes, the protocols do not require transmission of special packets (tokens); rather, they simply rely on the detection of bus activity at each station. The performance of these protocols in various traffic conditions and system configurations is evaluated via analysis and simulation.

I. INTRODUCTION

RECENT years have witnessed a rapid growth in local area communications needs due to the increase in user demands and the emergence of new applications such as real-time voice and video, high-speed printers, and graphics. LAN designers are being urged to provide systems able to handle very high throughputs among users several kilometers apart, while still satisfying delay constraints which become severe when real-time traffic is involved. A suitable choice of transmission medium, topology, and access protocol is necessary to meet these demands.

Among the possible transmission media, optical fiber appears to be the most cost-effective and promising technology. Fiber offers high bandwidth, immunity against electrical and magnetic interference, and protection against signal leakage. Other attractive features are its ease of installation due to weight and small size, and low attenuation compared to coaxial cable.

Fiber optics topologies can be configured in three basic ways: ring, star, and bus. In the ring, access is generally provided by a token scheme. Ring implementations require active repeaters and substantial logic working at channel speed (e.g., address recognition and flag setting) at each station. Cost and reliability, in spite of some fail-safe node proposals [1], may pose limits to the use of ring interfaces at very high speed. Furthermore, special procedures are needed to maintain synchronization, to recover from token loss or duplication, and to avoid unrecognized packets circulating endlessly in the ring.

Star topologies can be built by connecting several stations via an optical coupler. Star-configured networks have been implemented using either passive [6] or active [7] components. Most star implementations use the CSMA-CD access protocol which was successfully used in ETHERNET. Unfortunately, CSMA-CD performance becomes very poor when end-to-end propagation delay on the bus is comparable to (or larger than) packet transmission time [9].

Bus topology is the third alternative. Since optical fibers are unidirectional in nature, a unidirectional-bus system (UBS) must be used. In UBS, each station is connected to the bus and signals propagate in only one direction. To achieve broadcast communication two separate buses are needed where signals propagate in opposite directions, or the same bus must be folded in order to visit all stations twice. In recent years, extensive research has been conducted on UBS local area networks and some efficient protocols have been proposed. Expressnet [2] is based on a folded unidirectional bus and achieves conflict-free transmissions and bounded delay by means of a completely distributed implicit token protocol. A simple modification of Expressnet, D-net [10] uses central control to regenerate the token at each cycle.

U-net [3], Buzz-net [4], and Fasnet [5] are examples of UBS local area networks based on the twin-bus architecture. U-net and Buzz-net are asynchronous networks which use special patterns to implement the signaling scheme to control the channel. The need to recognize different transmission patterns may cause difficulties in the implementation. Fasnet is a synchronous slotted network, where slot generation and clock synchronization are the responsibility of the physical end stations. In Fasnet, stations must maintain bit synchronization with the bus at all times.

All the above mentioned systems can be implemented on either optical fiber or coaxial cable. When optical fibers are used, insertion loss is the major limitation to the maximum number of stations, N, which can be passively connected to the network. In view of this constraint, the twin-bus architecture is preferable to the folded-bus architecture since each segment must support N (instead of 2N) passive taps. Furthermore, the twin bus presents fewer techni-
This paper presents a family of protocols suitable for twin unidirectional-bus architectures. These protocols are innovative because they provide round-robin access to active stations (as any token-based scheme does), without using a token. In fact, the only required control feature at the interface is the ability to detect bus activity. For this reason the protocols are named Tokenless.

The basic operating principles of the Tokenless protocols are given in Section II, while details of different versions are described in Section III. Recovery and joining procedures are described in Section IV. Analytical and simulation results are presented in Sections V and VI.

II. PRINCIPLES OF OPERATION

The tokenless protocol (TLP) runs on the twin-bus architecture shown in Fig. 1. Stations are connected to each bus via two passive taps, a receive tap and a transmit tap. Stations receive packets and monitor channel activity through the receive tap. Specifically, the receive tap can observe presence or absence of activity (i.e., data) and detect events such as EOA (end of activity) and BOA (beginning of activity).

The transmit tap transmits (data) packets or an activity signal (AS). The activity signal keeps the downstream part of the channel busy. Its implementation (modulated or unmodulated carrier, random bits, continuous sequence of 1's, etc.) can be chosen according to the low-level encoding utilized for transmission on the channel.

A maximum reaction delay $d$ is assumed between the time a station senses EOA on one bus and the time it can start transmission on either bus. Likewise, there is a maximum delay $d$ between the sensing of activity from an upstream station and the interruption of an ongoing transmission. Moreover, an activity burst of duration $d$ is the minimum amount of energy reliably detected at any interface. The actual value of the parameter $d$ depends on the speed and transmission delays of the detection logics in the hardware implementation. Laboratory experiments show that detection of activity in optical fibers can be done reliably in nanosecond intervals.

A transmitting station always defers to an upstream transmission by aborting its own. The upstream transmission proceeds with only the first few bits corrupted by deferrals, regardless of the number of downstream stations attempting to transmit. If the preamble is sufficiently long, this feature guarantees that a packet which has been completely transmitted by a station is correctly received by all (downstream) stations.

Apart from the corruption of the first few bits in the preamble, the TLP protocols are generally collision free. The only exception is one version of the protocol, TLP-4, where, as we shall see, conflict may occur when an idle station becomes active and attempts to acquire channel synchronization. The conflict, however, is resolved in finite time. For this case, we assume that an interface detects a packet even when the packet is immediately preceded by a truncated transmission. The underlying assumption is that the beginning-of-packet flag cannot be replicated within the packet data nor contained in the activity signal described above. Flags can be implemented as reserved bit patterns (in which case bit stuffing is required to preserve data transparency), or as code violations on the bit encoding level.

The goal of the protocol is to guarantee round-robin transmissions among all back-logged stations, and to achieve good throughput/delay performance for a variety of traffic conditions and station placement. Furthermore, the need to detect special packets (e.g., tokens) is avoided, and control is completely distributed. These goals are achieved by EOA events propagating in the two buses alternatively. EOA events can be viewed as virtual tokens that allow stations to transmit packets in a round-robin fashion. One advantage of controlling the channel only through EOA events is the simple, reliable, and low-cost implementation even at very high speeds. Another advantage is the easy implementation of initialization and recovery procedures.

