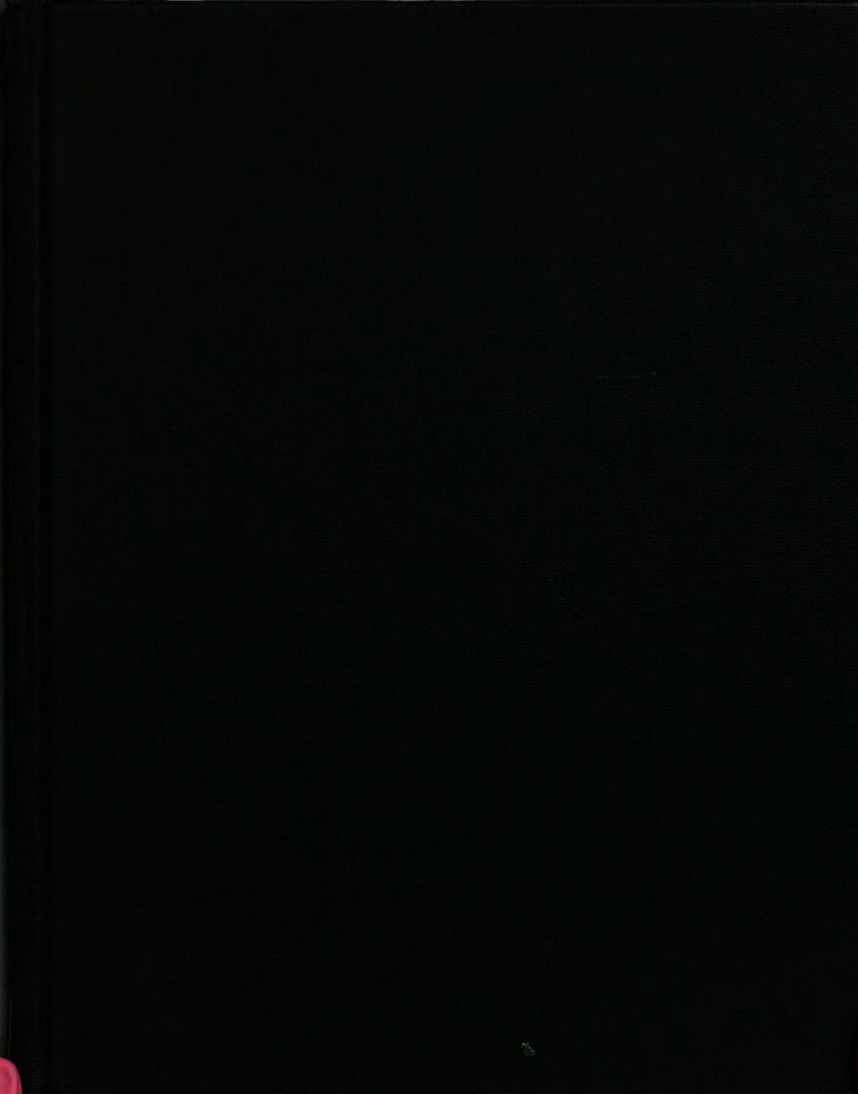
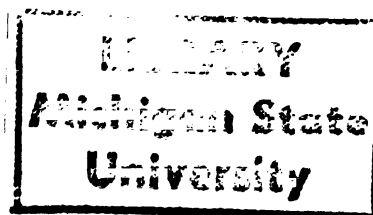




THESIS



This is to certify that the
dissertation entitled

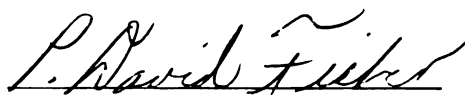
TOKEN-BUS NETWORK PERFORMANCE ENHANCEMENTS

presented by

M. A. Anura P. Jayasumana

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in Elect. Engineering


Major professor

Date December 21, 1984



RETURNING MATERIALS:
Place in book drop to
remove this checkout from
your record. FINES will
be charged if book is
returned after the date
stamped below.

SEP 11 1989
#24 225

300 A281

TOKEN-BUS NETWORK PERFORMANCE ENHANCEMENTS

By

M. A. Anura P. Jayasumana

A DISSERTATION

**Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of**

DOCTOR OF PHILOSOPHY

**Department of Electrical Engineering
and Systems Science**

1984

334-3303

To
MOTHER AND FATHER

ACKNOWLEDGMENTS

I wish to express my deepest gratitude to Dr. P.D. Fisher, my major Professor, for his help, guidance and encouragement. I feel very fortunate to have got a chance to work with him for my Ph.D. I also wish to express my sincere appreciation to Dr. Lionel Ni for his suggestions and evaluation of this research. I am very thankful to Dr. M.A. Shanblatt for his suggestions and review of this work. The interest and encouragement provided by Dr. S.R. Crouch is thankfully appreciated. Appreciation is also extended to Dr. S. Sayegh and A. Khurshid for their help during this research. I would like to thank B. MacArthur for his expert drawings. Also acknowledged is the support provided by the National Science Foundation (Grant No. MCS 79-09216) for this research. Finally, I am very thankful to my wife, Geetha, for her patience and understanding during my graduate studies and constant help during the preparation of this thesis.

ABSTRACT

TOKEN-BUS NETWORK PERFORMANCE ENHANCEMENTS

By

M. A. Anura P. Jayasumana

Token passing is an access scheme used in Local Area Networks, which allows multiple stations to share a communication channel without conflict. In token-bus networks, the right of access to the bus is passed from station to station by means of a control packet, known as the token. A major disadvantage of the token bus is the overhead involved in passing the token. This overhead has a significant effect on the performance of the network, especially when the offered load is low, the traffic distribution among stations is asymmetric or when priority schemes are implemented. This research investigates alternative methods for enhancing the performance of the token-bus network.

A channel-access scheme, called the token-skipping scheme, is described, which preserves the advantages of the token-passing bus while significantly reducing the overhead required to pass the token. For example, for throughputs up to about 0.65 of the bandwidth, a network with 100 nodes with Poisson arrivals and symmetric traffic

distribution among the nodes, has a mean delay of less than half that of a similar network using the standard token-passing channel-access scheme. An analytical model is developed to predict the behavior of the token-skipping scheme under the following conditions: Poisson arrival of messages, symmetric traffic distribution among the nodes and exhaustive service discipline. Simulation results are used to validate the analytical model. The token-skipping channel-access protocol has been used to enhance the performance of the IEEE 802.4 priority scheme, while ensuring upward compatibility with the proposed standard. Simulations show this modified scheme to be superior in terms of performance to the IEEE 802.4 scheme under a wide variety of load conditions. In addition, this protocol has been used to implement a priority scheme, which is hierarchically independent in performance among packets of different classes and fair among packets of the same class.

TABLE OF CONTENTS

	Page
LIST OF TABLES	v
LIST OF FIGURES	vi
1. INTRODUCTION	1
2. BACKGROUND	5
2.1 Local Area Networks	10
2.2 IEEE 802.4 Standards for the Token-Bus	11
2.3 Performance Evaluation of Local Area Networks	16
3. PERFORMANCE ENHANCEMENT OF TOKEN-PASSING BUSES	24
3.1 The Token-Skipping Scheme for Bus Networks	25
3.2 A Modified Priority Scheme for Token-Passing Buses	36
3.3 The Token-Skipping Priority Scheme for Token-Passing Buses	40
4. AN ANALYTICAL MODEL FOR THE TOKEN-SKIPPING SCHEME	49
4.1 Analysis of the Polling Scheme	49
4.2 Analysis of the Token-Passing Scheme	52
4.3 Analysis of the Token-Skipping Scheme	53
5. RESULTS AND INTERPRETATION	64
5.1 Token-Skipping Scheme under Non-Exhaustive Service of Messages	64
5.2 Token-Skipping Scheme under Exhaustive Service of Messages	73
5.3 Modified Priority Scheme for Token-Passing Buses	81
5.4 Token-Skipping Priority Scheme for Token-Passing Buses	88
6. SUMMARY AND CONCLUSIONS	99
APPENDIX	104
BIBLIOGRAPHY	113

LIST OF TABLES

Table		Page
5.1	Network parameters used for comparing the token-skipping scheme and the IEEE 802.4 channel-access scheme with round-robin service discipline	65
5.2	Speedup factors under symmetric traffic distribution	69
5.3	Speedup factors under asymmetric traffic distribution	72
5.4	Network parameters used for validating the analytical model for the token-skipping scheme	74
5.5	Network parameters used for comparing the modified priority scheme with the IEEE 802.4 priority scheme	82
5.6	Network parameters used for comparing the token-skipping priority scheme with the IEEE 802.4 priority scheme	89

LIST OF FIGURES

Figure	Page
2.1 Relationship of IEEE 802.4 standards to the ISO-OSI reference model	12
2.2 The logical ring and the physical bus	14
3.1 The logical ring and the physical bus	26
3.2 The logical ring and the set of extreme chords	29
3.3 An example of the path of the token in the token-skipping scheme	31
3.4 Updating of chords following the transmission of a token with $M=1$ by station X a) before token transmission, b) after token transmission	33
3.5 Updating of chords following the transmission of a token with $M=0$ by station X a) before token transmission, b) after token transmission	34
3.6 A four-layer logical ring	41
3.7 An example of the token-path in a hierarchically independent priority scheme	42
3.8 An example of the channel activity with the token-skipping priority scheme	47
4.1 A successful attempt to pass the token along a logical chord	56
4.2 An unsuccessful attempt to pass the token along a logical chord	57
5.1 Delay-throughput characteristics under symmetric load distribution for a network with 50 stations	67
5.2 Delay-throughput characteristics under symmetric load distribution for a network with 100 stations	68
5.3 Delay-throughput characteristics under asymmetric load distribution for a network with 50 stations	70

Figure	Page
5.4 Delay-throughput characteristics under asymmetric load distribution for a network with 100 stations	71
5.5 Delay-throughput characteristics for a network with $N = 50$, $S_m = 3$ and exhaustive service discipline	75
5.6 Delay-throughput characteristics for a network with $N = 50$, $S_m = 5$ and exhaustive service discipline	76
5.7 Delay-throughput characteristics for a network with $N = 50$, $S_m = 10$ and exhaustive service discipline	77
5.8 Delay-throughput characteristics for a network with $N = 100$, $S_m = 3$ and exhaustive service discipline	78
5.9 Delay-throughput characteristics for a network with $N = 100$, $S_m = 5$ and exhaustive service discipline	79
5.10 Delay-throughput characteristics for a network with $N = 100$, $S_m = 10$ and exhaustive service discipline	80
5.11 Delay-throughput characteristics [$T(I)$ vs. $S(I)$, $I=0, \dots, 3$] for the modified priority scheme when the traffic is equally distributed among the priority classes [i.e., when $G(I) = G/4$, $I=0, \dots, 3$]	84
5.12 Total throughput vs. total offered load [S vs. G] for the modified priority scheme when the traffic is equally distributed among the priority classes	85
5.13 Delay-throughput characteristics [$T(I)$ vs. $S(I)$, $I=1, 3$], for the modified priority scheme when all the traffic belongs to class I	86
5.14 Delay-throughput characteristics [$T(I)$ vs. $S(I)$, $I=0, \dots, 3$] for the token-skipping priority scheme when the traffic is equally distributed among the priority classes [i.e., when $G(I) = G/4$, $I=0, \dots, 3$]	91
5.15 Total throughput vs. total offered load [S vs. G] for the token-skipping priority scheme when the traffic is equally distributed among the priority classes	93

Figure	Page
5.16 Mean delay vs. throughput of packets of priority class 3 [$T(3)$ vs. $S(3)$] for the token-skipping priority scheme when the offered load in class 1 is held constant and there is no traffic in classes 0 and 2 [i.e., $G(1) = \text{const.}$ and $G(0) = G(2) = 0$]	94
5.17 Mean delay vs. throughput of packets of priority class 1 [$T(1)$ vs. $S(1)$] for the token-skipping priority scheme when the offered load in class 3 is held constant and there is no traffic in classes 0 and 2 [i.e., $G(3) = \text{const.}$ and $G(0) = G(2) = 0$]	95

CHAPTER 1

INTRODUCTION

During the past decade, local area networks have become one of the fastest expanding fields in computer and communication research. There are several factors which motivate this expansion. Technological advances have resulted in the availability of inexpensive, sophisticated computing devices. It is desirable to interconnect all the devices in a given site to provide for flexible use of facilities, share resources and to improve overall performance. Local area networks also offer the possibility of interconnecting equipment manufactured by different vendors.

Interconnection topology and channel-access protocol are two major characteristics of a local area network (LAN). This research program will investigate the performance of a token-bus architecture, which uses the bus topology and token-passing accessing scheme. In the token-bus architecture, all the nodes are connected to a common bus and the access to the bus is governed by a token which is passed sequentially from node to node. The node which possesses the token is allowed the use of the bus. Token-passing channel-access scheme guarantees the right of access to each node in the network within a predefined period of time. This deterministic nature is a critical

quality for certain applications such as process control and other real time tasks. Token-passing bus is emerging as the LAN standard for industrial environment due to its reliability and the deterministic property.

A LAN used in an office environment interconnects devices such as terminals, disk drives, micro-, mini- and main-frame computers and memory banks. In an industrial environment, it may interconnect devices such as alarms, robots, process control equipment, computers and memory banks. A LAN in general interconnects devices with diverse characteristics and requirements. The LAN should be capable of meeting the traffic handling requirements of each device connected to it. IEEE 802.4¹³ supports the different traffic requirements by offering four priority classes. It is the responsibility of the station management to allocate a fraction of bandwidth to each class of traffic. The traffic generated by a device can be characterized by its message length and the intermessage arrival time. For analytical purposes, the arrival process can be assumed to be a Poisson process. The message length could be fixed or have a distribution. It is possible to have more than one type of messages arriving at a node. The performance of the token-bus will depend on the number of nodes, the rate of arrival of traffic of each class at each node and the network management. The previous studies on token-bus architectures involve the delay and throughput considerations in the presence of a single class of traffic^{3,6,19,25,27,32,}

The goals of this research are to provide a better understanding of the capabilities and limitations associated with a local area network in handling traffic classes with diverse requirements and characteristics, and thereby increase the efficiency of information transfer. The specific tasks are outlined below:

1. Identify the limitations of token-bus network in handling different classes of traffic.
2. Devise a priority mechanism, that would allow the token-bus to meet the requirements of different classes of traffic efficiently. In devising the priority scheme, the qualities defined in Section 2.1, i.e., hierarchical independence, fairness, robustness and low overhead should be considered. This will require investigating network management issues involved in handling classes of traffic with different requirements, such as what global information should be exchanged between nodes, how can this information be collected and used efficiently and how can the broadcast nature of the token-bus architecture be exploited for efficient network management.
3. Investigate the performance of the token-bus architecture for different classes of traffic. The performance measures will include the delay-throughput characteristics and the offered load vs. throughput characteristics for the network.

This research work is aimed at improving the understanding of limitations and capabilities of a token-bus network in handling different classes of traffic. It will provide the designer of a token-bus network with a priority scheme that would allow stations with different requirements to be connected to the network efficiently. This research program also provides the user of a token-bus LAN with improved performance by providing efficient communication between devices connected to the network.

The basic concepts associated with token-bus architecture and LANs in general are discussed in Chapter 2. In Chapter 3, a channel-access scheme called token-skipping scheme is presented, which preserves the advantages of token-passing bus, while enhancing its performance. Chapter 3 also describes how to implement priority functions when the token-skipping scheme is used. An analytical model for the token-skipping scheme is described in Chapter 4. Chapter 5 contains simulation results comparing the token-skipping channel-access protocol and the priority schemes with the proposed IEEE 802.4 channel-access protocol. It also compares the analytical model with results obtained via simulation. Chapter 6 contains a summary of this research work and some future research possibilities.

CHAPTER 2

BACKGROUND

This chapter is devoted to reviewing the background information related to this research work. The basic concepts associated with LANs are discussed in Section 2.1. In Section 2.2, the proposed standards for the token-bus architecture is outlined. Section 2.3 describes performance evaluation techniques for local computer networks.

2.1 LOCAL AREA NETWORKS

A local area network can be considered as one of the three types of general communication networks, the other two being "long haul communication networks" and "I/O bus communication networks"¹⁴. A communication network can be regarded as a set of nodes joined by a set of links which carry information between nodes. Each communication path between a source and a destination is called a channel and a link may contain several channels. The three types of communication networks differ mainly on the length over which they have to carry information. I/O bus networks, which typically connect dependent components of a computer system (e.g., connect a processor with terminals, bulk memory devices, etc.) span distances of up to about 100 meters. Long haul networks have links which have lengths greater than

few hundred meters (e.g., connect different computers at branch offices of a company). The nodes in a local area network are separated by distances in the order of 10 m to 10 km. They often carry bursty data and use packet communication. Digital Private Branch Exchanges are a class of circuit switched local area networks, which are capable of handling voice and data. Burr³ defines a local area network as a network, where the end-to-end propagation delay is large compared with a single bit transmission duration and small compared with the message transmission duration.

There are two major applications for local area networks. LANs may be used to connect together a collection of hosts, terminals, and peripherals in a limited geographic area and also to allow them to access a remote host or other networks. In order to communicate, each device needs only to connect to the LAN at one place, whereas in the absence of an LAN, each device will need a separate connection to the device it wants to communicate with. The second area of applications of LAN's is in distributed processing. The current trends in hardware costs favor the use of a number of smaller machines to a single large machine to obtain a given computing power. It is also possible to have machines dedicated to specific functions, e.g., file storage, data base management and terminal handling, connected together using a local area network. The dedicated units will enhance the efficiency of the distributed system.

The geographic characteristics of a local area network allow inexpensive high bandwidth transmission media such as coaxial cables and twisted pairs to be used. The high bandwidth and low delay of LAN's can be exploited to simplify the control structure of communication protocols because it relaxes the requirement of minimizing the length of control or overhead information in a packet⁸.