III. THE PROTOCOL

The protocol basically consists of five procedures. Each of these procedures has a specifically defined purpose and is represented by a set of states in the protocol's state diagram. The probing procedure enables a station to recognize its turn to transmit in a round. The election procedure enables a station to determine whether it is an extreme (leftmost or rightmost active) station. The round-restart procedure enables an extreme station to initiate a round in the reverse direction. The recovery procedure provides recovery when illegal events are detected. The initialization procedure enables a newly active station to synchronize with other active stations, if any, or through the recovery procedure, to initialize the round-robin cycle in an empty net. An active station is a station that is neither idle nor powered off.

Different parameters and options may be chosen when specifying the full protocol. Before exploring the details of the different implementations, the common foundation of the various versions of tokenless protocol is presented below.

A. Basic Tokenless Protocol

In describing the basic protocol, $A$ is a variable designating one channel and $A$ designates the opposite channel. Events on channel $A$ are indicated by $EVENT(A)$.
The first procedure, or probing, enables a station to probe the end of the train of packets (which has been forming on one bus) and to append its packet to the train. Assume channel $A$ has been sensed busy by backlogged station $S_i$. If $EOA(A)$ occurs, $S_i$ starts transmitting an activity signal on channel $A$. Otherwise, after time-out $d$, $S_i$ assumes that it has found the end of the train and it starts packet transmission on both channels.

The above procedure prevents station $S_i$ from starting transmission on channel $A$ during an interpacket gap, before the end of the train. In fact, if a burst of activity were sent on channel $A$ during an interpacket gap, a collision with an upstream (with respect to channel $A$) transmitting station may occur. With the proposed procedure, actual packet transmission starts on both channels only when the end of a train of packets is detected. Prior to that, a signal is transmitted only on channel $A$. Thus, only the first few bits of the incoming packet on channel $A$ may be corrupted.

After completing the transmission of a packet, station $S_i$ sets time-out TES (extreme station) and continuously transmits the activity signal on channel $A$ until either a $BOA(A)$ is detected or time-out TES occurs. If $BOA(A)$ is detected, $S_i$ cancels time-out TES and repeats the probing procedure with $A$ and $A$ reversed. If TES is reached, $S_i$ realizes it is an extreme station and executes the round-restart procedure. The round-restart procedure enables the extreme station to initiate a round in the opposite direction. Different versions of TLP take slightly different actions at the end of a round. The details are discussed separately for each TLP version.

If both channels are initially idle, the initialization procedure is invoked. Namely, each station $S_i$ sets time-out ND (Network Dead) and waits for the first of two events: $BOA(A)$ on either channel or time-out ND. If $BOA(A)$ occurs on channel $A$, $S_i$ cancels time-out ND and performs as if channel $A$ were initially sensed busy. Alternatively, if time-out ND occurs (no other station is active in the network), $S_i$ begins the recovery procedure to initialize the idle network.

The recovery procedure is also invoked during normal operation when illegal events are detected on the channels. Illegal events may be symptomatic of a temporary malfunction in one interface, or may be caused by a station out of synchronism. A thorough discussion of recovery and joining procedures is given in Section IV.

### B. Various Implementations

The several ways to specify round restart, initialization, and to choose parameters ND and TES lead to different versions of TLP. Two versions, TLP-1 and TLP-3, require all powered-on stations to be active in the network. TLP-2 and TLP-4 require only backlogged stations to be active. Moreover, TLP-3 and TLP-4 use additional status variables to improve performance. All versions are completely distributed, follow the basic protocol described in the previous section, and use the same recovery procedure.

These four versions, TLP-1, TLP-2, TLP-3, and TLP-4, constitute the family of tokenless protocols.

1) Definitions: A station is idle if it is in the IDLE state. A station that is neither idle nor powered off is called an active station. In TLP-1, and TLP-3, a powered-on station is always active. In TLP-2 and TLP-4, an idle station only becomes active when a packet backlog is formed.

Variable $A$ denotes the channel where $EOA$ is expected, or where the station is currently transmitting the activity signal. Channel $A$ is called the synchronizing channel. The identity of $A$ changes during the execution of the protocol and is assigned value 0 or 1 depending on whether the synchronizing channel is, respectively, channel $L$ to $R$ (left to Right) or channel $R$ to $L$.

Parameter $R = 2\tau + 2d$ is fundamental in the implementation of the protocols. $\tau$ is the end-to-end propagation delay. $R$ may be interpreted as the interval of time needed for $EOA$ to be propagated from one end station to the opposite end station ($\tau$), detected at the latter station and regenerated as a $BOA$ on the other channel (reaction delay $d$), propagated back to the former station ($\tau$), and finally detected ($d$).

The $EOA$ signal at the end of a “train” can be viewed as a virtual token circulating on the bus between end stations. A station physically located inside the present sweep of the virtual token is called an inside station. Similarly, a station physically located outside the present sweep of the virtual token is called an outside station. An outside station may be an idle station (in TLP-2 or TLP-4) or a station which has just been powered on.

2) TLP-1: TLP-1 is described below in detail. This description includes background information which also pertains to TLP-2, TLP-3, and TLP-4. The state diagram shown in Fig. 2 defines the TLP-1 operation.

States ON and I at the left of the figure represent the initialization procedure. Also at the left, states R1 and R2 represent the recovery procedure. $R$ is a pseudostate which simplifies the drawing of state transitions into recovery. States WFT, TTT, ST, and TXP represent the probing and transmission procedures. State ES executes the round-restart procedure.