The major elements of a LAN may be classified into three groups: a transmission medium, interface units and a set of protocols. Most LAN's currently in operation use coaxial cable or twisted pairs as the transmission medium. When coaxial cables are used, the LAN can use broadband or baseband signalling. Baseband networks have the advantage of simplicity and low cost, while broadband networks have a higher bandwidth, which can be used to achieve higher throughput or to support simultaneous transmissions of data, video and voice. Optical fiber is another option that is currently being investigated¹⁶. It can support data rates as high as 1 Gbps and has advantages such as immunity to electromagnetic interference²³.

The interface unit, used to connect a node or a device to the transmission media, can be viewed as having two parts: a network oriented part that performs the transmission control functions required by the network, and a host specific part that fits into the I/O structure of the particular device or host and controls the exchange of data between the node and the network dedicated portion of the interface.

The set of protocols implemented in the devices and interface units connected to the network controls the transmission of information from one node to another via the hardware elements of the network. Stack, et al.¹¹ define protocols as the procedures and conventions used to regiment the event progression required for orderly, mutually understood interaction between processes. The requirements that have to be met by protocols include flexibility (to accomodate new uses and features), completeness (to properly respond to all the relevant network conditions), deadlock avoidance, error detection and recovery, priority mechanisms, communication network feature compatibility, and acceptable throughput and delay performances.

To reduce the design complexity, the networks are designed hierarchically as a series of layers, each layer performing a small set of closely related functions. The LAN manufacturers closely follow a reference model, called the Open Systems Interconnection (ISO-OSI) model, developed by the International Standards Organization, which defines seven layers and the function of each layer^{13,14,17}.

Two design parameters, the topology and the channel-access protocol, together with the transmission medium, characterize a local area network. The network topology refers to the pattern of interconnection used to connect the nodes of the LAN. Three topologies, star, ring and bus, are widely used in LANs. The star topology relies on a central node for the control of its operation. The central node may be a packet switching device or a circuit switching device such as a Private Branch Exchange (PBX). Digital

FBX's allow the integration of voice and data on a single network. The ring topology connects its nodes in a closed network and circulates all messages in one direction. Often, messages are amplified and repeated at each node they pass through.

This research will deal with the bus topology. In a bus network, all the nodes are connected to a common bus such as a coaxial cable or a twisted pair. A node connected to the bus can listen to all the transmissions by other nodes and read only those that are intended for it. Most modern LANs use either the ring topology or the bus topology because they do not rely on a central node. Both these networks need a channel-access protocol to determine which node may transmit at a given time.

There are a number of channel-access protocols that have been proposed for the use in LANs. They can, however, be classified into three basic groups: fixed assignment (dedicated) protocols, random assignment (contention) protocols and demand assignment protocols^{22,30}. In fixed assignment protocols, the channel bandwidth is allocated to stations in a static manner. Time Division Multiplexing (TDM) and Frequency Division Multiplexing (FDM) belong to this category. In random assignment protocols, a node transmits a message at a time determined by itself using the information available to that node. ALOHA and Carrier Sense Multiple Access (CSMA) protocols belong to this type. The demand assignment protocols use the communication channel explicitly to carry control information issued by the nodes and used by them to determine the access order of the shared channel.

Unlike fixed assignment protocols, demand assignment protocols tend to utilize the bandwidth more effectively by not assigning the channel to idle nodes. Unlike the random access protocols, they also avoid waste of bandwidth due to collisions. The access protocol used in this research, token passing, belongs to the group of demand assignment protocols.

The token-passing scheme, being a deterministic scheme, gives an upper bound on the delay a node has to undergo before transmitting a message. This allows it to be used in factory automation, process control and other real time applications. Unlike contention protocols, the token passing is not limited by a maximum data rate. Of the number of topologies and channel-accessing schemes that can be used in LANs, the IEEE 802 standards committee has proposed to standardize three channel-accessing schemes: CSMA/CD¹³, token-passing bus¹⁴ and token-passing ring.

Different types of devices, used for different applications are connected to an LAN. These devices have diverse requirements and unless their messages are handled differently, depending on their needs, some applications will not have their needs met and some resources on the network may be used in a very inefficient manner. This problem is handled by the use of different classes of service^{9,16}. The datalink layer supports the service class requirement by a priority scheme, in which the packets of a higher priority class are dynamically allocated a higher bandwidth than the packets with a lower priority.

Rom and Tobagi¹⁸ outline the requirements of an acceptable priority scheme as

1. Hierarchical independence of performance - The performance of the scheme as seen by messages of a given priority class should be independent of the load in the lower priority classes.
2. Fairness within each priority class - The packets of the same priority class should be able to contend equally on the channel bandwidth.
3. Robustness - The operation and performance of the priority scheme should be unaffected by errors in station information.
4. Low overhead - The overhead required to implement the priority scheme should be minimal.

2.2 IEEE 802.4 STANDARDS FOR THE TOKEN-BUS

Proposed IEEE 802.4 standard is compatible with physical and data link layers of the ISO-OSI reference model as shown in Figure 2.1. The data link layer is divided into two sublayers Medium Access Layer (MAC) and Logical Link Control Layer (LLC). Proposed IEEE 802.4 standards¹⁹ specify the functions of physical layer and the MAC sublayer, the transmission medium, and the interfaces between the transmission medium and the physical layer, physical layer and MAC sublayer, MAC and LLC sublayers, the station management and MAC layer, and the station management and physical layer. IEEE 802.2¹¹ specifies the logic link

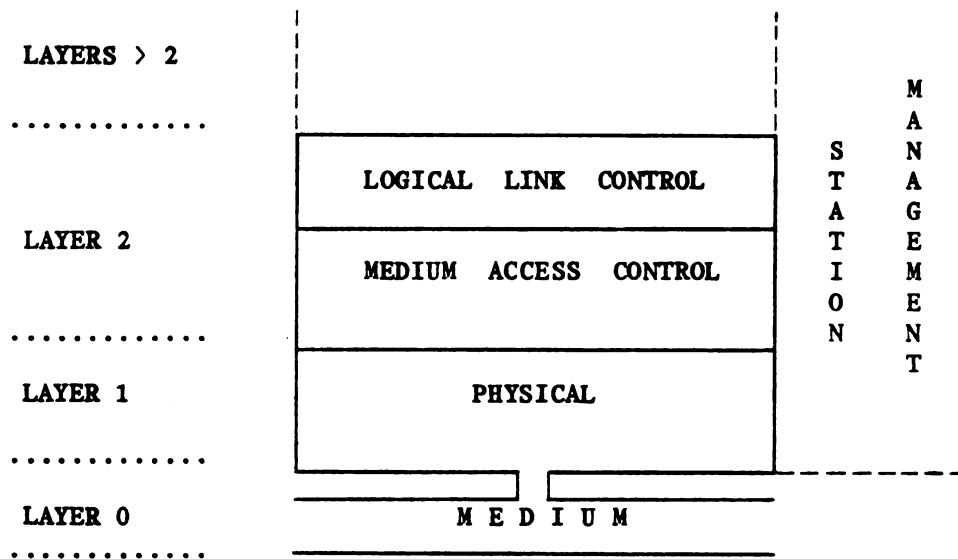


Figure 2.1 Relationship of IEEE 802.4 Standards to the ISO-OSI Reference Model¹³.

control layer. A brief description of the token-bus scheme (as specified in IEEE 802.4) is given below.

In the token-passing access method, a token controls the right to access to the physical medium. When a station receives the token, it becomes the temporary master of the network and has the right to transmit one or more messages subject to the constraints imposed by the priority scheme discussed below. If the node does not have any messages to send or when its token-holding time expires, the token is passed to the next station. This creates a logical ring as shown in Figure 2.2, around which the token passes. Hence, in the steady state, in the absence of noise, there will be alternate data and token transfer phases. The logical ring maintenance responsibilities such as ring initialization and lost token recovery are distributed among all token using stations of the network.

The MAC sublayer is responsible for recognizing and accepting the token and passing the token to the next station. The responsibilities of the LLC sublayer include the initiation of control signal interchange and organization of data flow¹¹. Three different physical layer entities are specified in proposed IEEE standard 802.4¹³:

1. Phase Continuous Frequency Shift Keying on an omnidirectional bus with a data rate of 1 Mbps, the data being Manchester coded;
2. Phase Coherent Frequency Shift Keying on an omnidirectional bus with data rates of 5 Mbps and 10 Mbps;

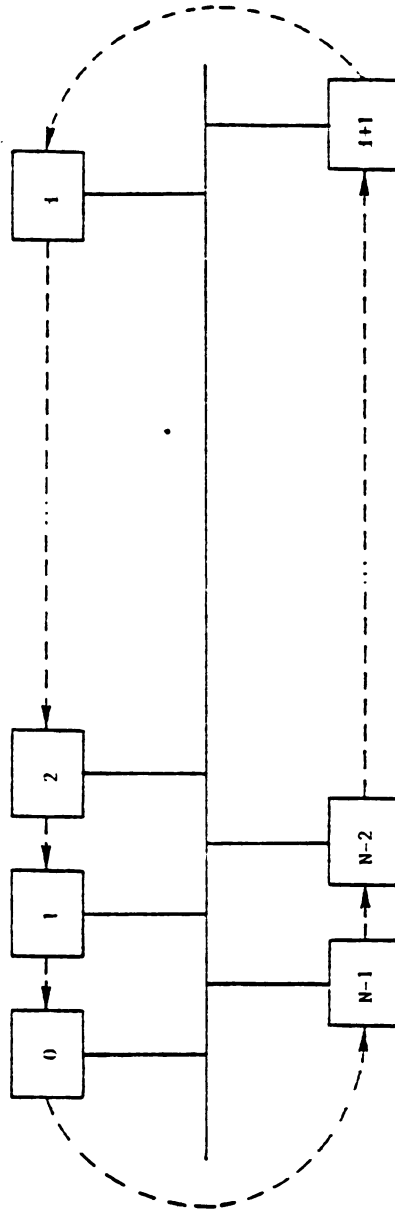


Figure 2.2 The logical ring and the physical bus.

3. Multilevel Duobinary Amplitude Modulation/Phase Shift Keying
with data rates of 1 Mbps, 5 Mbps, 10 Mbps.

The proposed IEEE 802.4 standards provide an optional priority mechanism. The priority of each frame is indicated when the LLC sublayer requests the MAC sublayer to send a data frame. The MAC sublayer offers four levels of priority classes, called "access classes". There is a separate queue for frames in each class to wait pending transmission. Any station not having the priority option transmits every data frame with the highest priority value. The object of the priority scheme is to use the network bandwidth to transmit the high priority frames and send low priority frames when there is sufficient bandwidth available.

The priority scheme works as follows: Any station, which receives the token, transmits frames of the highest priority class in a time not exceeding some maximum time called the high-priority token-hold time set by the station management. This high-priority token-hold time period prevents any single station from monopolizing the network. After sending the high priority frames, it starts servicing the queue of the next access class. Each access class at a node is assigned a "target-token-rotation time". For the three lower access classes, the station measures the time it takes for the token to circulate around the logical ring. If the token returns to the station in less than the target-token-rotation time, the station is allowed to send frames of that particular access class until the target-token-rotation time has expired. If the target-token-rotation time has expired by the time the

token returns, the station is not allowed to send frames of that access class. The fraction of bandwidth that will be allocated to various classes is controlled by the target-token-rotation time for each access class. The responsibility of setting these values lies with the station management.

2.3 PERFORMANCE EVALUATION OF LOCAL AREA NETWORKS

There are two major approaches for performance evaluation of a local area network: direct measurement and modeling. Direct measurement is accurate, but is not possible during design stages and also involves considerable human and machine costs. Due to the broadcast nature, a passive monitor can do the measurements without affecting the performance of a token-bus network. In modeling, a model is developed by identifying the system components and component attributes (state variables) and the rules of interaction between them to predict the performance. Franta and Chalmatac¹⁵, Reiser²⁷, Sauer and Chandy²⁹ and Tobagi et al.³¹ describe the modeling and measuring techniques used in LANs.

Some parameters which are used to evaluate the performance of an LAN are defined below. Channel throughput is defined as the fraction of the bandwidth of the channel utilized in transmitting packets of data. Channel capacity is the maximum possible value of the channel throughput that can be achieved with the protocols employed in the LAN. Delay is defined to be the delay experienced by a message following its

arrival at a node until its successful transmission. Ease of adding and deleting stations, reliability and extendability are some qualitative measures used in performance evaluation of LANs.

Analytic models, simulation models and hybrid (of simulation and analytic) models are used to predict the performance of LANs. The major limitation of analytic models is the need to use formal descriptions of interaction rules. Queuing theory and theory of stochastic processes are often used tools for analytical modeling^{1,17,18,19}. Simulation techniques are used in cases where the interaction between states of the system are analytically intractable. Simulation is also used for validation of analytical models based on simplifying assumptions.

Queuing theory is a tool often used for analyzing network performance. A queuing system can be characterized by

1. the interarrival time distribution;
2. the service time distribution;
3. the number of servers;
4. the queuing discipline (FIFO etc.); and
5. the amount of buffer space in the queue.

Generally, the mathematical tractability requires that the message arrival process be based on infinite or finite source processes such that the interarrival time τ is an exponential random variable, i.e., having a distribution function

$$\text{Prob}[\tau \leq t] = 1 - e^{-\lambda t}. \quad (2.1)$$

λ is the mean message arrival rate. In such a process, called the Poisson process, the number of arrivals is Poisson distributed, i.e., the probability of k arrivals in an interval t is given by¹⁷

$$p_k(t) = (\lambda t)^k e^{-\lambda t} / k!. \quad (2.2)$$

This is the so called infinite source model. In the finite source model, the arrival rate depends on the number of messages currently in the system. The service time of a packet refers to the time the packet transmission takes once the transmission is started. This depends on the packet length. If the packet size is fixed, the service time is constant.

Performance analysis of the token-bus access scheme is very similar to that of a polled access scheme. In token-bus, poll can be considered as passing from node to node instead from a central node issuing a poll. The performance of such a system is analyzed by Reiser³⁷ assuming that when a station receives a poll, the entire buffer is emptied.

Arthurs and Stuck³ have analyzed the polling scheme with both finite source model and infinite source model for the arrival statistics. They have obtained expressions for the mean delay, mean throughput and the channel utilization for both exhaustive service i.e., the buffer is emptied when the node receives the token, and nonexhaustive service. Kuehn³⁸ has dealt with multiqueue systems with nonexhaustive cyclic service. He provides an approximate analysis of

multiqueue system $M^{(x)}/G/1$ with batch Poisson input, general service times, general overhead times and a single server operating under a cyclic strategy with nonexhaustive service queue. Pack and Whitaker²⁴ have compared a number of variants on polling in a multidrop communication context. Stuck²⁵ has calculated the maximum mean throughput rates in local area networks.

Konheim and Meister¹⁹ have analyzed the polling scheme used in communication networks under equal traffic distribution among the nodes and exhaustive service. The results of this analysis of polling scheme have been used to analyze other demand assignment access protocols in Local Computer Networks^{2, 21, 22, 24, 26}. The basic relationships for polling scheme derived by Konheim and Meister¹⁹ are reviewed in the rest of this section. Consider a communication system consisting of N buffered nodes connected using a single communication channel. The nodes are polled in sequence and the data is removed from the node buffer. When the buffer has been emptied, i.e., exhaustive service, the channel is used for system overhead for an interval called the reply interval. The system then continues with the poll of the next node. Traffic is assumed to be identically distributed among the nodes and the reply intervals have a random length. The packet size is assumed to be constant and each packet carries one message.

The time axis is assumed to be divided into slots of size Δ . Each slot is able to carry one data unit. Let $X_j^{(i)}$ be the number of data units entering the i th node during the j th slot. $X_j^{(i)}$ are non-negative, integer valued, independent and identically distributed

random variables with mean μ and variance σ^2 respectively. A poll of each of the N nodes is called a cycle, i.e., a cycle consists of N consecutive service intervals during which the channel is made available to remove data from the nodes. Each service interval is followed by a reply interval. Let $R_{i,j}$ be the reply interval in the j th cycle following the service interval of the i th node. $R_{i,j}$ are positive, integer valued, independent and identically distributed random variables which are also independent of the data arrival processes. Let r and δ^2 be the mean and the variance of the reply interval. Then from Konheim and Meister¹⁹, when $r < \infty$ and $N\mu < 1$, the stationary expected cycle length T and its variance Δ_T^2 are given by

$$T = \left[Nr / (1-N\mu) \right] \quad (2.3)$$

and

$$\Delta_T^2 = \left[\delta^2 N / (1-\mu)(1-N\mu) \right] + \left[\sigma^2 r N^2 / (1-\mu)(1-N\mu)^2 \right]. \quad (2.4)$$

The stationary expected queuing and transmission delay undergone by messages, D , is given by

$$D = 1 + \left[\delta^2 / 2r \right] + \left[\sigma^2 N / 2(1-N\mu) \right] + \left[1 + Nr / (1-N\mu) \right] (1-\mu) / 2. \quad (2.5)$$

All the times being given in terms of slot times.

The above results are now used to predict the behavior of a system with Poisson arrival of messages. The following notation is used:

λ -- Mean packet arrival rate at a node in packets per packet time.

This is also equal to the variance of the number of packets arriving at a node within a packet time for the case where packet arrival follows Poisson distribution.

N -- Number of nodes in the network;

S -- The total throughput normalized with respect to the bandwidth;

a_0 -- Slot time normalized with respect to the packet time i.e., a slot time is equal to a_0 packet times. Slot time is a small unit of time which is used to express the mean and variance of the token-passing time;

r -- Mean time to pass the token between two stations in terms of slot times;

δ^2 -- Variance of the token-passing time (δ is given in slot times);

T_t -- Token-transmission time in slot times;

T_{rw} -- Response window duration in slot times.