A station enters ON only when it is powered-on. If one of the channels is busy (i.e., bus is busy, or $BIB$ in Fig. 2) $A$ is set to the value of that channel (0 or 1) and the state moves to WFT (Wait For Token) where the station waits for the synchronizing signal $EOA(A)$ as described in Section II. If both channels are idle (i.e., Both Buses Idle, or $BBI$), $S_i$ sets time-out $ND = R + d$ and moves to state I. $ND$ guarantees that if any station is active in the network it will be heard before any other action is taken. The reason for setting $ND$ to the given value will be clear after the round-restart procedure is explained. If activity is sensed in a channel ($BOA$ detection) before time-out $ND$ is reached, $A$ is set to the corresponding channel, and the station leaves the initialization procedure moving to state WFT. Otherwise, when ND is reached, the recovery procedure is executed by states R1 and R2. In the state diagram, the dotted lines which converge toward R1 (through R) repre-
sent transitions due to illegal events. The recovery procedure is standard for all versions of TLP and will be explained in Section IV.

States WFT, TTT, ST, and TXP allow a station to identify its turn and transmit a backlogged packet at the proper time. When in WFT, the station waits for EOA(A) as described in Section III-A. If there is a backlogged packet in the station, detection of EOA(A) causes a move to state TTT (Try To Transmit) after setting time-out d. In TTT the station transmits activity signal AS for an interval d. The purpose of TTT is to detect the end of a train of packets with an interpacket gap of, at most, d. If no BOA(A) is detected before time-out d expires, the state moves to TXP (Transmit Packet) and the first packet in the queue is transmitted on both channels. At the end of packet transmission, (i.e., ETX), the state moves to ES (End-Station Detection mode) and the activity signal is transmitted on channel R. If BOA(A) was detected while in TTT, the state moves back to WFT.

If the station has no backlog, it moves from WFT to ST (Set Time-Out). State ST performs as TTT except that no packet transmission occurs. Consequently, the state changes directly from ST to ES, without passing through TXP.

The round-restart procedure is performed in state ES. In this state, the station transmits the activity signal in the channel opposite of the channel where the virtual token is propagating. If the station senses BOA(A), then it concludes it is not an end station and moves to state WFT. If the station does not sense a BOA(A) within the interval R, then it realizes it is an extreme station. If the extreme station has a backlogged packet, it moves to TXP immediately. Otherwise, it remains in state ES and behaves accordingly after inverting the identity of the synchronizing channel.

Observe that if only one station is active in the network, then the periods of activity in either channel are separated by idle intervals of duration R. Also note that any new active station must wait for $ND = R + d$ seconds in state I before starting recovery. Thus, the new joining station will be automatically synchronized with the network if there is another active station since $ND > R$.

An example of the operation of TLP-1 for a network with 10 stations is given in the space-time diagram shown in Fig. 3. The time intervals, A, B, C, D, and E represent rounds. In round A, the virtual token propagates from left to right, stations 1, 3, 4, 5, 7, 8, 9, and 10 are powered on, and stations 7 and 8 transmit packets. In round B, the virtual token propagates from right to left, station 6 is powered on, and stations 6, 4, and 1 transmit packets. Rounds C, D, and E are similar.

TLP-1's greatest advantage is simplicity. Only the first station which finds the network dead must execute the initialization procedure. All other stations detect activity when they come alive and gracefully join the set of active stations. Ease of network joining is a consequence of time-out R enforced at the end of each round. However, performance is impaired by this extra overhead.

Performance also degrades under other special circumstances. In TLP-1 a powered-on station always performs activity on the channel even if the station has no

Fig. 2. TLP-1 state diagram.

Fig. 3. TLP-1 Space–time diagram.
TLP-2 is preferred to TLP-1. The virtual-token sweep is confined only to the span of the network covering the stations involved in transmission. When the active stations are confined to a small area, as in TLP-2, the unnecessary delay because the virtual token must sweep the entire bus, rather than only the section of the bus containing the stations involved in transmission.

In an attempt to remove some of the TLP-1 limitations other versions of the protocol have been investigated. These versions are described below.

3) **TLP-2**: In TLP-2, the unnecessary propagation delay observed in TLP-1 is eliminated by allowing the virtual token to sweep only between extreme stations where it has a packet to transmit. This efficiency is achieved by forcing a station with no backlog to idle. Fig. 4 shows the state diagram for TLP-2.

As a difference from TLP-1, state ST is no longer necessary (only backlog stations are active) and ON is replaced by states IDLE and B (i.e., Backlogged). IDLE is initially entered when a station is powered on. While in IDLE, transition to B only occurs when a packet is backlogged. Upon leaving ES, the system moves back to IDLE if no backlogged packet is present.

Transitions from B are similar to those from ON in TLP-1, except that time-out ND is set to 2R, allowing newly backlogged stations to smoothly join the other active stations [8].

In terms of state-diagram complexity, TLP-1 and TLP-2 are very similar. Performance, however, may differ substantially. When the active stations are confined to a small section of the network and remain busy for several rounds, TLP-2 is preferred to TLP-1. The virtual-token sweep is confined only to the span of the network covering the active stations, and stations do not incur initialization overhead since the channels are constantly busy. At heavy load, when all the stations are backlogged, TLP-1 and TLP-2 perform identically. If the load is light, TLP-2 shows inferior performance because the channel often becomes idle, and a newly backlogged station must execute the *initialization* procedure before transmitting on an idle channel.

4) **TLP-3**: A substantial contribution to overhead in both previous versions of the protocol is given by time-out R enforced between rounds. This delay can be reduced by observing that in TLP-1, if a station is an extreme station in a round, then in the next round the station is likely to be an extreme station again. TLP-3 works similarly to TLP-1, with the exception that an extreme station starts a new round in the opposite direction as soon as time-out 2d has elapsed since the last action of the station on the channel. Time-out 2d in TLP-3 is generally negligible compared to time-out R used in TLP-1. The result is a substantial performance improvement. Time-out 2d is sufficient to guarantee that a new powered-on outside station joins the set of active stations in a finite time. This joining procedure is explained in more detail in Section IV.