The following results are derived²¹ using Konheim and Meister's¹⁹ analysis, for the case where message arrivals at a node follows Poisson distribution, i.e., probability that k packets arrive at a node in t packet times is given by Equation (2.2). Hence the mean number of slots required to serve the data arriving in one slot μ and its variance σ^2 are given by

$$\mu = \lambda \quad (2.6)$$

$$\sigma^2 = \lambda/a_0. \quad (2.7)$$

Under stationary conditions, i.e., $r < \infty$ and $N\mu < 1$,

$$S = N\lambda. \quad (2.8)$$

Using (2.6), (2.7) and (2.8) in (2.5), (2.3) and (2.4) respectively, we obtain the following results. When $r < \infty$ and $N\lambda < 1$, the stationary expected delay (queuing and transmission) normalized with respect to the packet time is given by²¹

$$D = 1 + \left[\delta^2 a_0 / 2r \right] + \left[S / 2(1-S) \right] + (a_0/2) \left[1-S/N \right] \left[1 + Nr/(1-S) \right]. \quad (2.9)$$

The stationary expected cycle length T and its variance Δ_T , normalized with respect to the packet time are

$$T = Nra_0 / (1-S) \quad (2.10)$$

and

$$\Delta_T^2 = \left[\delta^2 N a_0^2 / (1-S/N)(1-S) \right] + \left[r a_0 S N / (1-S/N)(1-S)^2 \right]. \quad (2.11)$$

The above analysis provides the mean delay undergone by messages and the mean and variance of the cycle time when the packet arrival process is Poisson, the load is equally distributed among the nodes and under exhaustive service of messages.

CHAPTER 3

PERFORMANCE ENHANCEMENT OF TOKEN-PASSING BUSSES

A major disadvantage of the token-bus local area network architecture is the bandwidth used to pass the token through idle stations. This has a significant effect on the performance of the network when the traffic is asymmetrically distributed among the nodes and/or when the offered load is low. When priority schemes such as the IEEE 802.4 scheme are implemented, a station receiving the token may not transmit any messages due to the constraints imposed by the priority scheme. The token-skipping scheme described in Section 3.1 enhances the performance of the network by allowing the token to bypass most of the nodes which would not use the token to transmit any packets. Section 3.2 describes how the IEEE 802.4 priority scheme is modified by using the token-skipping scheme to achieve better performance characteristics. Section 3.3 describes a priority scheme that possesses such characteristics as hierarchical independence of performance among different classes of packets and fairness among packets of the same class.

3.1 THE TOKEN-SKIPPING SCHEME FOR BUS NETWORKS

In the standard token-passing channel-access scheme, the token passes from node to node in a logical ring as shown in Figure 3.1. The major disadvantage of this scheme is the need to pass the token through idle stations. The overhead required for this could be minimized by allowing the token to skip idle stations. In the proposed scheme, this is done by allowing the token to be passed along a chord of the logical ring in the absence of messages at the stations bypassed by the chord. First, few terms used to describe the token-skipping scheme are defined. In the definitions below, an asterisk (*) used as a superscript indicates that the same terminology is used in the IEEE 802.4 scheme^{1,3} with identical meaning.

The token carries the following information in addition to the preamble, cyclic redundancy code and delimiters.

SA* -- The source address;

DA* -- The destination address;

M -- A one-bit flag, which indicates the presence of untransmitted packets at the source node of the token.

The term response window refers to the maximum time, T_{rw}^* , during which a station expects to hear the response from another station following the transmission of a frame. It is given by

$$T_{rw} = T_p + 2 * T_s + T_{sm}. \quad (3.1)$$

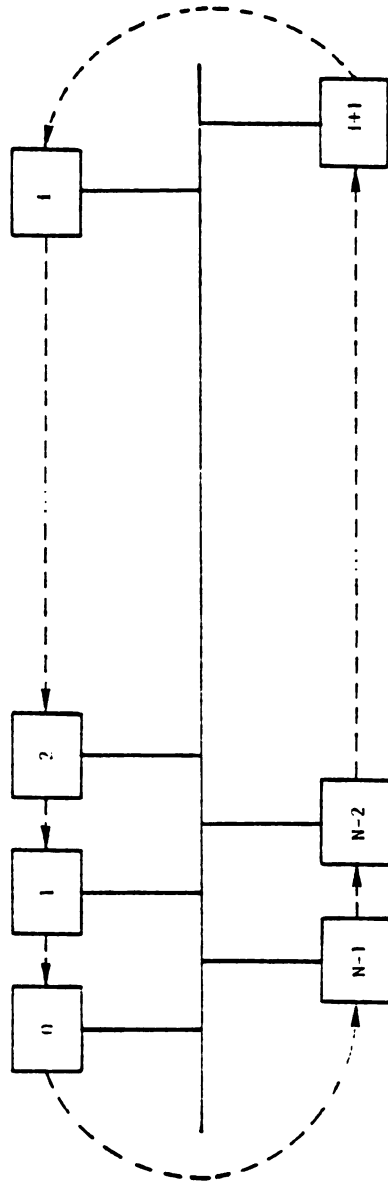


Figure 3.1 The logical ring and the physical bus.

where, T_p^* , T_s^* and T_{sm}^* are the round trip propagation delay, station delay (i.e., the time a station requires to receive and respond to a query by another station) and a safety margin to assure proper reception of a response, respectively.

Each station is assumed to contain the following information:

TS^* — This station, i.e., its own address;

RS^* — Address of the next station along the logical ring (called NS elsewhere¹³);

PS^* — Address of the previous station on the logical ring;

CS — Address of a station at the other end of a selected chord passing through TS ;

R — Ring flag, a one-bit flag indicating whether the token should be transferred along the logical ring or along the selected chord, i.e., to CS .

TS , PS and RS are used and initialized the same way as proposed in the IEEE 802.4 scheme¹³. In the token-skipping scheme, a station maintains the value of CS , the chord station, to be the address of a station within a logical distance S_m along the logical ring, which had untransmitted messages during the last cycle of the token. If there is no such node, it points to the node at a logical distance S_m . This value of CS is called the extreme value of CS and is denoted by CS' . S_m is a constant, called the maximum skip distance. The maximum number of idle stations that are skipped by a token is limited to $(S_m - 1)$ and hence its value directly affects the performance of this scheme. CS' may be initialized by one of the following schemes:

1. When a station adds on to the logical ring it selects its chord station, CS' to be the same as the next station RS along the logical ring. Periodically, a node would transmit a special frame, which would cause the stations in the network to set their ring flags. So in the next cycle, the token would travel from station to station along the logical ring, which may be tracked by the stations on the network to find out CS' , the address of a station at a distance S_m .
2. When a station adds on to the logical ring, it transmits a special control frame with the address of its successor on the logical ring, RS and the value of S_m . Following this, the next S_m nodes will respond by transmitting their addresses one by one. This allows the new node to determine the address of the station at a distance S_m .

Figure 3.2 shows the logical ring, together with the extreme set of chords for a network with N nodes with addresses $0, 1, \dots, N-1$.

In the token-skipping scheme, the token may be passed from one station to another in one of three ways:

1. Explicit token passing along the ring: If $R = 1$ then a station sets $SA = TS$ and $DA = RS$ in the token during token transmission. In this case, the token is accepted by the next station, RS along the logical ring.
2. Explicit token passing along a chord: If $R = 0$ then a station sets $SA = TS$ and $DA = CS$. Now the token is followed by a response window. Any station that lies between SA and DA ,

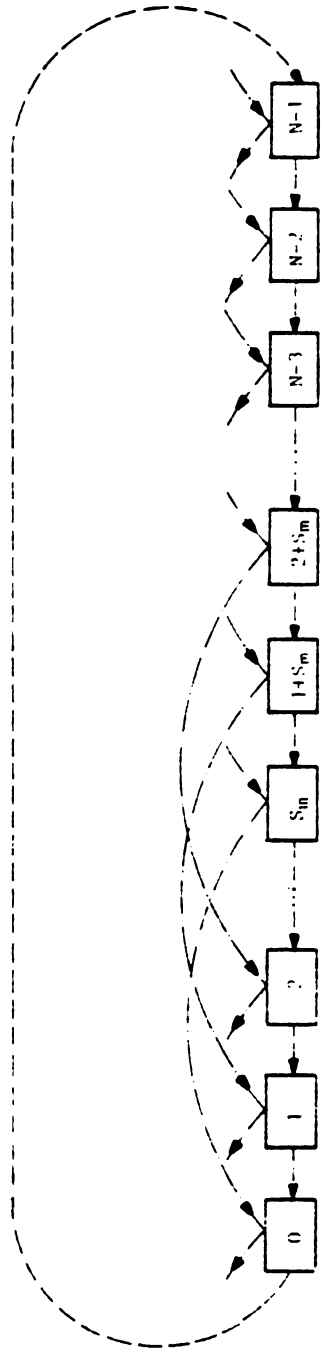


Figure 3.2 The logical ring and the set of extreme chords.

having messages to transmit, will respond with a burst during this window. An idle window indicates that all the stations bypassed by the chord are idle, and hence the station at the other end of the chord, DA will accept the token. When $CS = RS$, no response window is used since CS and TS are adjacent and no other station is skipped.

3. Implicit token pass along the ring: In the above case, a busy response window prevents DA from accepting the token. The station following SA, RS on logical ring will accept the token. This is possible because each station knows the address of previous station on the logical ring.

When an attempt to pass the token along a chord fails, it is accepted by the station next to the source station of the token. In this case all stations bypassed by the chord will set their R flags, implying that the token now has to move along the ring until a station with a message is encountered. The ring flag R at a station is cleared when the station transmits a token or a message appears on the bus. Figure 3.3 shows an example of the token path under this scheme.

A major advantage of a broadcast medium, such as a bus, is the fact that all transmitted information can be received by all the stations in the network. This fact is made use of in updating the selected logical chord or address of CS for each station. The value of CS at each station is dynamically updated with every token pass according to the following rules:

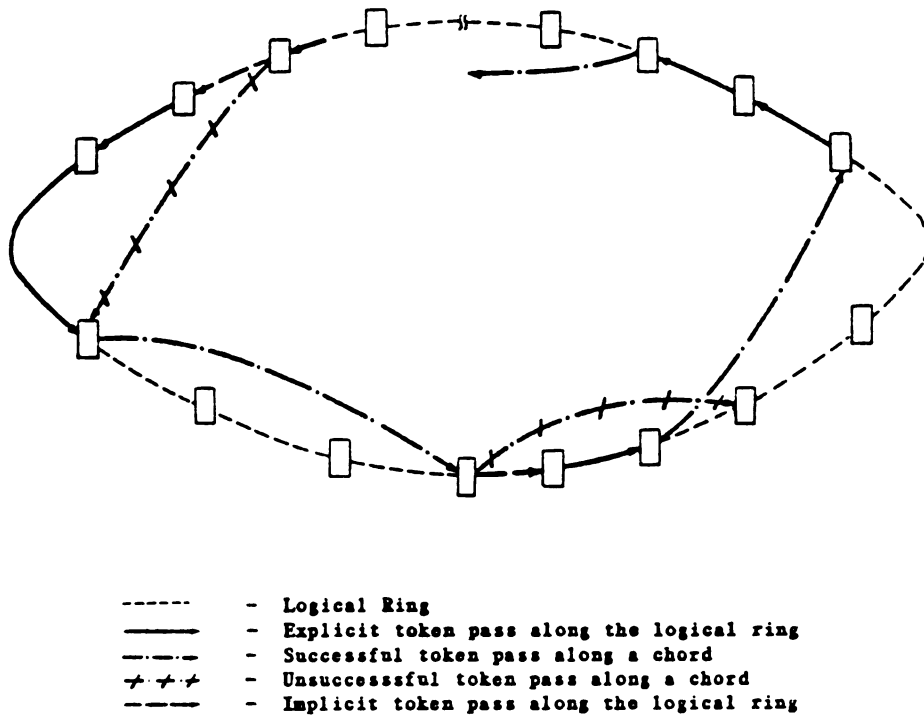


Figure 3.3 An example of the path of the token in the token-skipping scheme.

1. When a token with $M = 1$ appears on the bus, it indicates that the source station SA of the token has untransmitted messages, so each station will see whether SA is bypassed by its selected chord, i.e., whether SA lies in between TS and CS. In this case, it updates its CS to be SA. This is illustrated in Figures 3.4a and 3.4b.
2. When a token with $M = 0$ appears on the bus, each station compares the source address SA of the token with the CS of the station. If they are equal, then the station knows that its CS no longer has messages in its buffer and it resets the CS to its extreme value CS'. Figures 3.5a and 3.5b illustrates this for a network with N nodes with addresses $0, 1, \dots, N-1$, respectively.

Thus each station selects its CS to be the closest station which had a message in its buffer at the time it last transmitted the token. However, the maximum number of stations that are allowed to be bypassed by a chord is limited to $(S_m - 1)$. Hence if there are no stations having a message within a logical distance S_m , a station will select the station at a distance S_m to be its CS. Note that the limiting case $S_m = 1$ corresponds to the standard token-bus scheme. A formal description of the token-skipping scheme is given in Appendix A.

The performance of this scheme for a given load depends on S_m , the maximum skip distance. The maximum number of idle stations that the token can attempt to skip is limited to $(S_m - 1)$. If the length of a chord is s ($1 \leq s \leq S_m$), the time saved, T_s , by a successful token

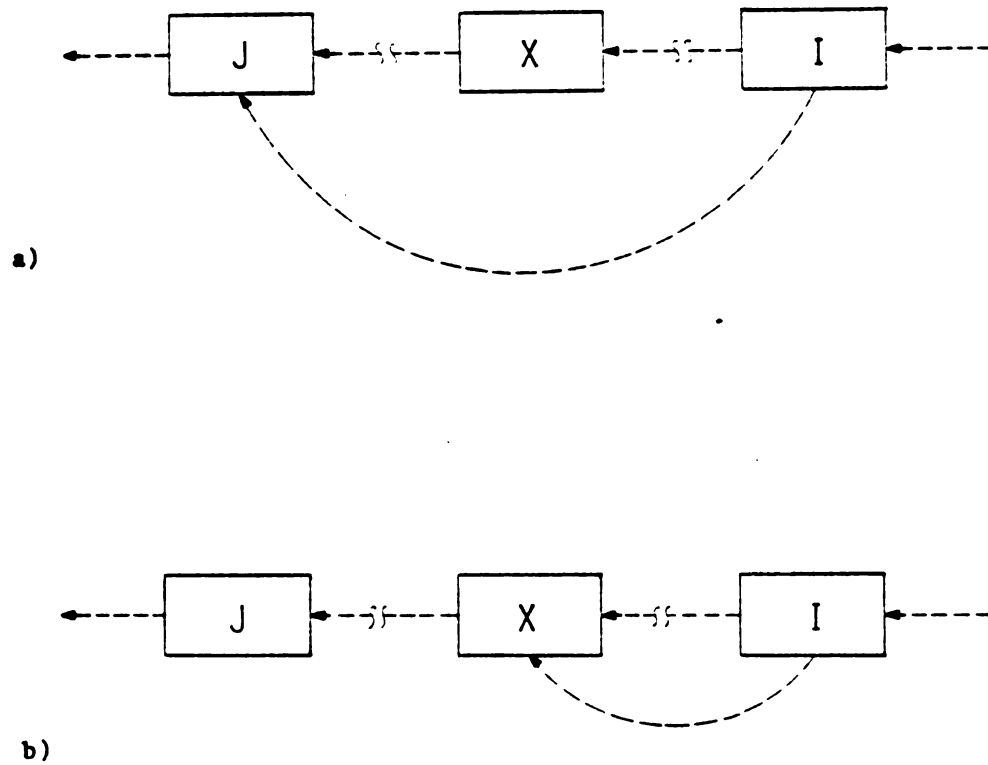
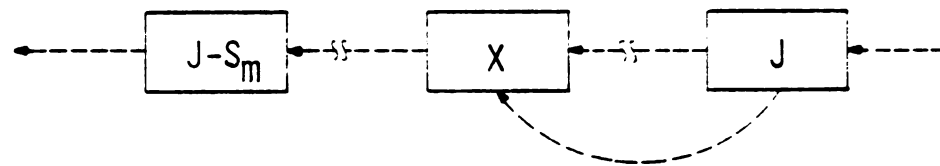
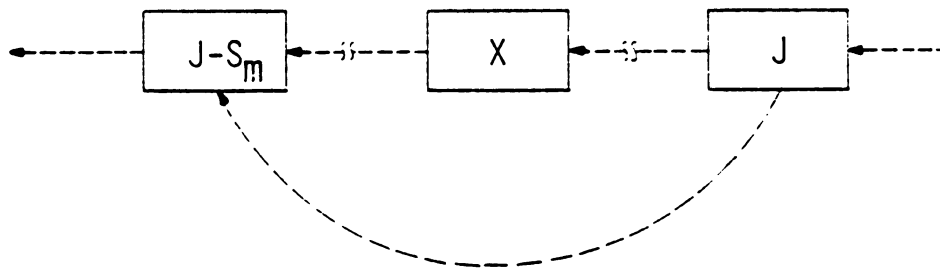


Figure 3.4 Updating of chords following the transmission of a token with $N=1$ by station X
a) before token transmission,
b) after token transmission.



a)



b)

Figure 3.5 Updating of chords following the transmission of a token with $M=0$ by station X
 a) before token transmission,
 b) after token transmission.

pass along a chord is given by

$$T_s = (s-1) T_t - T_{rw} \quad (3.2)$$

where T_t is the time required for the token to be transmitted and T_{rw} is the response window time given by Equation (3.1). The time lost, T_1 , by an unsuccessful token passing attempt is

$$T_1 = T_{rw}. \quad (3.3)$$

If s , the length of a chord is small, the advantage to be gained by a successful token pass along a chord is small. However, as s increases, there will be a higher chance of the intermediate stations having messages awaiting transmission. In general, for a moderate load, increasing S_m can be expected to provide a decrease in the queuing delay due to the savings in the number of token passes. Further increase in S_m , however, will cause an increase in the delay due to increase in the number of unsuccessful token-passing attempts. The optimum value of S_m can be expected to decrease with increased load for symmetric traffic distribution among stations. In cases where the number of idle stations is high, for example when few of the stations are responsible for a large portion of the traffic the modified scheme should fare even better due to its ability to skip idle stations.