The state diagram for TLP-3 is shown in Fig. 5. As a difference from TLP-1, TLP-3 substitutes state ES with states ES0 and ES1. In addition, a flag $E(A)$ signals whether or not a station is the most upstream active station in channel $A$. ES0 is entered after a packet transmission if flag $E(A)$, corresponding to the future synchronizing channel $A$, is 0. ES0 performs similarly to ES. However, if time-out $R$ is reached while in ES0, $E(A)$ is set to 1, indicating that the station is currently an extreme station on that channel. Transition from ES0 to recovery-state $R$ only occurs if activity on channel $A$ [i.e., $BOA(A)$] is detected. If $BOA(A)$ occurs, the state moves to WFT, as in the normal procedure.

ES1 is entered after a packet transmission if flag $E(B)$, corresponding to the future synchronizing channel $B$, is 1. ES1 performs similarly to ES0 except that time-out 2d is used instead of time-out $R$ and any activity on either channel (while in ES1), triggers recovery. Transition into recovery resets $E(0)$ and $E(1)$ to 0.

Transition from ES0 to ES1 occurs when time-out $R$ expires, if $E(A) = 1$ and there is no backlog. If $E(A) = 0$ and there is no backlog, the state remains in ES0. In case of backlog, the state moves back to TXP. The reverse is true for transitions from ES1 to ES0 if time-out 2d is substituted for $R$.

TLP-3 is always superior to TLP-1 except in unrealistic cases where stations turn on and off continuously. In such a situation, collisions could force additional recovery delays of up to $2R + d$ per round (see Section IV). Under this circumstance TLP-1 performs better because overhead (not including propagation delay) is kept at $R$ per round.

The cost of this improved performance is a more complex state diagram and the additional use of status flags. These flags are needed as internal hardware variables, thus requiring more elaborate implementation.

5) **TLP-4**: TLP-4 combines features of both TLP-2 and TLP-3. The token sweep is confined between the most widely separated backlogged active stations, as in TLP-2.
Extreme stations preserve their status in flag variables set in the same manner as in TLP-3. The extreme-station flag variable allows round reversal with a minimum overhead of $2d$.

Fig. 6 shows the state diagram for TLP-4. As in TLP-2, state ST is unnecessary, because only backlogged stations are active. Also, transitions between ESO and ESI do not exist because absence of backlog moves the state to IDLE. The need for state WT is explained in Appendix I. Essentially, it prevents a lock-up condition which could cause infinite delays in accessing the network.

Because flag variable status is preserved when the station returns to IDLE, initialization delay is reduced by allowing a station with a channel flag set to 1 to transmit immediately on the corresponding channel, if both channels are sensed idle at packet arrival. If the station is an extreme station on both channels, the last value of $A$ determines the synchronizing channel. This procedure is executed concurrently with the transition from B to TTT. Also different from the initialization in TLP-2, a station does not start recovery if both channels are sensed busy while in B. Sensing both channels busy probably means that a recovery is occurring. There is no need to cause additional delay by starting a new recovery. The station simply remains in state B waiting for a channel to become idle.

Then, the station moves to state WFT synchronized on the busy channel. The busy channel flag is also reset to 0.

At heavy load, when all stations are active, TLP-4 performs identically to TLP-3. There are no collisions or initializations. Overhead between rounds is kept at $2d$ seconds.

At light load, first stations go through the initialization procedure. However, the extreme-station flag corresponding to the synchronizing channel is set to 1 after the first successful transmission on that channel, when the station is the current extreme station. Subsequently, the station can access the network, as in random mode, without any delay. If the station has a large packet backlog, the packets will be transmitted one after the other with interpacket gaps of $2d$ seconds. While no collision occurs, stations access the network freely. Delay at light load is decreased to almost zero.

At intermediate load, TLP-4 performance may degrade considerably. The token sweep may become confined to a small section of the network, so that a station outside this section must force a collision to get in. Furthermore, a station may join the network synchronized by one channel; then, if it has a flag set for the other channel, it reverses the round at the end of its packet transmission, even if there is another active station upstream. This premature round
reversal causes a collision with the upstream station transmission, triggering recovery and causing extra delay. This occurrence becomes more frequent as the ratio $\tau/T$ increases ($T$ is the packet length).

When leaving state B and going to state I, time-out $ND$ is set to $2R$, as in TLP-2. TLP-4 outperforms the other protocol versions under light-load conditions, in which case access delay becomes negligible. At heavy load, TLP-4 and TLP-3 both perform optimally. At intermediate-load levels an uneven distribution of the pattern with large backlog at only a few stations may favor TLP-4 over the other versions, as simulation results in Section VI show. For very large networks, the improvement may be considerable.

IV. RECOVERY AND JOINING

For all TLP versions, the recovery procedure is executed by states R1 and R2. R is a pseudostate which simplifies the drawing of state transitions to recovery. These transitions are drawn in dotted lines to distinguish them from transitions between regular states.

TLP protocols are structured so that stations sense only one channel busy at any one time. Furthermore, while the channels are “synchronized,” packet transmission is collision free and BOA events are expected only on the channel which is currently busy, or on which the station is presently transmitting an activity signal.

Transition into R1 is triggered by detection of simultaneous activity on both channels or by detection of upstream activity during packet transmission (collision). Either condition may be caused by station malfunctioning, or newly powered-on stations (TLP-3 and TLP-4), or newly backlogged stations (TLP-4).

Newly active inside stations are always transparently absorbed by the network (the joining process occurs without extra overhead). State ON (TLP-1, TLP-3) or B (TLP-2, TLP-4) guarantees the correct behavior by moving the station to state WFT when one of the channels is initially sensed busy. The variable $A$ is set to the busy channel.

A newly active outside station is still transparently absorbed in TLP-1 and TLP-2 because of delay $R$ between rounds. This station detects end-of-train in the synchronizing channel, and its transmission reaches the current extreme station before the round is reversed. It then becomes the new extreme station.

However, in TLP-3 and TLP-4, a newly active outside station can join the active network only after recovery is executed. In fact, the extreme station situated upstream of the joining station reverses the round before the joining station can transmit successfully. In both versions, the delay $2d$ in round reversal forces a collision following the
joining station attempt to transmit at the end-of-train in the round. Recovery then takes place.