In this protocol, the exceptions such as multiple tokens, lost tokens, station failures etc., can be handled the same way as in the

standard token-passing scheme¹³. In Chapter 4, an analytical model is developed for this scheme under exhaustive service discipline. Section 5.1 contains results of simulations of this scheme under non exhaustive service of messages and compares it with the standard token-passing scheme. In Section 5.2, the analytical model is compared with simulation results under exhaustive service of messages.

3.2 A MODIFIED PRIORITY SCHEME FOR TOKEN-PASSING BUSES

This section describes how the token-skipping concept is used to modify the IEEE 802.4 priority scheme to achieve a better performance. A node in a network using the IEEE 802.4 priority scheme may not transmit packets due to two reasons: It does not have any packets to transmit or the constraints imposed by the priority scheme does not allow packets to be transmitted. However, the token has to pass through each node irrespective of whether they are able to transmit packets or not. Token-skipping scheme can be used to allow the token to skip those stations that would not transmit messages upon receiving the token.

In the description below, P classes of messages are assumed with 0 and $P-1$ corresponding to the lowest and the highest priority respectively. In this scheme, the message flag M of the token indicates the presence of untransmitted packets of highest priority class at the source node of the token. Hence the logical chords are selected so that they point to the closest node along the logical ring

which had untransmitted packets of the highest class of priority during the last cycle of the token. If there is no such node within a logical distance S_m , the chord is selected to point to the node at a logical distance S_m .

The priority scheme makes use of the following times:

$HPTHT^*$ -- High-priority token-hold time;

$TRT(i)^*$, $i=0, \dots, P-2$ -- Target-token-rotation time for Class i ;

$TOL(i)$, $i=0, \dots, P-2$ -- Tolerance for Class i .

Each station using the priority scheme also has the following timers:

$TRTIM(i)^*$, $i=0, \dots, P-2$ -- Token-rotation timer for Class i ;

THT^* -- Token-hold timer.

These timers control the number of packets a station transmits upon receiving the token as explained below.

A node is said to be 'active' if

1. The node has untransmitted packets of the highest priority class, or
2. The node has untransmitted packets of Class i , $i=0, \dots, P-2$ and the residue of the token-rotation timer of Class i is greater than the tolerance of Class i , i.e.,

$$TRTIM(i) > TOL(i). \quad (3.4)$$

The basic difference between this scheme and the IEEE 802.4 priority scheme lies in the way that the token is passed among

stations. Here, the token may be passed from one station to another in one of three ways described in Section 3.1: explicit token passing along the logical ring, explicit token passing along a logical chord and implicit token passing along the logical ring. However, only the active stations bypassed by the chord are allowed to respond during the response window following an attempt to pass a token along a chord.

The IEEE 802.4 scheme uses only the first method, i.e., the explicit token passing along the logical ring, to pass the token. Consequently, the token has to pass through all the stations of the logical ring in a given cycle, inspite of the fact that some nodes may not have packets to transmit and some of them may be restrained from transmitting packets by the priority scheme. In the modified scheme, the token is able to bypass most such nodes, resulting in an improved channel utilization.

The priority mechanism operates very similarly to that proposed in IEEE 802.4 scheme¹³. The token-hold timer (THT) and the token-rotation timers (TRTIM(i), $i=0, \dots, P-2$) at a station, all run concurrently counting downward from an initial value to zero, at which point they stop counting and are said to be expired. Each access class also has a queue of frames to be transmitted. When a node receives the token, it first transmits the highest priority packets for a time not exceeding the limit set by HPHT, the high-priority token-hold time. Next, the station serves the token-rotation timers and queues from higher to lower priority classes as follows: The residual value of the token-rotation timer, TRTIM(i), is loaded into the token-hold timer,

THT, and the token-rotation timer is loaded with the target-token-rotation time, $TRT(i)$, for that class. If the THT is not expired, the station transmits packets of the current class until the packets are exhausted or the THT expires. When either event occurs, the station begins to service the next lower priority class. After the lowest class is serviced, the station performs any required logical ring maintenance functions and passes the token according to the method outlined above. When a token is successfully passed along a chord, the stations bypassed by that chord reload the token-rotation timers with target token-rotation times. A formal description of the priority scheme is given in Appendix A.

When $S_m = 1$, CS, the neighbor along the selected chord, and RS, the neighbor along logical ring, for a given station are the same, and, hence, the scheme is identical to the proposed IEEE 802.4 scheme. This requires the token to pass through every station in a given cycle, resulting in a lower bandwidth utilization.

In Section 5.3, the performance of the modified priority scheme is compared with the priority scheme of the IEEE 802.4 using simulation results. This priority scheme lacks the properties of hierarchical independence and fairness as discussed in Chapter 2. In the next section, we define a new priority scheme which possesses these properties.

3.3 THE TOKEN-SKIPPING PRIORITY SCHEME FOR TOKEN-PASSING BUSES

In the token-passing access scheme, the control of the bus is passed from station to station by means of a token. The nodes of a token-bus network can be arranged in a logical ring according to the order of addresses. In the normal token-passing scheme, this coincides with the path followed by the token¹³. In the presence of messages of different priorities, we may assume that the logical ring consists of several layers, one layer for each priority class. Figure 3.6 shows a four-layer logical ring corresponding to four distinct priority classes of messages.

Hierarchical independence can be achieved if the token is passed among the stations as follows: In the absence of higher classes of traffic, the token will move in the lowest layer of the logical ring. When messages are present in a higher layer, the token will go to that layer, transmit those messages and return to the lower layer at the same node at which it left the lower layer. The token may service messages of more than one class before returning to the original layer. An example of the path of the token is shown in Figure 3.7. Fairness is achieved because the right of access in a given priority class (layer) is passed in the order of addresses along the logical ring. Thus, if the conventional token-passing scheme is used, where the token is passed by a node to its neighbor, the token will have to pass through a large number of idle nodes. And this may drastically affect network performance. By allowing the token to skip idle stations, this

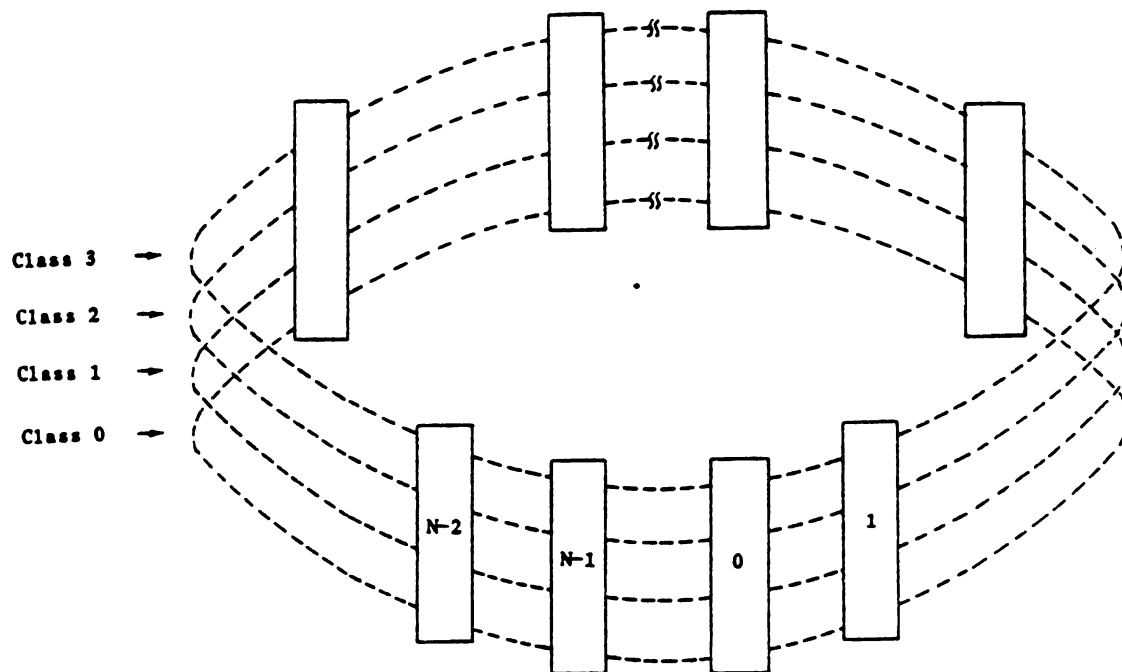


Figure 3.6 A four-layer logical ring.

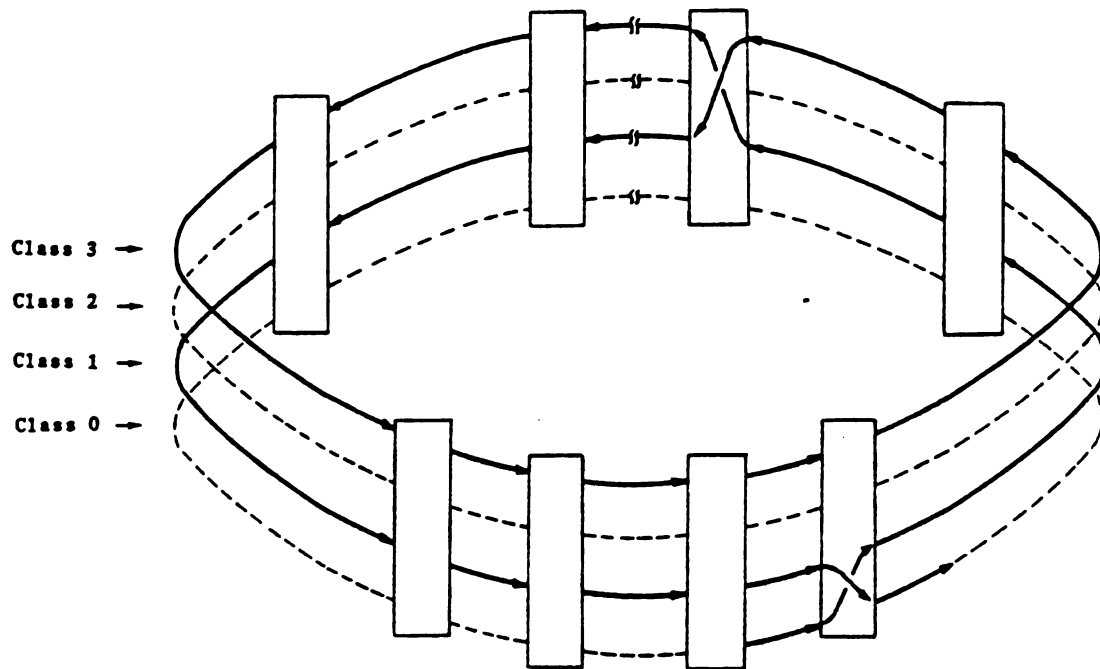


Figure 3.7 An example of the token-path in a hierarchically independent priority scheme.

overhead can be reduced. The scheme described below is aimed at achieving the above goals.

P classes of messages are assumed with 0 corresponding to the lowest priority and $P-1$ to the highest priority. Within each layer, token passing (intra-layer token passing) is carried out according to the token-skipping scheme described in Section 3.1, which has substantially improved delay throughput characteristics over the standard token-passing scheme, due to its ability to skip idle nodes. The priority scheme involves the transfer of the token between different layers (inter-layer token passing).

This scheme requires each station to have a ring flag and a chord station for each class of packets, i.e.,

$CS(i)$, $i=0, \dots, P-1$ — Address of a station at the other end of a selected chord passing through TS;

$R(i)$, $i=0, \dots, P-1$ — Ring flag, a one-bit flag indicating whether the token should be transferred along the logical ring or along the selected chord, i.e., to $CS(i)$.

In order to find out the highest class of packets available in the stations connected to the network, a control frame, priority assessment frame (PAF) is used. It carries the current priority level and is followed by several response windows. For example, a PAF with priority ($P-3$) is followed by up to two response windows. In the first, any station having a class $P-1$ packets will respond. If this is not busy, a second response window follows, in which any station with class ($P-2$) packets will respond.

The priority scheme makes use of the following flags at each node:

$MW(i)$, $i=1, \dots, P-1$ — Messages Waiting flag indicates the presence of untransmitted packets of the i th class at other nodes connected to the network. This is cleared each time the node sends the token of class i or is bypassed by one. This flag is set whenever another station sends a token with the $M(i)$ flag bit of the token set.

$MT(i)$, $i=1, \dots, P-1$ — Messages Transmitted flag indicates whether or not any packets of class i were transmitted by any station during the last cycle of the token. This is used to make the scheme robust under errors in case of multiple stations having TCON flags (described below) set. This is also reset when the node sends or is bypassed by a token of class i . It is set whenever a packet of class i appears on the bus.

$TCON(i)$, $i=1, \dots, P-1$ — Token Converted flag is set when the station converts a lower priority token to i th priority. When a node with $TCON(i) = \text{true}$ sends a token with priority i , it sets the $M(i)$ bit of the token. This ensures that the token does not skip the station that converted the token to class i . A node with $TCON(i) = \text{false}$ will set the $M(i)$ bit of the token only if it has untransmitted packets of class i in its buffers. This is reset when the token is converted back to the previous priority.

$PPR(i)$, $i=1, \dots, P-1$ — This is a storage location called the previous priority. Whenever a node converts the token to priority i , the priority of the current token is stored in $PPR(i)$. $TCON(i)$ and $PPR(i)$ enables the token to be returned to the previous ring at the same node which converted it to a higher priority.

Each station has a timer PAT (Priority Assessment Timer). Each station updates the PAT timer to its initial value, Inter-Priority Assessment Time (IPAT), which is the same for each node of the network, following the appearance of a priority assessment frame, PAF, on the bus. When this timer expires, the node currently holding the token opens a Priority Assessment Period (PAP) by broadcasting a PAF. This allows the messages of higher priorities to acquire the token. However, if the current priority is the highest, this timer is ignored. The upper bound on the time the higher classes of messages have to wait before requesting the token depends on IPAT.

When a node receives a token (of priority I), the following priority scheme procedure is executed.

1. If any packets of class I are present, send packets of class I until one of the following conditions occurs:
 - 1.1 Packets of class I are exhausted.
 - 1.2 $N(I)$ packets are transmitted, where $N(I)$ is the maximum number of packets of class I that may be transmitted once the token is received.
 - 1.3 When I is not the highest priority, the Priority Assessment Timer, PAT, expires.
2. The token is transferred from one layer to another, i.e., priority of the token is changed, if one of the following three situations occurs:
 - 2.1 I is not the highest priority, and the token has circulated in the current (Ith) layer more than the time limit set by the PAT, i.e. PAT has expired.
 - 2.2 The token returns to the node which converted the token to the current class and finds that no packets of class I are waiting at any node (i.e., $MW(I)$ is false) or if no packets of class I were transmitted during the last cycle

of the token (i.e., $MT(I)$ is false). The last condition provides robustness against situations when multiple nodes erroneously have their $TCON(I)$ flags set. In this situation set the priority to the previous priority, i.e., clear $TCON(I)$ flag, set the priority to $PPR(I)$, the priority of the token before it was converted to this class.

- 2.3 The station finds that no packets of current class are waiting, and none was transmitted during the last cycle of the token. This condition would not normally occur and is used to guard against the case of failure of the node which generated the token of class I, i.e., the one with $TCON(I)$ set, getting disabled.

For any of the above cases, the highest class with messages waiting is determined by transmitting a Priority Assessment Frame, PAF. If any of the response windows are busy, i.e., packets of a higher priority class are present, set the priority to the corresponding value, i.e., set $TCON(H)$ flag, where H is the highest priority class in which the packets are present, as determined by the PAF. Store the current priority in $PRP(H)$, i.e., $PRP(H) = I$. The new priority is H.

3. Send the token along the layer corresponding to the current priority using the intra-layer token-passing scheme described earlier.

Figure 3.8 gives an example of the channel activity under this scheme. In general, the channel will see a repetition of a sequence consisting of a PAF followed by packet and token transmissions of a single class. A transmission period of class I, $TP(I)$, is defined as a continuous time period during which only the packets and tokens of class I are transmitted. They are separated by Priority Assessment

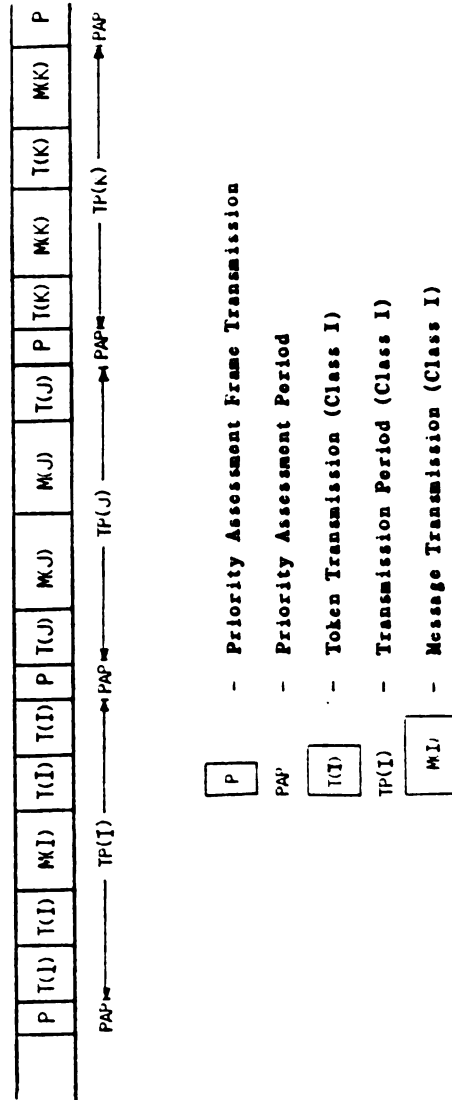


Figure 3.8 An example of the channel activity with the token-skipping priority scheme.

Periods (PAPs), and their duration is upper bounded by

$$TP(I) \leq IPAT + TMSG(I), \quad I=0, \dots, P-2 \quad (3.5)$$

where IPAT is the inter-priority assessment time and TMSG(I) is the maximum length of a packet of class I. For the highest class of priority, the transmission period, $TP(P-1)$, lasts as long as it is required to transmit all the messages of that class. A PAP is followed by a transmission period of the highest priority class for which messages are present.