Stations perform the recovery procedure in a distributed fashion. The recovery is completed in a finite time. The following steps are executed during recovery:

a) detection of abnormal condition and transition into R1,
b) transmission of activity on both channels for \( R = 2\tau + 2d \) while in state R1; after \( R \) has expired, move to R2, and
c) in R2, continue to transmit activity on channel L to R. After both channels are sensed idle, the station executes the standard procedure as if channel L to R had been initially detected busy. In TLP-2 and TLP-4, the state moves from R2 to TTT because a backlog always exists. In TLP-1 and TLP-3, if a backlog exists, the state moves from R2 to TTT. Otherwise, the state moves to ES or ES0, respectively, where the station checks whether or not it is an extreme station.

In Appendix I we show that recovery is completed in \( 2R + d \) seconds in the worst case, for any of the protocol versions.

V. PERFORMANCE ANALYSIS

In this section we develop analytic models for TPL performance. These models differ from conventional polling models in that the stations with backlogged packets transmit in sequential order from 1 to \( N \) and from \( N \) to 1, alternatively. This alternate operation makes channel-access time dependent on the position of the station on the bus. In fact, if stations are uniformly spaced and traffic is balanced, only the central station can access the network at uniformly distributed time intervals. All other stations observe alternatively shorter and longer access-time intervals. This asymmetry in time-access distribution introduces some unfairness in delay performance but does not affect station throughput, which is the same for all stations.

Before introducing the models, some definitions are in order.

- \( N \): Number of stations connected to the network.
- \( M \): Number of active stations with heavy backlog.
- \( T \): Packet transmission time (includes preamble overhead). This time is assumed constant.
- \( \tau_{ij} = \tau_{ji} \): Propagation delay between stations \( i \) and \( j \). Stations are assumed to be uniformly spaced along the buses.
- \( \tau = \tau_{i1} + \tau_{1N} \): End-to-end propagation delay on the bus.
- \( S_L \): Rightmost active station.
- \( S_f \): Leftmost active station.
- \( S_r \): rth station.
- \( T_{ES} \): Round-restart time out.
- \( d \): Reaction time.

A. Network Utilization

Under equilibrium conditions, network utilization \( U(M) \) is defined as the ratio between the time in a round spent for packet transmissions and the round duration, given that \( M \) stations are active and always transmitting in each round. The subscript \( i \) refers to TLP version \( i \).

The round duration, \( \tau(M) \), defined as the time between the detection of the end of round at one end station and the detection of the next end of round at the other end station, is given by \( \tau(M) = M(T + 2d) + T_{ES} + \tau_r \).

In reviewing this expression, we note that \( \tau(M) \) is maximum when \( S_L \) and \( S_R \) are the extreme stations \( (\tau_r = \tau_{1N} = \tau) \). Station reaction time is usually equal to a few bit times and, therefore, \( 2d \ll T \). \( T_{ES} \) represents the time needed for a station to discover that it is an extreme station and is equal to the time-out set during the round-restart procedure. In TLP-1 and TLP-2, \( T_{ES} \) is \( R \) seconds. In TLP-3 and TLP-4, \( T_{ES} \) may be assumed \( 2d \) at heavy load. In fact, at heavy load the identity of the extreme stations does not change; thus, no extra overhead is incurred due to initialization.

Based on the above considerations we find

\[
U_{1,2}(M, a) = \frac{1}{1 + \frac{a}{M}} = \frac{1}{1 + \frac{3a}{M}}
\]

and

\[
U_{3,4}(M, a) = \frac{1}{1 + \frac{a}{M}} = \frac{1}{1 + \frac{a}{M}}
\]

where \( a = \tau/T \).

Maximum network utilization is achieved for \( M = N \) (i.e., all the stations are active).

B. Delay Performance

Delay in this section is defined as the average insertion delay (ID), namely, the interval between the time when the packet reaches the head of the transmitting queue and the time when the successful transmission begins. The average is over all stations and over time. Analytic expressions for ID at light load (IDL) and at heavy load (IDH) are derived. Results for general load can be obtained only via simulation and are presented in Section VI.

1) Light Load: In TLP-4, insertion delay at light load is negligible. The first packet transmitted after power-on suffers a delay \( 3R \) due to network initialization, but all subsequent packets are transmitted immediately after arrival. In case of multipacket traffic at a single station, packets are transmitted with an interpacket gap of \( 2d \) seconds. The probability of collision during message transmission is assumed negligible.

In TLP-2 insertion delay is the time needed to initialize the idle network, which is \( 3R \). All single packets suffer this delay. In case of multipacket messages, the first packet suffers delay \( 3R \) and subsequent packets are transmitted with an interpacket gap \( R \).
For TLP-1 and TLP-3 all stations are assumed powered on. Therefore, $S_1$ and $S_N$ are the extreme stations. Consider station $S_i$. At light load, access instants for $S_i$ are alternatively separated by $x_i$ and $y_i$ time intervals where $x_i = n(x_i)(T + 2d) + T_{ES} + 2\tau_i$, and $y_i = n(y_i)(T + 2d) + T_{ES} + 2\tau_N$. Here, $n(\cdot)$ represents the number of packets transmitted in the corresponding interval.

The average insertion delay for packets generated at station $S_i$ at random points in time is

$$\text{IDL}_i = \frac{x_i}{2} \text{Prob} \{\text{arrival in } x_i\} + \frac{y_i}{2} \text{Prob} \{\text{arrival in } y_i\} + \frac{x_i + y_i}{2} \frac{1}{x + y} + \frac{y_i}{2} \frac{1}{x + y} = \frac{T_{ES} + 2\tau_i^2}{2(T_{ES} + 2\tau)} + \frac{T_{ES} + 2\tau_N^2}{2(T_{ES} + 2\tau)}.$$  

(3)

Maximum IDL occurs at the end stations and minimum IDL occurs at the central station(s).