In this chapter, we defined a new channel-access scheme for broadcast networks, which preserves the advantages of the token-passing channel-access scheme, but requires less overhead in terms of bandwidth for token-passing. In Chapter 4, an analytical model is developed for the token-passing scheme. Chapter 5 contains results of the analytical model and simulation models for the token-skipping scheme as well as the two priority schemes described above.

CHAPTER 4

AN ANALYTICAL MODEL FOR THE TOKEN-SKIPPING SCHEME

In Chapter 3, a channel-access scheme for broadcast networks was described. It preserves the advantages of token passing while reducing the overhead required to pass the token. In Section 2.3, the analysis of polling scheme developed by Konheim and Meister¹⁹ was explained. In this chapter, we develop an analytical model for the token-skipping scheme under exhaustive service discipline. In Section 4.1, the analysis of the polling scheme given in Section 2.3 is summarized. In Section 4.2, this analysis is extended for the standard token-passing scheme. An analytical model for token-skipping scheme is developed in Section 4.3.

4.1 ANALYSIS OF THE POLLING SCHEME

In this section, the notation and the basic assumptions used for modeling the polling scheme are restated and the results are summarized. Consider a network consisting of N buffered nodes connected using a single communication channel. The model in Section 2.3 used the following assumptions:

1. The load in the network is equally distributed among the nodes;
2. Message arrival at a node follows a Poisson distribution;
3. Packet time is constant;
4. The nodes are polled in sequence;
5. When a node is polled, the data is removed from the node buffer. When the buffer has been emptied, i.e., exhaustive service, the channel is used for system overhead for an interval called the reply interval;
6. The reply intervals are independent and identically distributed with mean r and variance δ^2 , and are also independent of the message arrival process.

The following notation is used:

- λ -- Mean packet arrival rate at a node in packets per packet time;
- N -- Number of nodes in the network;
- S -- The total throughput normalized with respect to the bandwidth;
- a_0 -- Slot time normalized with respect to the packet time i.e., a slot time is equal to a_0 packet times. Slot time is a small unit of time which is used to express the mean and variance of the token-passing time;
- r -- Mean time to pass the token between two stations in terms of slot times;
- δ^2 -- Variance of the token-passing time (δ is given in slot times);
- T_t -- Token-transmission time in slot times;
- T_{rw} -- Response window duration in slot times;

D — Mean delay (queuing and transmission) undergone by packets, normalized with respect to the packet transmission time.

From Section 2.3, when $r < \infty$ and $N\lambda < 1$, the throughput of the network is given by

$$S = N\lambda \quad (4.1)$$

and the stationary expected delay normalized with respect to the packet time is given by

$$D = 1 + \left[\delta^2 a_0 / 2r \right] + \left[S / 2(1-S) \right] + (a_0/2) \left[1-S/N \right] \left[1 + Nr/(1-S) \right]. \quad (4.2)$$

The stationary expected cycle length T and its variance Δ_T , normalized with respect to the packet time are given by

$$T = Nra_0 / (1-S) \quad (4.3)$$

$$\Delta_T^2 = \left[\delta^2 Na_0^2 / (1-S/N)(1-S) \right] + \left[ra_0 SN / (1-S/N)(1-S)^2 \right]. \quad (4.4)$$

Under Poisson distribution, the probability that k packets arrive at a node in a time interval of t packet times is given by

$$P_k(t) = (\lambda t)^k e^{-\lambda t} / k!. \quad (4.5)$$

In the next section, we extend this analysis to model the standard token-passing scheme. In Section 4.3, an analytical model is developed for the token-skipping scheme under exhaustive service discipline.

4.2 ANALYSIS OF THE TOKEN-PASSING SCHEME

In the standard token-passing scheme, the token is passed from node to node in a sequence. This is equivalent to polling each node in the network in a sequence. Hence the analysis of Section 4.1 can be used for the token-passing scheme, with the time required to pass the token from one node to the next corresponding to the reply interval. Assuming the time required for passing the token to the next station be a constant, T_t slot times. i.e. ,

$$r = T_t \quad (4.6)$$

and

$$\delta^2 = 0. \quad (4.7)$$

Hence, from Equation (4.2), when $N\lambda < 1$, the mean delay D , undergone by messages is given by

$$D = 1 + \left[S / 2(1-S) \right] + (a_0/2) \left[1 - S/N \right] \left[1 + Nr/(1-S) \right]. \quad (4.8)$$

In the next section, we develop an analytical model for the token-skipping scheme.

4.3 ANALYSIS OF THE TOKEN-SKIPPING SCHEME

In the token-skipping scheme, the token is not passed from node to node in a sequence. However, the token is allowed to skip nodes only if they do not have any messages. For the purpose of analysis, this can be considered as equivalent to polling each of the nodes bypassed by the chord along which the token is passed. Hence we may consider passing of the token from a node to another node at a logical distance s , along a logical chord, in a time T , as being equivalent to passing the token through each of the skipped stations sequentially, with each token pass requiring T/s time units. Now we can use the model of Section 4.1 to analyze the token-skipping scheme, under exhaustive service of messages. However, the assumption that the token-passing times are independent, identically distributed and also independent of the message arrival processes is not strictly valid in this case.

Four approximations are used in the analysis of the token-skipping scheme. The first is the assumption that the token-passing times are independent and identically distributed. The

other three approximations are used in calculating the mean token-passing time and are described later. The first approximation allows the equations derived in Section 2.3, and given in Section 4.1 to be readily used for the analysis of the token-skipping scheme. First the mean and variance of token-passing time among the nodes have to be found.

As we consider the case of exhaustive service, the chord along which the token is passed always points to the node at a distance S_m from the source station, where S_m is the maximum skip distance. The token-skipping scheme under exhaustive service can be briefly explained as follows. Following the transmission of messages, a node will send a token addressed to a station at a logical distance S_m . This is followed by a response window during which any node that is bypassed by the token, which has messages to be transmitted, will respond. If there are no such stations, i.e., the response window is idle, the destination node will accept the token. However, if the response window is busy, then the node next to the source station on the logical ring accepts the token. Now the token travels along the logical ring until the node with messages is encountered.

Let p_s be the probability that an attempt to pass the token along a chord is successful, i.e.,

$$p_s = \text{Prob}[\text{An attempt to pass the token} \\ \text{along a chord is successful}]. \quad (4.9)$$

In the analysis below, only the case where $S_m \geq 2$ is considered. For $S_m = 1$, the results are same as for the standard token-passing scheme given in Section 4.2. In order to calculate the mean and variance of the token-passing times, consider the path of the token in the two cases where an attempt to pass the token along a chord succeeds and fails.

1. An attempt to pass the token is successful. This event has a probability p_s . In this case, shown in Figure 4.1, the token is passed to a station at a distance S_m in a time $(T_t + T_{rw})$. Hence the token can be assumed to have passed through each of the intervening stations with a token-passing time of $(T_t + T_{rw})/S_m$.
2. An attempt to pass the token along the logical chord is unsuccessful. This event has a probability of $(1 - p_s)$. In this case, shown in Figure 4.2, the token is passed on to the next station in a time $(T_t + T_{rw})$. Then the token travels along the logical ring until a node with messages present is found. This station is at a distance between 1 and $(S_m - 1)$ from the source station. Here, the second approximation is made, which is, the distance token travels along the logical ring is $S_m/2$. This means one token pass occurs with time $(T_t + T_{rw})$ followed by $(S_m/2 - 1)$ token passes requiring a time T_t each. Since this is an approximation, this can be assumed for both even and odd values of S_m . In the actual case however, the token may travel a distance anywhere between 1

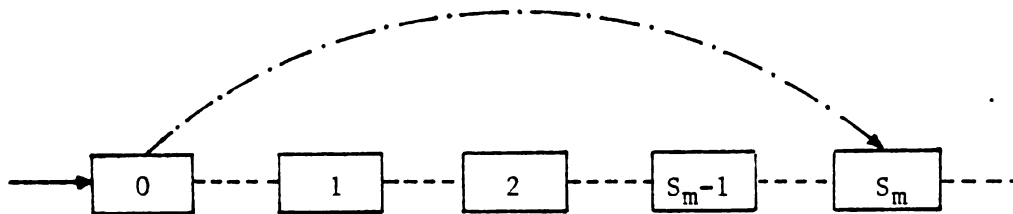


Figure 4.1 A successful attempt to pass the token along a logical chord.

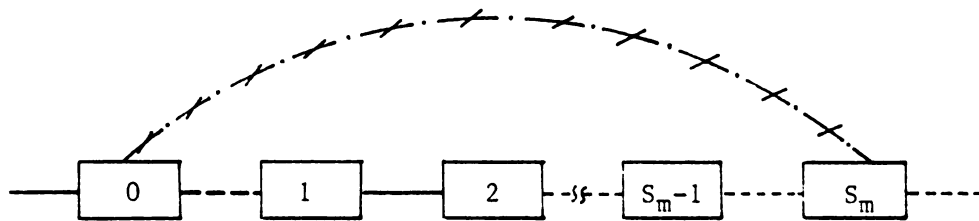


Figure 4.2 An unsuccessful attempt to pass the token along a logical chord.

and $(S_m - 1)$ before encountering a node with untransmitted messages. Approximating this by $(S_m/2)$ seems to yield fairly accurate results, especially for small S_m . For the case $S_m = 2$, this is exactly the case.

Hence the stationary probability distribution $p(R)$ of the token-passing time, R , is given by

$$p(R) = \begin{cases} p_s S_m / [(1+p_s)(S_m/2)] & \text{when } R = (T_t + T_{rw})/S_m \\ (1-p_s) / [(1+p_s)(S_m/2)] & \text{when } R = (T_t + T_{rw}) \\ (1-p_s)[(S_m/2)-1] / [(1+p_s)(S_m/2)] & \text{when } R = T_t \\ 0 & \text{elsewhere.} \end{cases} \quad (4.10)$$

Hence the mean and variance of the token-passing time are given by

$$r = \left[T_{rw} + p_s T_t + (1-p_s)(S_m/2)T_t \right] / \left[(S_m/2)(1+p_s) \right] \quad (4.11)$$

and

$$\delta^2 = \left[[p_s(T_t + T_{rw})^2/S_m] + (1-p_s)[(T_t + T_{rw})^2 + ((S_m/2) - 1)T_t^2] \right] / \left[(1+p_s)S_m/2 \right] - r^2. \quad (4.12)$$

Rearranging (4.11),

$$p_s = [A - r] / [B + r] \quad (4.13)$$

where,

$$A = T_t + 2T_{rw}/S_m \quad (4.14)$$

and

$$B = T_t - 2T_t/S_m. \quad (4.15)$$

Now it remains to find p_s , the probability of success in passing the token along a chord. Consider the case of node 0 attempting to pass the token to the node S_m as shown in Figure 4.1.

From Equation (4.9),

$$\begin{aligned} p_s &= \text{Prob}[\text{Success of an attempt to pass the token along a chord}] \\ &= \text{Prob}[\text{No station bypassed by the chord has packets}]. \end{aligned}$$

Since the message arrivals at nodes are independent of each other,

$$\begin{aligned} p_s &= \prod_{i=1}^{i=S_m-1} \text{Prob}[\text{no packets arrived at node } i \\ &\quad \text{since the last token passed it}] \\ &= \prod_{i=1}^{i=S_m-1} \text{Prob}[\text{no packets arrived at node } i \text{ in a time } T_i] \end{aligned}$$

where T_i is the time elapsed since the token last passed node i . T_i

also is the time the token required to go through the last $(N-i)$ nodes. Using Equation (4.5),

$$p_s = \prod_{i=1}^{i=S_m-1} \exp[-\lambda T_i]$$

i.e.,

$$p_s = \exp\left[-\lambda \sum_{i=1}^{i=S_m-1} T_i\right]. \quad (4.16)$$

Here we make our third approximation, namely,

$$T_i = [(T + \Delta_T)(N-i) / N] \quad (4.17)$$

i.e., the time it takes the token to go through the last $(N-i)$ nodes is approximated by $(T + \Delta_T)(N-i)/N$. The stationary average time it takes the token to circulate around the logical ring is T , which yields the stationary expected value of T_i to be $T(N-i)/N$. However, use of this value has a tendency to over estimate p_s . The selection of T_i as in Equation (4.17) yields fairly accurate results as seen by the simulation results given in Section 5.2. T and Δ_T are given by Equations (4.3) and (4.4) respectively as

$$T = Nra_0 / (1-S) \quad (4.3)$$

and

$$\Delta_T^2 = \left[\delta^2 N a_0^2 / (1-S/N)(1-S) \right] + \left[r a_0 S N / (1-S/N)(1-S)^2 \right]. \quad (4.4)$$

Except for very small values of S , e.g., $S < .1$, the first term in Equation (4.4) can be neglected compared to the second. Hence, we approximate Δ_T by

$$\Delta_T^2 = \left[r a_0 S N / (1-S/N)(1-S)^2 \right]. \quad (4.18)$$

Solving Equations (4.3), (4.16), (4.17) and (4.18), we get

$$p_s = e^{-k a_0 r [1 + (k'/r)^{1/2}]} \quad (4.19)$$

where k and k' are given by

$$k = \left[[(S/N)(S_m - 1)(2N - S_m)] / 2(1-S) \right] \quad (4.20)$$

$$k' = \left[(S/N) / a_0(1 - S/N) \right]. \quad (4.21)$$

Hence, using Equations (4.12) and (4.19) yields

$$e^{-k a_0 r [1 + (k'/r)^{1/2}]} = (A-r)/(B+r). \quad (4.22)$$

Summarizing, we have

$$e^{-ka_0 r [1 + (k'/r)^{1/2}]} = (A-r)/(B+r) \quad (4.22)$$

where

$$A = T_t + 2T_{rw}/S_m \quad (4.14)$$

$$B = T_t - 2T_t/S_m \quad (4.15)$$

$$k = \left[\frac{[(S/N)(S_m-1)(2N-S_m)]}{2(1-S)} \right] \quad (4.20)$$

and

$$k' = [(S/N) / a_0(1 - S/N)]. \quad (4.21)$$

Hence, knowing the parameters T_t , T_{rw} , S_m and N , for a given value of S , Equations (4.14), (4.15), (4.20) and (4.21) can be used to evaluate the constants A , B , k and k' respectively. Now Equation (4.22) can be solved numerically to yield r , the mean token-passing time. This value of r can be used to calculate p_s , the probability of success in passing a token along a chord and the variance of the token-passing time using Equations (4.13) and (4.12) given below.

$$p_s = [A - r] / [B + r] \quad (4.13)$$

$$\delta^2 = \left[\frac{[p_s(T_t + T_{rw})^2 / S_m] + (1 - p_s)[(T_t + T_{rw})^2 + ((S_m/2) - 1)T_t^2]}{(1 + p_s)S_m/2} \right] - r^2 \quad (4.12)$$

Now Equation (4.2) can be used to find the stationary expected queuing delay of the network.

$$D = 1 + \left[\delta^2 a_0 / 2r \right] + \left[S / 2(1 - S) \right] + (a_0/2) \left[1 - S/N \right] \left[1 + Nr / (1 - S) \right] \quad (4.2)$$

In this chapter, we developed an analytical model for the token-skipping scheme under the following assumptions:

1. The traffic is symmetrically distributed among the nodes;
2. The packet arrival at nodes follow Poisson distribution;
3. Fixed packet size;
4. Exhaustive service of messages.

In the next chapter, we present simulation results comparing the token-skipping scheme and the proposed priority schemes with the IEEE 802.4 channel-access protocol. In Section 5.2, the results from the analytical model are compared with those obtained using simulations.

CHAPTER 5

RESULTS AND INTERPRETATION

Chapter 3 defined the token-skipping channel-access scheme for broadcast networks and two methods by which priorities can be implemented using the token-skipping scheme. In Chapter 4, an analytical model was developed for the token-skipping scheme under exhaustive service of messages. In this chapter we present simulation results comparing the token-skipping channel-access scheme with the IEEE 802.4 channel-access scheme and also comparing the analytical model with simulation results. Message arrival at nodes is assumed to follow a Poisson process. The set of nodes forming the logical ring is assumed to be stationary, i.e., the simulation does not consider the case where stations switch on and off the logical ring.

5.1 TOKEN-SKIPPING SCHEME UNDER NON-EXHAUSTIVE SERVICE OF MESSAGES

In this section, we provide results of simulations to compare the token-skipping scheme and the IEEE 802.4 scheme under round robin service discipline, i.e., each node is allowed to transmit only one packet of data upon the reception of the token. Poisson arrival of messages at each station is assumed. The network parameters used for the simulation are given in Table 5.1. These parameters were selected

Table 5.1 Network parameters used for comparing the token-skipping scheme and the IEEE 802.4 channel-access scheme with round-robin service discipline.

Number of stations, N	= 50 and 100
Bandwidth of the bus	= 1 Mbps
Bus length	= 1 km
Round trip propagation delay	= 10 μ s
Message length (fixed)	= 1000 bits
Size of response window	= 25 μ s
Token transmission time	= 100 μ s
Maximum skip distance	= 1, 3, 5 and 10

based on the values proposed in the IEEE 802.4 standards¹³ and other publications related to local area networks^{4,10}. The token-skipping scheme and the IEEE 802.4 scheme are compared using the speedup factor defined as the ratio of the mean delays undergone by messages in IEEE 802.4 scheme and the token-skipping scheme. Two networks, one with 50 stations and the other with 100 stations, were simulated. The maximum skip distance, S_m was selected to have values 1 (corresponding to the standard token bus), 3, 5 and 10. Simulation was carried out for two types of traffic distributions among stations.

1. Symmetric traffic distribution: The mean message arrival rate at each station is assumed to be the same. The variation of mean delay vs. the throughput for $N = 50$ and 100 are shown in Figures 5.1 and 5.2 respectively with the corresponding speedup factors given in Table 5.2.
2. Asymmetric traffic distribution: In this case, 20% of the stations (selected consecutively) were responsible for 80% of the traffic offered to the network. Figures 5.3 and 5.4 show the delay throughput characteristics for the cases $N = 50$ and 100 respectively. The corresponding speedup factors are given in Table 5.3.