TLP-1 shows $(3/2)\tau \leq \text{IDL}_i \leq (5/3)\tau$, and TLP-3 shows $(1/2)\tau \leq \text{IDL}_i \leq \tau$, which demonstrates that the difference in IDL among stations is always less than $\tau/2$. Averaging over all stations yields

$$\text{IDL}_{TLP-1,3} = \frac{1}{N} \sum_{i=1}^{N} \text{IDL}_i = \frac{1}{N} \left[ \frac{T_{ES}^2}{2} + T_{ES}\tau + \frac{\tau^2}{3} \right].$$  

(4)

2) Heavy Load: At heavy load, stations always have a packet to transmit, and the time intervals between consecutive access rights at station $S_i$ are alternatively

$$x_i + (2(N - i) + 1)(T + 2d) + T_{ES} + 2\tau_i,$$

and

$$y_i(2(i - 1) + 1)(T + 2d) + T_{ES} + 2\tau_i.$$  

The average insertion delay is

$$\text{IDH}_i = \frac{x_i + y_i}{2} - T = (N - 1)T + 2Nd + T_{ES} + \tau = \text{IDH}.$$  

IDH is independent of station location and increases linearly with the number of stations. As expected, ID is bounded for any value of offered traffic.

VI. SIMULATION RESULTS

As discussed in the previous section, analytic results can be obtained only for extreme load conditions (either light or heavy) and for symmetric traffic pattern. For a thorough evaluation and comparison of the various TLP versions, a more general set of traffic conditions must be tested. This is possible only via simulation.

In this section, we report the results corresponding to three different simulation experiments. In the first two experiments, the input rate is uniform over all the stations. The difference is in the length of the cable (1000 m in the first case, and 10 000 m in the second case). In the third experiment, an unbalanced traffic situation is considered; namely, one station has high input rate, while the remaining $N - 1$ stations offer light background load.

The first two experiments permit us to validate the analysis in Section V in light and heavy load conditions. As we shall see, the agreement is very good. In addition, the experimental results are used to understand the tradeoffs between the different versions of TLP protocols in various traffic conditions.

All experiments assume a network with 15 stations ($N = 15$), infinite buffers per station, channel speed of 1 Gbit/s, fixed packet length of 1000 bits, and packet interarrival time exponentially distributed. The preamble in each packet is 100 bits. The length of the cable is assumed to be 1000 m in the first experiment and 10 000 m in the second and third experiment. Propagation delay in the medium is assumed to be 5 $\mu$s/km. The reaction time $d$ is set to 10 ns (i.e., 10 bits at 1 Gbit/s). In collecting simulation results, 95 percent confidence intervals are obtained through batch runs. Trial runs were used to identify and exclude the transient phase of each run.

A. Equally Loaded Stations — 1000 m Span

This experiment is used to compare the behavior of the various TLP protocols in a uniform traffic environment. In addition, a validation of the analytic models is possible at light and heavy loads.

The experiment consists of a sequence of runs executed for different values of the (uniform) station-input rate, ranging from very light load to saturation load. For each run, insertion delay (ID) and bus utilization are monitored and are plotted in Fig. 7. (Note: bus utilization excludes preamble O/H.)

First, we wish to validate the analytic results of Section V. Recall that the value of the parameter $a = \tau/T$ in this case (1000 m, 1000 bits per packet, and 1 Gbit/s channel) is $5$. The parameter $a$ has a critical impact on network performance. At very light load, the analytic models provide the following values of insertion delay.

<table>
<thead>
<tr>
<th>TLP</th>
<th>IDL</th>
<th>$\tau$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLP-1</td>
<td>$1.56$</td>
<td>$7.8$ $\mu$s</td>
</tr>
<tr>
<td>TLP-2</td>
<td>$3R$</td>
<td>$30$ $\mu$s</td>
</tr>
<tr>
<td>TLP-3</td>
<td>$0.69$</td>
<td>$3.45$ $\mu$s</td>
</tr>
<tr>
<td>TLP-4</td>
<td>$0$</td>
<td></td>
</tr>
</tbody>
</table>

The experimental values for zero load (extrapolated from the measurements in Fig. 7) show excellent agreement with the above analytic results.

At heavy load, the analytic models provide the following values of insertion delay IDH and bus utilization.

<table>
<thead>
<tr>
<th>TLP</th>
<th>IDH</th>
<th>$U$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLP-1</td>
<td>$30.7$ $\mu$s</td>
<td>$0.47$</td>
</tr>
</tbody>
</table>
Again, the agreement with the analytic models is excellent.

At intermediate loads, we observe two interesting effects. First, the insertion delay of TLP-2 decreases for an initial increase in load. This is due to the fact that at very light load, each packet transmission requires channel initialization. This explains the high delay at zero load. As load increases, more and more stations become active. Thus, the channel tends to be synchronized most of the time and does not require initialization. This explains the reduction in section delay for increase in load.

Secondly, the insertion delay for TLP-4 increases sharply for an initial increase in load, and then levels off for higher loads. This is due to the fact that at very light load, each station can transmit immediately. However, as load builds up, collisions occur which force reinitialization and thus introduce additional delay. At intermediate loads, the situation stabilizes and the channel becomes permanently synchronized, thus drastically reducing the frequency of collisions.

From this experiment we conclude that TLP-3 offers the lowest delay for a very wide range of channel utilizations. As for the effective throughput on the channel, TLP-3 and TLP-4 are tied at 70 percent utilization, and definitely outperform TLP-1 and TLP-2.

B. Equally Loaded Stations — 10 000 m Span

This experiment is identical to the previous one except for the length of the network, which is now 10 000 m. The purpose of the experiment is to evaluate the effect of the parameter $a$ on performance. In the previous experiment, we had $a = 5$. Now we have $a = 50$.