The token-skipping scheme provides speedup over the standard token-passing scheme for all the values of $S_m > 1$ for throughputs up to about 80%. At higher loads under symmetric traffic, the standard token-passing scheme fares better. When the number of idle nodes in the network is high, e.g., at light load or under asymmetric load

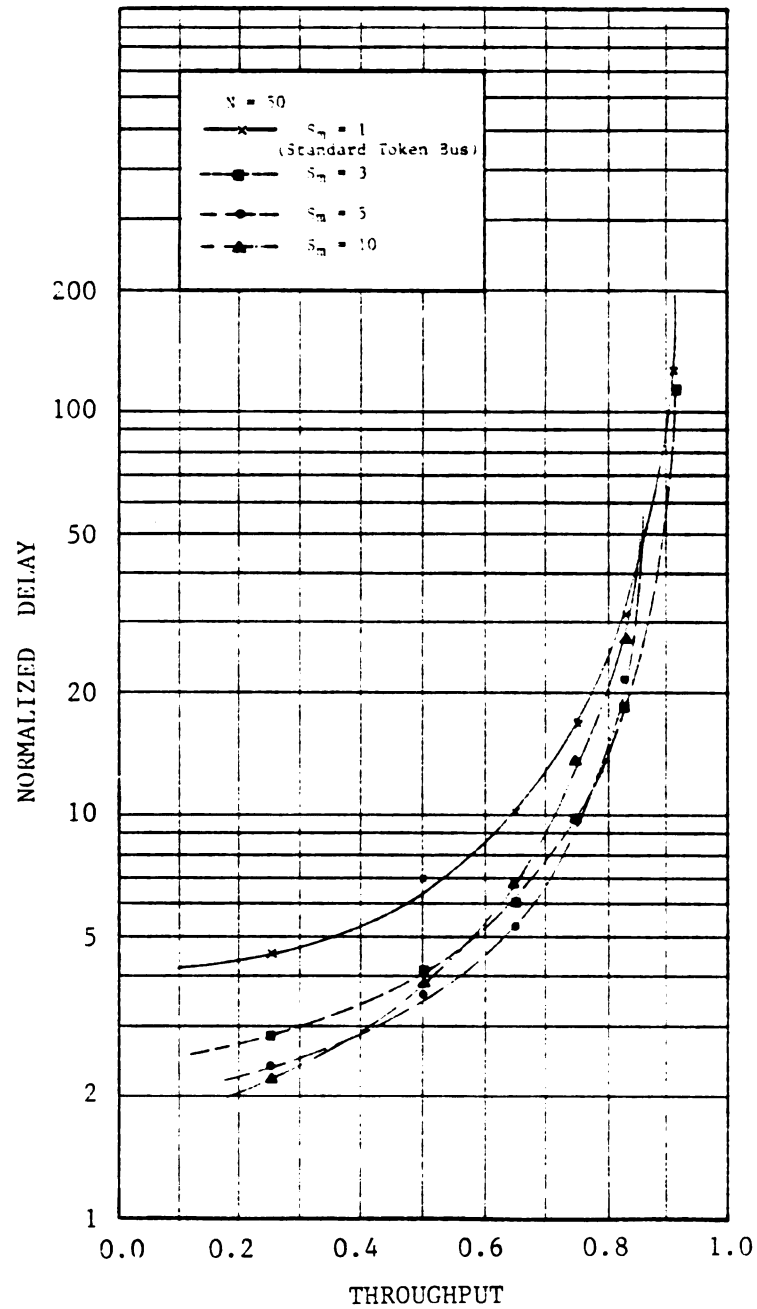


Figure 5.1 Delay-throughput characteristics under symmetric load distribution for a network with 50 stations.

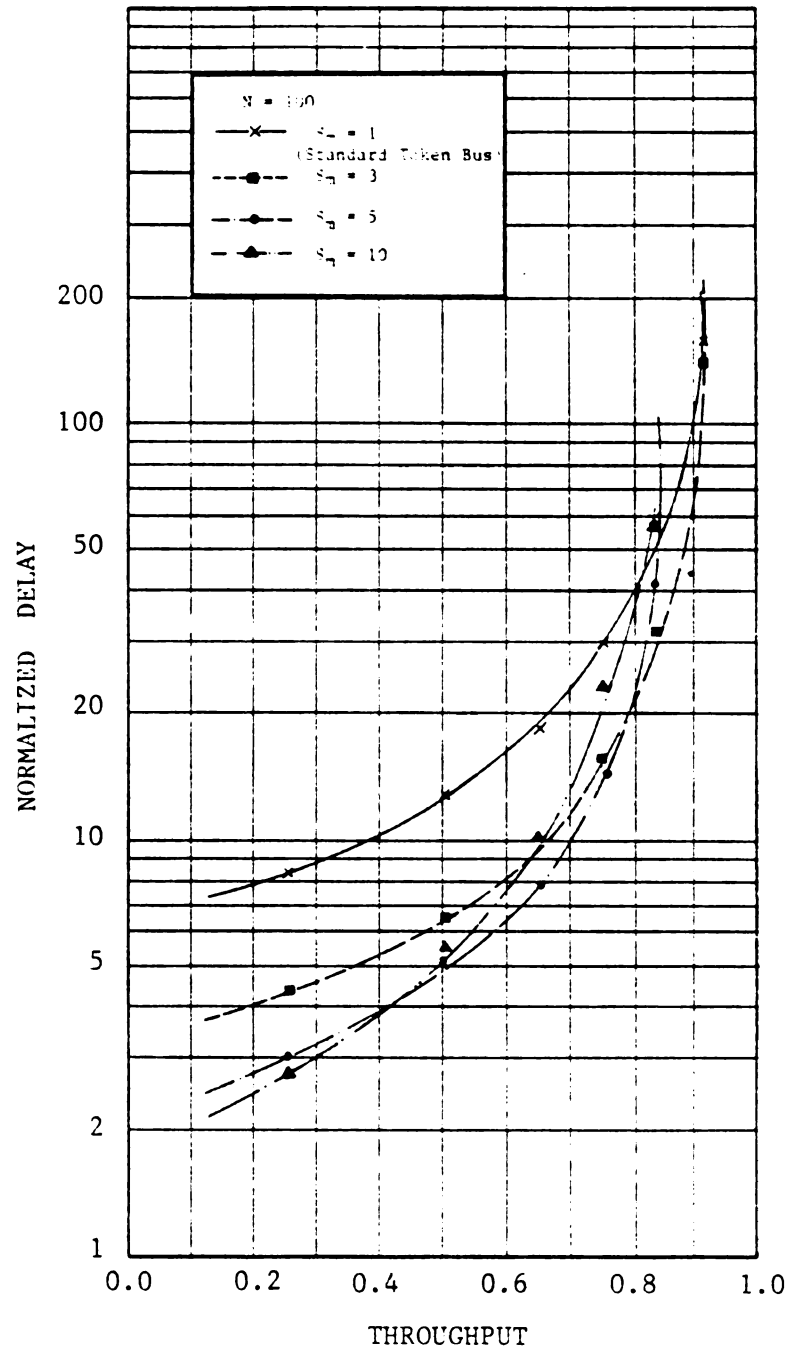


Figure 5.2 Delay-throughput characteristics under symmetric load distribution for a network with 100 stations.

Table 5.2 Speedup factors under symmetric traffic distribution.

THROUGHPUT	SPEEDUP FACTOR					
	N = 50			N = 100		
	$S_m = 3$	$S_m = 5$	$S_m = 10$	$S_m = 3$	$S_m = 5$	$S_m = 10$
.25	1.6	1.9	2.1	1.9	2.6	3.0
.50	1.7	2.0	1.8	2.0	2.5	2.4
.65	1.7	1.9	1.5	2.0	2.3	1.9
.75	1.7	1.7	1.2	1.9	2.0	1.3
.83	1.6	1.4	1.1	1.8	1.4	1.0

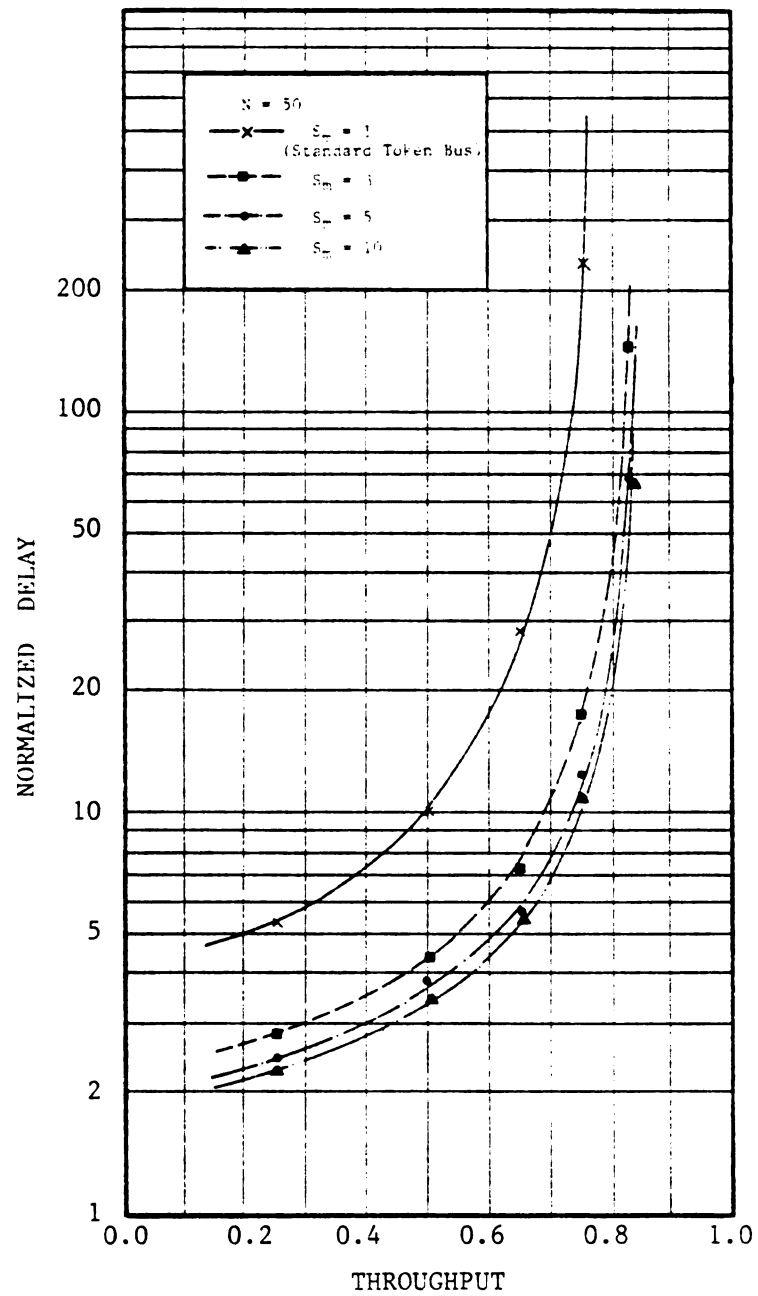


Figure 5.3 Delay-throughput characteristics under asymmetric load distribution for a network with 50 stations.

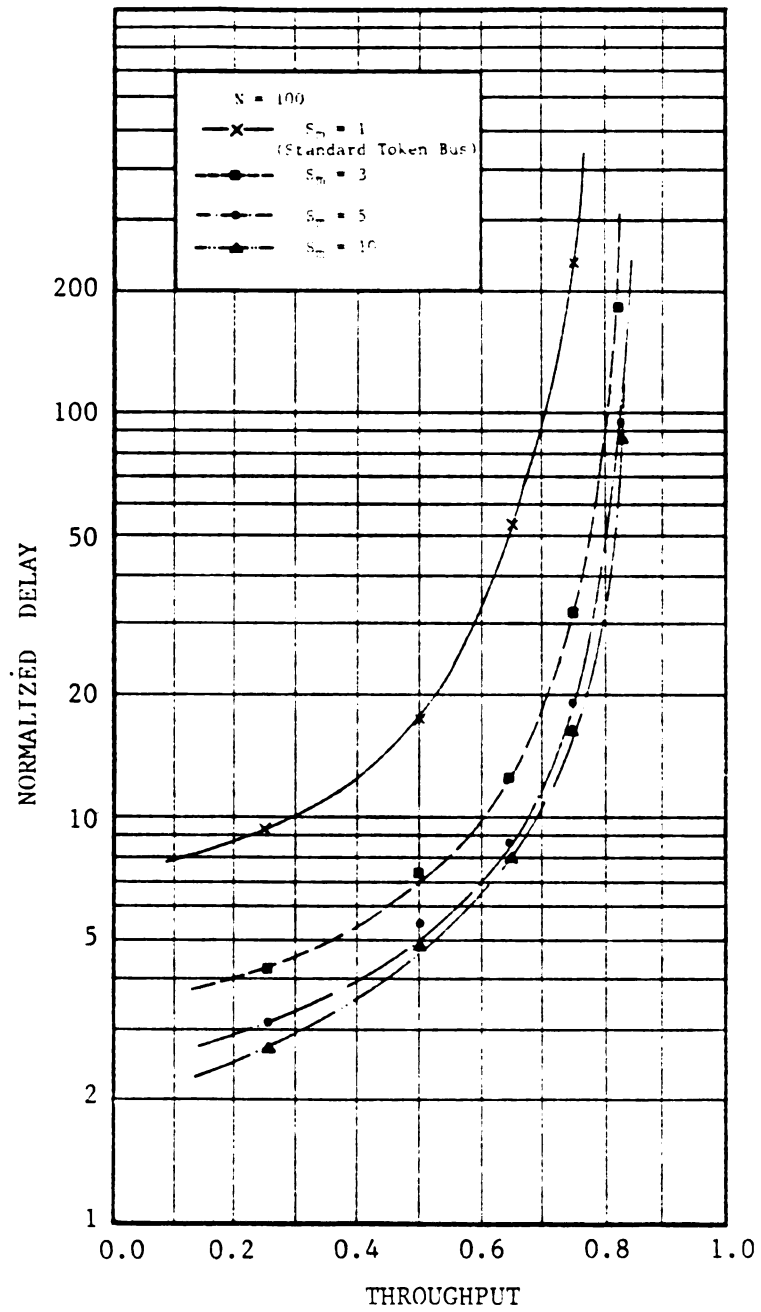


Figure 5.4 Delay-throughput characteristics under asymmetric load distribution for a network with 100 stations.

Table 5.3 Speedup factors under asymmetric traffic distribution.

THROUGHPUT	SPEEDUP FACTOR					
	N = 50			N = 100		
	$S_m = 3$	$S_m = 5$	$S_m = 10$	$S_m = 3$	$S_m = 5$	$S_m = 10$
.25	2.2	2.9	3.4	1.9	2.2	2.3
.50	2.4	3.2	3.6	2.2	2.6	2.9
.65	4.3	6.0	6.6	3.9	5.0	5.3
.75	7.4	12.1	14.3	13.3	20.0	21.4

distribution among nodes the token-skipping scheme provides a high speedup factor. For example, simulation results show a speedup of 20 on a network with 100 stations at a throughput of 75% under the asymmetric load distribution. The increase in speedup with the increase in load under the asymmetric traffic distribution is due to the rapid increase in delay associated with the standard token-passing scheme under these circumstances. This can be attributed to the fact that at a given time there are more idle nodes in asymmetric case than under symmetric load distribution.

5.2 TOKEN-SKIPPING SCHEME UNDER EXHAUSTIVE SERVICE OF MESSAGES

In this section, we compare the delay-throughput characteristics of the token-skipping scheme under exhaustive service discipline obtained via simulations and the analytical model developed in Chapter 4. The network parameters used for this purpose are shown in Table 5.4. The traffic is assumed to be symmetrically distributed among the nodes with packet arrivals following a Poisson process. Figures 5.5, 5.6 and 5.7 show the delay-throughput characteristics for a network having 50 nodes when the maximum skip distance is 3, 5 and 10, respectively. Figures 5.8, 5.9 and 5.10 shows the corresponding results for a network with 100 nodes. These results show that the analytical model is able to predict the delay undergone by messages with reasonable accuracy when the ratio S_m/N is small. As S_m increases, the results of the analytical model tend to diverge from the

Table 5.4 Network parameters used for validating the analytical model
for the token-skipping scheme.

Number of stations, N	=	50, 100
Number of priority classes, P	=	4
Bandwidth of the bus	=	1 Mbps
Length of the bus	=	1 km
Round trip propagation delay, T_p	=	10 μ s
Packet length, including header ^p	=	1000 bits
Size of response window, T_{rw}	=	20 μ s
Token transmission time	=	100 μ s

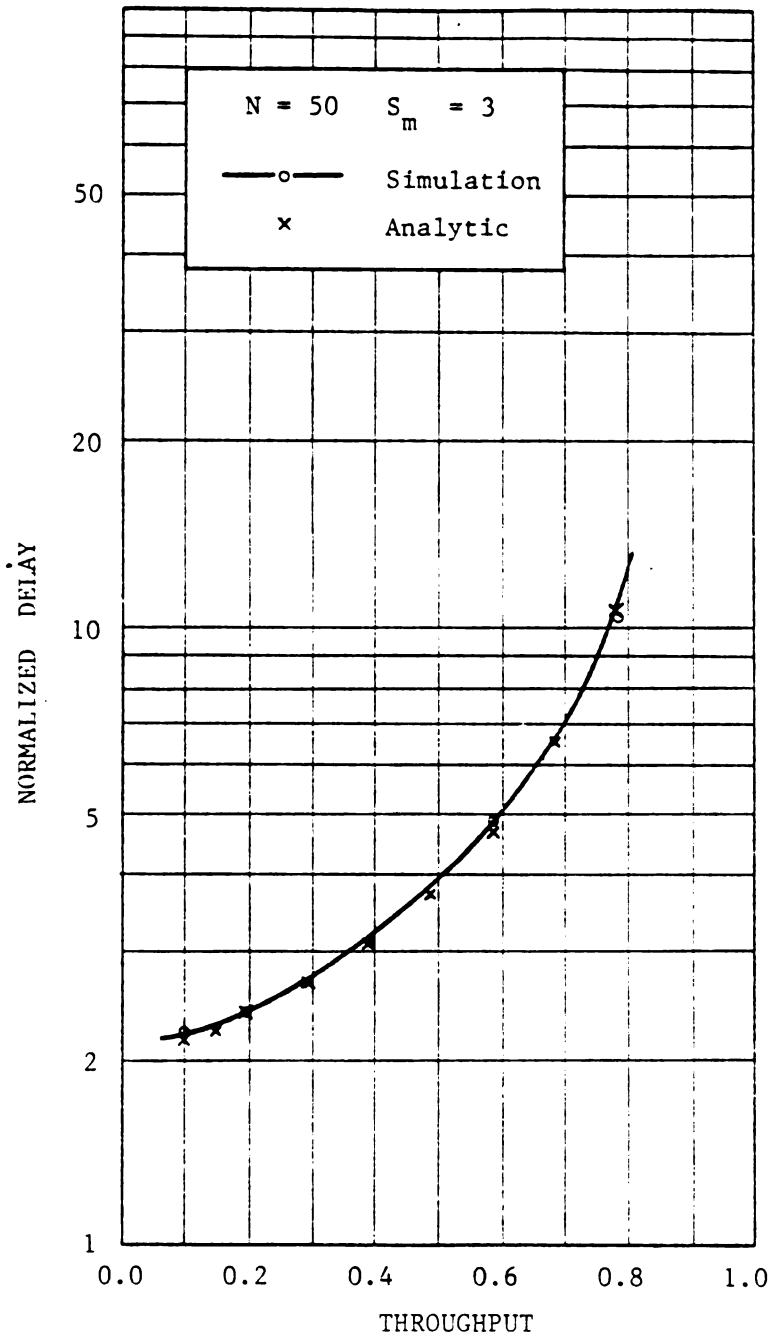


Figure 5.5 Delay-throughput characteristics for a network with $N = 50$, $S_m = 3$ and exhaustive service discipline.

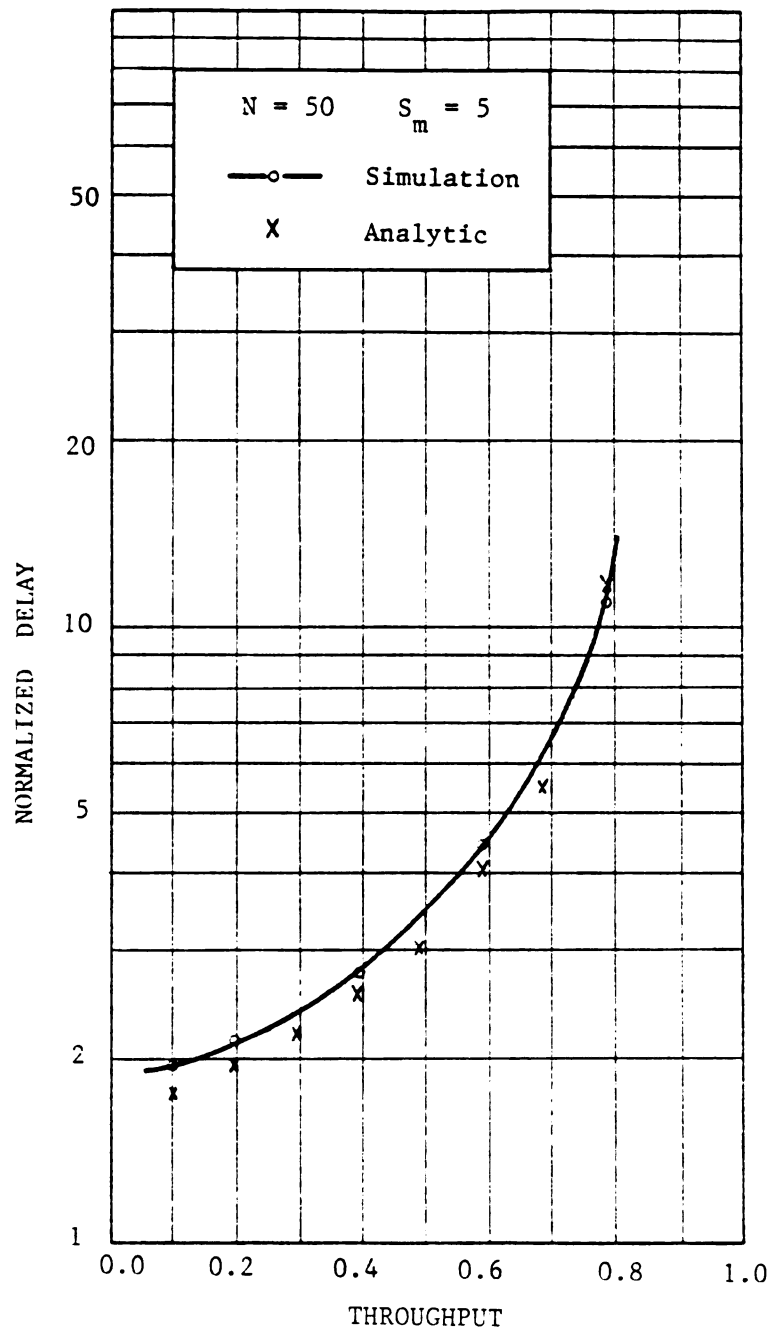


Figure 5.6 Delay-throughput characteristics for a network with $N = 50$, $S_m = 5$ and exhaustive service discipline.

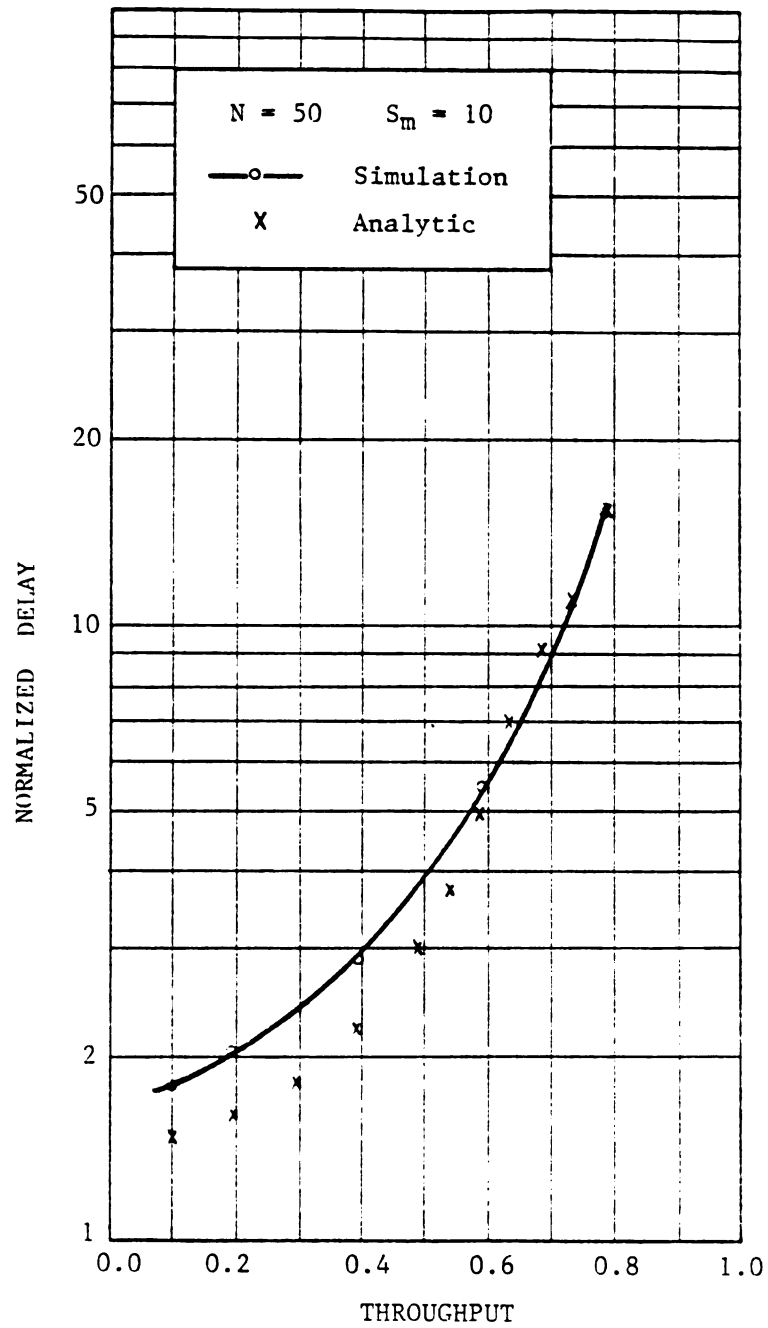


Figure 5.7 Delay-throughput characteristics for a network with $N = 50$, $S_m = 10$ and exhaustive service discipline.

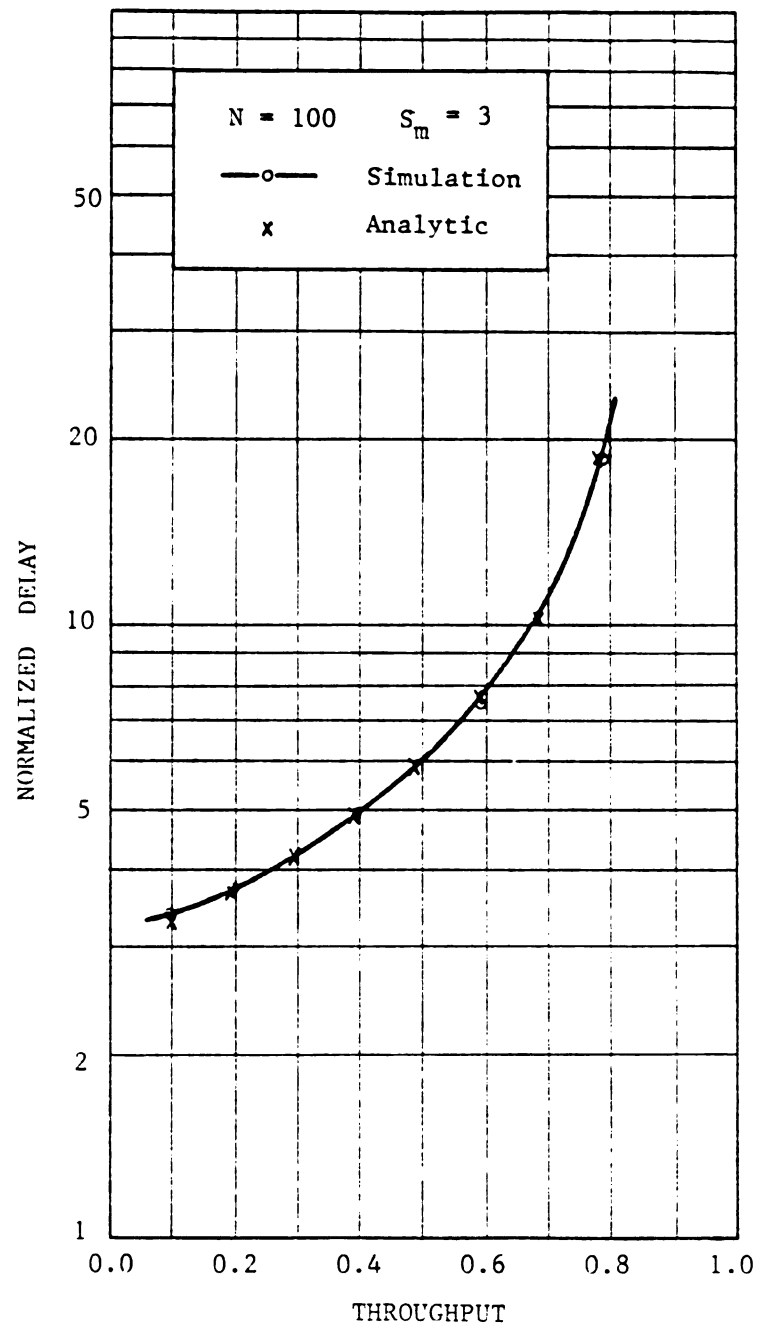


Figure 5.8 Delay-throughput characteristics for a network with $N = 100$, $S_m = 3$ and exhaustive service discipline.

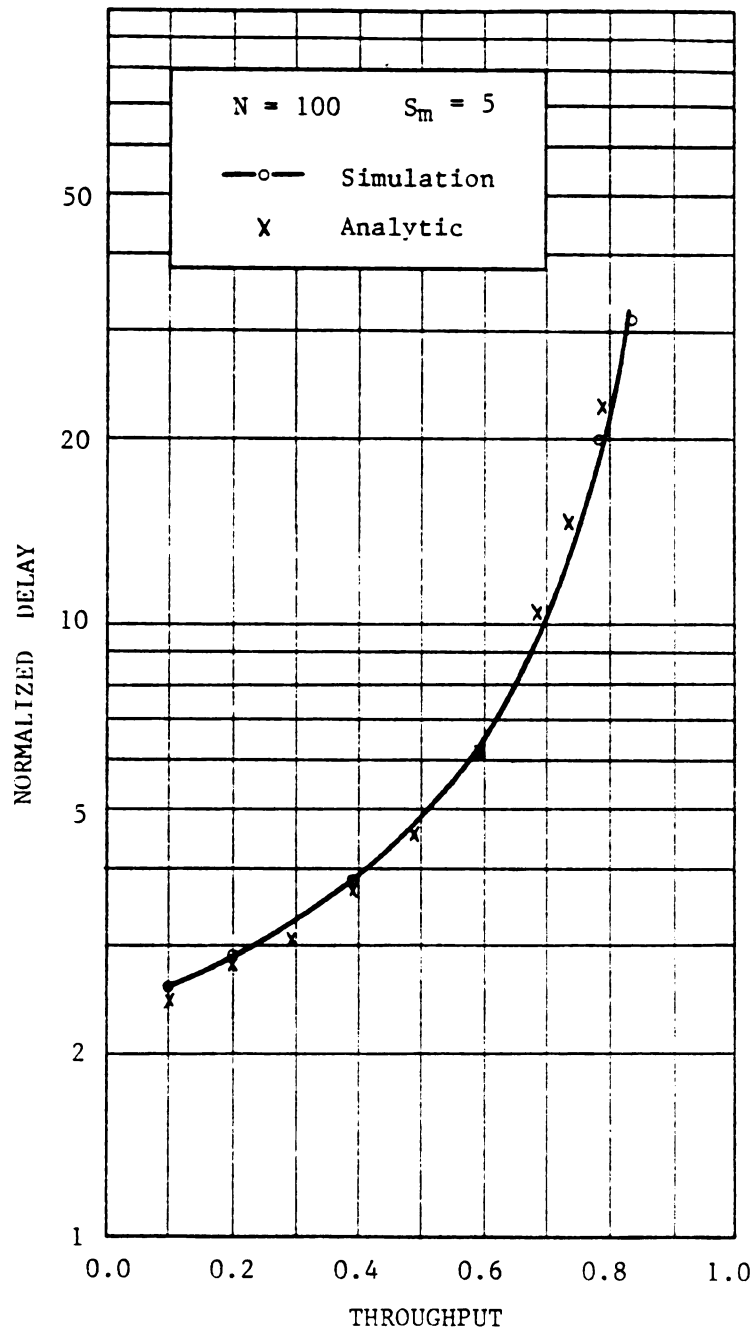


Figure 5.9 Delay-throughput characteristics for a network with $N = 100$, $S_m = 5$ and exhaustive service discipline.

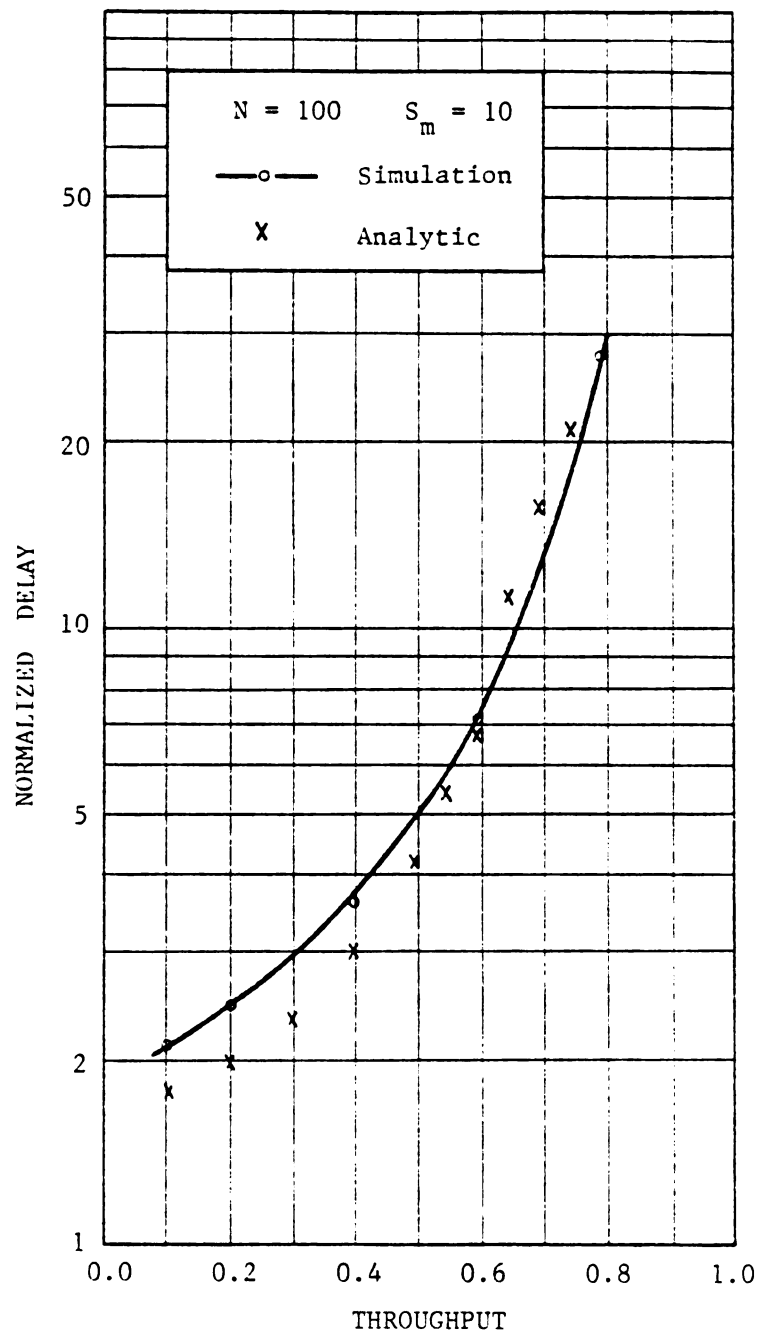


Figure 5.10 Delay-throughput characteristics for a network with $N = 100$, $S_m = 10$ and exhaustive service discipline.

simulation results. This can be attributed to the assumptions made in deriving the analytical model. However, selecting a large value for S_m will degrade the performance of the network. Hence this analytical model can be used to predict the performance of the network for practical values of S_m for a given network.