The results of the simulation experiment are shown in Fig. 8. First, we validate the analytic model in Section V. At very light load, the model predictions are

\[
\begin{align*}
\text{TLP-1: } & \quad IDL_1 = 78 \mu s \\
\text{TLP-2: } & \quad IDL_2 = 300 \mu s \\
\text{TLP-3: } & \quad IDL_3 = 34.5 \mu s \\
\text{TLP-4: } & \quad IDL_4 = 0
\end{align*}
\]

Again, we observe good agreement with simulation results. At heavy load, the analysis predicts the following.

\[
\begin{align*}
\text{TLP-1: } & \quad 165.7 \mu s, \quad 0.091 \\
\text{TLP-2: } & \quad 165.7 \mu s, \quad 0.091 \\
\text{TLP-3: } & \quad 65.7 \mu s, \quad 0.23 \\
\text{TLP-4: } & \quad 65.7 \mu s, \quad 0.23
\end{align*}
\]

Agreement with the experimental results is excellent.

Comparing the results in Fig. 8 to those in Fig. 7, we note that the relative ranking of the various TLP schemes is the same: TLP-3 is still the best, except for the very light load region where TLP-4 prevails. We also observe a maximum in the TLP-4 insertion delay. As explained earlier, this is due to the fact that collision and reinitialization overhead first increases, then decreases with increasing load.

The overall effect on performance of the increase of $a$ from 5 to 50 is quite dramatic. Insertion delays increase by a factor of 5 to 10; throughput is reduced by a factor of 3 to 5. The throughput degradation is most significant for TLP-1 and TLP-2, as expected, since these schemes introduce a gap of 2 $\tau$ s between each round. TLP-3 and TLP-4 do not introduce this overhead, thus, are less sensitive to the increase in end-to-end propagation delay.

C. Single Heavy-Loaded Station with Uniform Background Traffic

In this experiment, the input rate of one of the stations is progressively increased from zero to saturation while the remaining stations offer a fixed background load of 5 Mbits/s. This background load is very light, considering the fact that channel speed is 1 Gbit/s. The network span is 10 000 m.
The purpose of this experiment is to reproduce an unbalanced load situation which is fairly common in high-speed networks, namely, one or a few stations involved in very high data-rate transfers (e.g., digitized video or image), while the remaining stations are transmitting a more conventional type of traffic.

In this experiment, we are interested in monitoring overall throughput and delay performance as well as the impact of heavy station-rate increase on background stations’ delays.

The results are displayed in Figs. 9, 10, 11 and in Table I. Fig. 9 shows insertion delay ID and queueing delay QD for the heavy station (station no. 8 in our example) as a function of increasing offered load for protocols TLP-1, TLP-2, and TLP-3. Insertion delay ID was already defined before. Queueing delay QD is the total time spent by a packet in the station input buffer. We note that at light load ID and QD coincide, as expected, since there is practically no wait in the input buffer. Also, ID at light load is consistent with the values reported in Fig. 8. As the load at station 8 increases, ID increases slightly (or even decreases, as is the case for TLP-2) while QD increases asymptotically as the bus becomes saturated. We note that TLP-3 can practically double the throughput of the other two schemes, due to the fact that the gap between rounds is eliminated.

Fig. 10 shows ID and QD values for the background stations for TLP-1, TLP-2, and TLP-3, as a function of increasing load at station 8. We note that ID and QD practically coincide, that is, no queueing occurs in station input buffers due to the light background load. Furthermore, the insertion delay does not seem to be affected by the increase in station 8 load, thus confirming the fairness of the TLP protocols.

Fig. 11 shows ID and QD for TLP-4. A separate diagram was necessary since TLP-4 throughput performance is one order-of-magnitude better than the performance of the previous schemes. In Fig. 11, the performance of both heavy station (i.e., station 8) and background stations are shown. We note that at light load the insertion delay is comparable to that reported in Fig. 8 for the uniform load experiment. As load increases, the background station delay remains approximately constant, while the delay for the heavy station decreases dramatically. This is explained by the fact that in TLP-4, when a station is transmitting alone on the bus, it can transmit packets one immediately after the other with practically no gap in between.

Table 1 summarizes the throughput performance (expressed in terms of maximum bus utilization) of the various TLP protocols for this experiment. TLP-4 exhibits a performance one order-of-magnitude better than the other schemes. Total bus utilization of TLP-4 is in excess of 20 percent. This is a surprisingly good result, considering the fact that the upper bound for TLP-4 with uniform load is 22 percent, as shown in Fig. 8.

D. Comparing TLP Versions

TLP-3 and TLP-4 decisively outperform TLP-1 and TLP-2 in all examples. For equally loaded and symmetrically spaced stations, TLP-3 outperforms TLP-4, and TLP-1
TABLE I
Maximum Bus Utilization in Single Heavy-Loaded Station Experiment

<table>
<thead>
<tr>
<th>PROTOCOL</th>
<th>MAX BUS UTILIZATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLP-1</td>
<td>0.012</td>
</tr>
<tr>
<td>TLP-2</td>
<td>0.014</td>
</tr>
<tr>
<td>TLP-3</td>
<td>0.024</td>
</tr>
<tr>
<td>TLP-4</td>
<td>&gt;0.20</td>
</tr>
</tbody>
</table>

outperforms TLP-2. However, single heavy-load station and asymmetrical placement favor the adaptive versions TLP-2 and TLP-4.

TLP-4 is the only version to show no deterioration of performance for single heavy-loaded traffic. The achieved maximum utilization of TLP-4 in this case is more than 10 times better than the next best maximum utilization (TLP-3).

For all protocols, background stations are unaffected by the heavy-load traffic. This tolerance is valuable because it guarantees a fair share of resources. Furthermore, bounded delays are inherent to all protocols.

The fact that TLP-4 provides much lower queueing delays than the other protocols makes TLP-4 an excellent choice for applications where bursty high-bandwidth traffic occurs (file transfers, graphics, etc.). The insensitivity of background performance to heavy traffic from a single source guarantees proper service to interactive and priority traffic.

VII. CONCLUSIONS

This paper has described four versions of tokenless protocols for dual-bus architectures. The control operation of the protocols is solely based on the detection of activity on the channel and is completely distributed. The protocols are such that the circuitry needed at line speed is simple and small. The network can support 20–30 stations connected via passive taps, providing reliable operation.

Access is collision free in TLP-1, TLP-2, and TLP-3. In TLP-4, collisions may occur when a previously idle station becomes active. Even in this case, however, the collision is resolved in a finite time (on the order of the round-tip delay) and packet delay remains bounded.

Joining and recovery actions have been analyzed and TLP behavior under adverse conditions was proved correct. Exact expressions for behavior at light and heavy load were derived. Results for unbalanced message traffic were obtained via simulation and were found to agree with analytic results. The analysis of the results shows that TLP-4 outperforms the other versions under most conditions. Background traffic has been shown to be relatively insensitive to foreground heavy traffic. The above findings indicate that TLP is a suitable choice for implementation of very high speed local area networks.

APPENDIX I

Claim: Steps a), b), and c) in Section IV guarantee complete recovery within 2R + d seconds in the worst case.

Proof: Assume that station S_i is the first station to start recovery at time t_o. Define T[EVENT_{i}(A)] as the time that event EVENT detected or originated at S_i tap on channel A reaches S_j tap on the same channel. Hence, t_o = t_j[BOA(\cdot)].

S_j activity signal, transmitted on both buses, hits another station S_k at t_k[BOA(\cdot)] = t_o + \tau_j. Here, if S_k is not yet in recovery, it moves to state WFT and waits for EOA in the normal procedure. Otherwise, S_j starts recovery. In the latter case, the activity signal transmitted on both channels by S_j starts at t_j[BOA(\cdot)] = t_o + \tau_i + \tau_j + \tau_k + d. R seconds later, S_j activity on channel R to L stops, and S_j moves to R2. Any active station S_k, between S_j and S_i, not yet in recovery, moves into recovery at t_k[BOA(\cdot)] = t_o + \tau_i + \tau_j + \tau_k + 2d + R, and S_k moves to R2.

Assume S_j, l \leq i, and S_k, r \geq i, are, respectively, the leftmost and the rightmost stations on recovery. S_j starts recovery at most at time t_R1 = t_j[BOA(RL)] and enters state R2 by time t_R2 = t_R1 + d + r = t_o + \tau_i + d + R. Nevertheless, S_i can only detect channel R to L idle by

\[ t_{idle} = \max \{ t_j[BOA_k(RL)] \mid S_k \text{ in recovery} \} \]

\[ = \max \{ t_o + \tau_i + \tau_k + \tau_{sl} + 2d + R \mid 1 \leq k \leq r, S_k \text{ in recovery} \} \]

\[ = \{ t_o + \tau_i + \tau_{sl} + 2d + R \}, \]

which depends only on the position of the extreme stations involved in recovery.

From the expressions above, t_{idle} \geq t_{R2}. Therefore, S_i starts packet transmission at the latest by t_s = t_{idle} + d. The worst case for t_s occurs for i = l = 1 and r = N.

Therefore,

\[ \max \{ t_s \} = t_o + 2\tau + 3d + R \]

\[ = t_o + 2R + d, \]

and complete recovery occurs within 2R + d from the detection of illegal events on the channels.

Now the need for state WT in TLP-4 is explained. Assume for a moment that transitions to WT go directly to IDLE. Following recovery in TLP-4, if S_i has only one backlogged packet it can return to idle after leaving ESO and setting E(RL) = 1. However, if S_i receives another packet before it detects activity on channel L to R, S_i may transmit and cause another recovery before stations on the left of S_i have the opportunity to transmit. This behavior may repeat following each succeeding round, possibly preventing the low-numbered stations from transmitting packets.

State WT prevents such a lock up from occurring. After the transmission from the next active downstream station reaches S_i (at most, R s after S_i leaves ESO), S_i is in synchronism again. Consequently, a station leaves WT and goes to IDLE after BOA(\cdot) has been detected or time-out R has expired. Time-out R is set when state WT is entered.
REFERENCES


Luigi Fratta (M'74) received the Doctorate in electrical engineering from the Politecnico di Milano, Milano, Italy, in 1966. From 1967 to 1970, he worked at the Laboratory of Electrical Communications, Politecnico di Milano. As a Research Assistant at the Department of Computer Science, University of California, Los Angeles, CA, he participated in data network design under the ARPA project from 1970 to 1971. From November 1975 to September 1976, he was with the Computer Science Department of the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, working on modeling, analysis, and optimization techniques for teleprocessing systems. In 1979 he was a Visiting Associate Professor at the Department of Computer Science, University of Hawaii, Honolulu, HI. In the Summer of 1981 he was at the Computer Science Department, IBM Research Center, San Jose, CA, working on local area networks. At present, he is a full Professor at the Dipartimento di Elettronica of the Politecnico di Milano. His research interests concern communication networks.

Dr. Fratta is a member of the Association for Computing Machinery and the Italian Electrotechnical and Electronic Association.

Paulo Rodrigues received the M.Sc. degree from COPPE, Rio de Janeiro, Brazil in 1977 and the Ph.D. degree from the University of California, Los Angeles, CA, in 1984, both in computer science. He has been a consulting computer analyst and professor at the Federal University of Rio de Janeiro (UFRJ), Brazil, since 1973, where he actively participated in the development of micro systems and peripheral devices. In 1978, he was director of EBC, a Brazilian computer industry, and in 1979, he joined the Ph.D. program at UCLA where he worked as a research assistant in high-speed fiber optic local area network research. He is presently in the research division of the Computer Center (NCE) at UFRJ. His current technical interests are in design and performance analysis of computer networks, local area networking supporting integrated services, interconnection of computer networks, and distributed operating systems.

Mario Gerla (M'75) received the graduate degree in engineering from the Politecnico di Milano, Milano, Italy, in 1966, and the M.S. and Ph.D. degrees in engineering from University of California, Los Angeles, CA, in 1970 and 1973, respectively.

From 1973 to 1976, he was Network Planning Manager at Network Analysis Corporation, New York, and led several computer network design projects for both government and industry. From 1976 to 1977, he joined Tran Telecommunications, Los Angeles, CA, where he participated in the development of an integrated packet and circuit network. In 1977 he joined the University of California, Los Angeles, and is now a Professor in the Department of Computer Science. His research interests include the design and control of distributed computer communication systems and networks, and the development of protocols for high-speed local area networks.