5.3 MODIFIED PRIORITY SCHEME FOR TOKEN-PASSING BUSES

Here, we present the results of a simulation of the IEEE 802.4 priority scheme and the modified priority scheme described in Section 3.2 for a network with four classes of messages present. All the simulations given in this section have made use of the network parameters given in Table 5.5. Poisson arrival of messages at each node and class is assumed. And traffic distribution among stations is assumed to be symmetric.

The following notation is used in this section.

$G(i)$ — Offered load in Class i normalized with respect to the bandwidth;

G — Total offered load (normalized) = $\sum_i G(i)$;

$S(i)$ — Throughput of messages of Class i normalized with respect to the bandwidth;

S — Total throughput of the network (normalized) = $\sum_i S(i)$;

$T(i)$ — Mean delay (queuing + transmission) undergone by messages of Class i normalized with respect to packet time.

Table 5.5 Network parameters used for comparing the modified priority scheme with the IEEE 802.4 priority scheme.

Number of stations, N	=	50
Number of priority classes, P	=	4
Bandwidth of the bus	=	1 Mbps
Length of the bus	=	1 km
Round trip propagation delay, T_p	=	10 μ s
Packet length, including header	=	1000 bits
Size of response window, T_{rw}	=	20 μ s
Token-transmission time	=	100 μ s
High-priority token-hold time, HPTHT	=	1000 μ s
Target-token-rotation time for		
Class 0, TRT(0)	=	10000 μ s
Class 1, TRT(1)	=	15000 μ s
Class 2, TRT(2)	=	20000 μ s
Tolerance of		
Class 0, TOL(0)	=	120 μ s
Class 1, TOL(1)	=	120 μ s
Class 2, TOL(2)	=	120 μ s

The delay-throughput characteristics are given in Figure 5.11 for each class, $T(i)$ vs. $S(i)$, $i=0, \dots, 3$. This simulation assumed that all the four classes have the same traffic intensity, i.e., $G(0)=G(1)=G(2)=G(3)=G/4$. The variation of total throughput, S , vs. offered load, G , for this case is shown in Figure 5.12.

Figure 5.13 shows the delay-throughput characteristics when all the messages belong to a single class. It gives the simulation results of the cases where all the messages belong to Class 3 (i.e., $G(0)=G(1)=G(2)=0$ and $G(3)=G$) and where all the messages belong to Class 1 (i.e., $G(0)=G(2)=G(3)=0$ and $G(1)=G$).

Simulation of the case when the traffic is equally distributed among the different classes, shows that the modified priority scheme provides a significant improvement in delay-throughput characteristics for all the classes of messages compared to the IEEE 802.4 scheme. The total throughput of the modified scheme is always greater than or equal to that of the IEEE 802.4 scheme, and under certain conditions it is larger by as much as 0.15 of the bandwidth. This is to be expected since, in the IEEE 802.4 scheme, the token passes through a large number of nodes without sending a packet due either to the node not having any packets or the token-rotation timers (TRTIM's) of the classes of messages present being expired. In the modified scheme, the token skips most such nodes.

The simulation of extreme cases, where messages present belong only to one of the four classes (Figure 5.13), also show the modified scheme to be superior with higher throughputs and lower delays. The

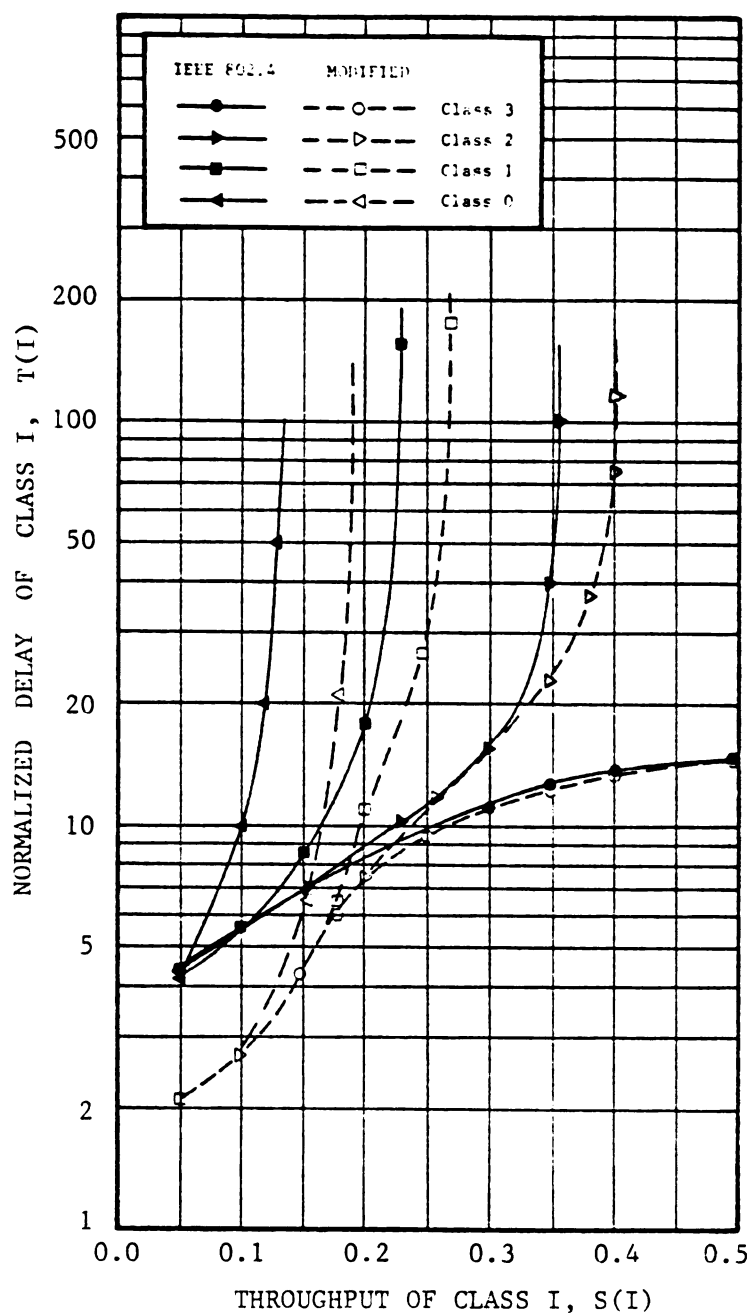


Figure 5.11 Delay-throughput characteristics [$T(I)$ vs. $S(I)$, $I=0, \dots, 3$] for the modified priority scheme when the traffic is equally distributed among the priority classes [i.e., when $G(I) = G/4$, $I=0, \dots, 3$].

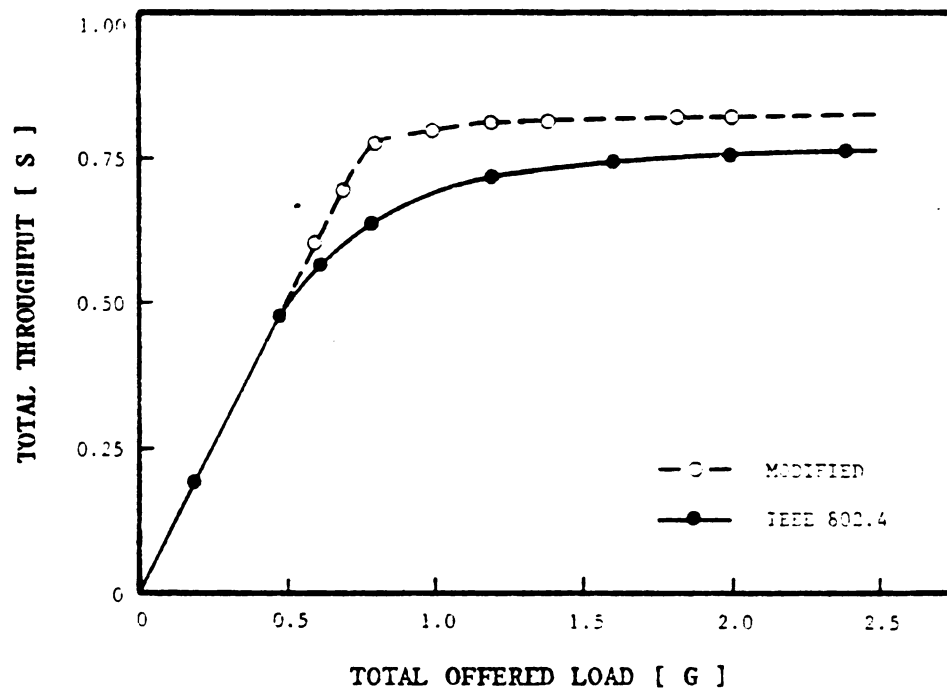


Figure 5.12 Total throughput vs. total offered load [S vs. G] for the modified priority scheme when the traffic is equally distributed among the priority classes.

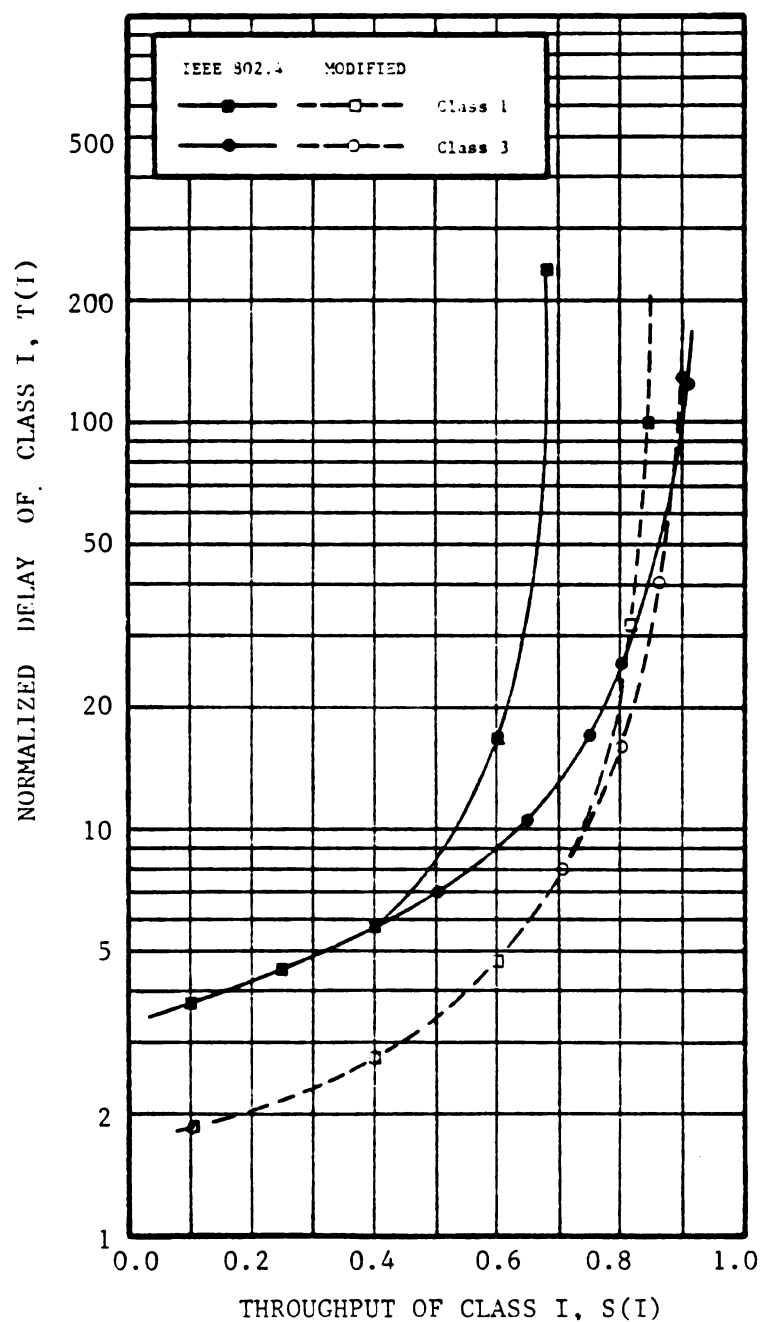


Figure 5.13 Delay-throughput characteristics [$T(I)$ vs. $S(I)$, $I=1,3$], for the modified priority scheme when all the traffic belongs to class I.

only case where the modified scheme fares worse than the IEEE 802.4 scheme is when all the messages belong to the highest class of priority and the traffic intensity is very high. But this too may be remedied by lowering the value of S_m under such conditions. In fact, when $S_m = 1$, the two cases yield identical results. Another fact apparent from Figure 5.13 is that the delay-throughput characteristics, when all the messages belong to a single class, is dependent on the priority class. However, this dependence is much less in the modified scheme than that in the IEEE 802.4 scheme. The delay undergone by messages is independent of the class to which they belong for throughputs up to about 0.7 in the modified scheme compared to 0.4 in the IEEE 802.4 scheme. In IEEE 802.4 scheme, the maximum throughput that is achieved when all the messages belong to Class 1 is about 0.22 lower than that when all the messages belong to the highest priority class. In the modified scheme, this difference is only about 0.06.

The modified scheme requires the use of a response window following the transmission of a token along a chord. The additional hardware required to implement this would be minimal as the IEEE 802.4 scheme also makes use of the concept of response windows for ring initialization and maintenance. Since the scheme uses only one response window, it does not have the synchronization problems associated with the virtual token-passing protocols^{7,21}. In the modified scheme, each node is required to have two state variables, CS and R, in addition to those present in the IEEE 802.4 scheme. Initialization and maintenance of these require additional hardware.

The proposed scheme is compatible with the IEEE 802.4 standard in that, a node implementing this scheme can be connected to a network having IEEE 802.4 nodes, by setting the parameter S_m to be unity, and by having provision to send the token without the M flag bit. A node with IEEE 802.4 specifications would also fit in to a network following the modified scheme if one of the bits of the token can be allocated to be the M flag. Such a node will always set the M flag, and, consequently, it will get the token in each of the cycles of the token.

Simulation of the token-skipping scheme without the priorities has shown that the advantages to be gained by token-skipping is high when the number of nodes is high or under asymmetric traffic distribution among nodes. The same can be expected in case of the modified priority scheme too.

5.4 TOKEN-SKIPPING PRIORITY SCHEME FOR TOKEN-PASSING BUSSES

In this section, results of simulations comparing the token-skipping scheme described in Section 3.3 and the IEEE 802.4 priority scheme are presented. All the simulations given in this section have made use of the network parameters given in Table 5.6. Poisson arrival of messages at each node and class is assumed. And traffic distribution among stations is assumed to be symmetric. In the absence of schemes to get the optimum values that can be used for timers in IEEE 802.4 scheme and, for inter-priority assessment time (IPAT), maximum skip distances and number of packets allowed to be

Table 5.6 Network parameters used for comparing the token-skipping priority scheme with the IEEE 802.4 priority scheme.

Number of stations, N	=	50
Number of priority classes, P	=	4
Bandwidth of the bus	=	1 Mbps
Length of the bus	=	1 km
Round trip propagation delay	=	10 μ s
Packet length, including header	=	1000 bits
Size of response window, T_{rw}	=	20 μ s
Token transmission time	=	100 μ s
Number of packets that are allowed to be transmitted per token, $N(I)$, $I=0, \dots, 3$	=	1
Priority assessment frame time	=	100 μ s
Inter-priority assessment time, IPAT	=	2500 μ s
Maximum skip distance of		
class 0, $S_m(0)$	=	1
class 1, $S_m(1)$	=	3
class 2, $S_m(2)$	=	7
class 3, $S_m(3)$	=	11
High-priority token-hold time, HPTHT	=	1000 μ s
Target-token-rotation time for		
class 0, TRT(0)	=	10000 μ s
class 1, TRT(1)	=	15000 μ s
class 2, TRT(2)	=	20000 μ s

transmitted per token in TSPS, these values were selected to make the comparison as meaningful as possible. High-priority token-hold time for IEEE 802.4 scheme is selected to be equal to one packet time because in the token-skipping scheme, only one packet per token is allowed. The values for target-token-rotation times have been selected to preserve a certain degree of hierarchical independence without adversely affecting the performance of messages of lower priority classes. There is some disparity in that according to the parameters selected, IEEE 802.4 scheme may transmit more than one packet per token, while the token-skipping scheme does not.

The delay-throughput characteristics of each class for the case, where all the four classes have same traffic intensity, i.e., $G(0)=G(1)=G(2)=G(3)=G/4$, are given in Figure 5.14. The token-skipping scheme provides a lower delay, compared to IEEE 802.4 scheme, for the highest class of messages. This is to be expected due to the hierarchical independence of TSPS. However, to make way for the higher priority messages, the delay in lower priority classes has increased. For messages of class 2, it provides a lower delay for throughputs up to 0.3. The IEEE 802.4 scheme offers a better quality of service to messages of class 2 in the range of 0.3–0.35. Note that for a class, when the delay is finite, the throughput and the offered load are equal. Similarly, TSPS provides a lower delay for class 1 messages up to a throughput of about 0.1 while the IEEE 802.4 scheme provides a better service in the range 0.1–0.22. The messages of lowest priority have a lower delay in the IEEE 802.4 scheme. By increasing the value

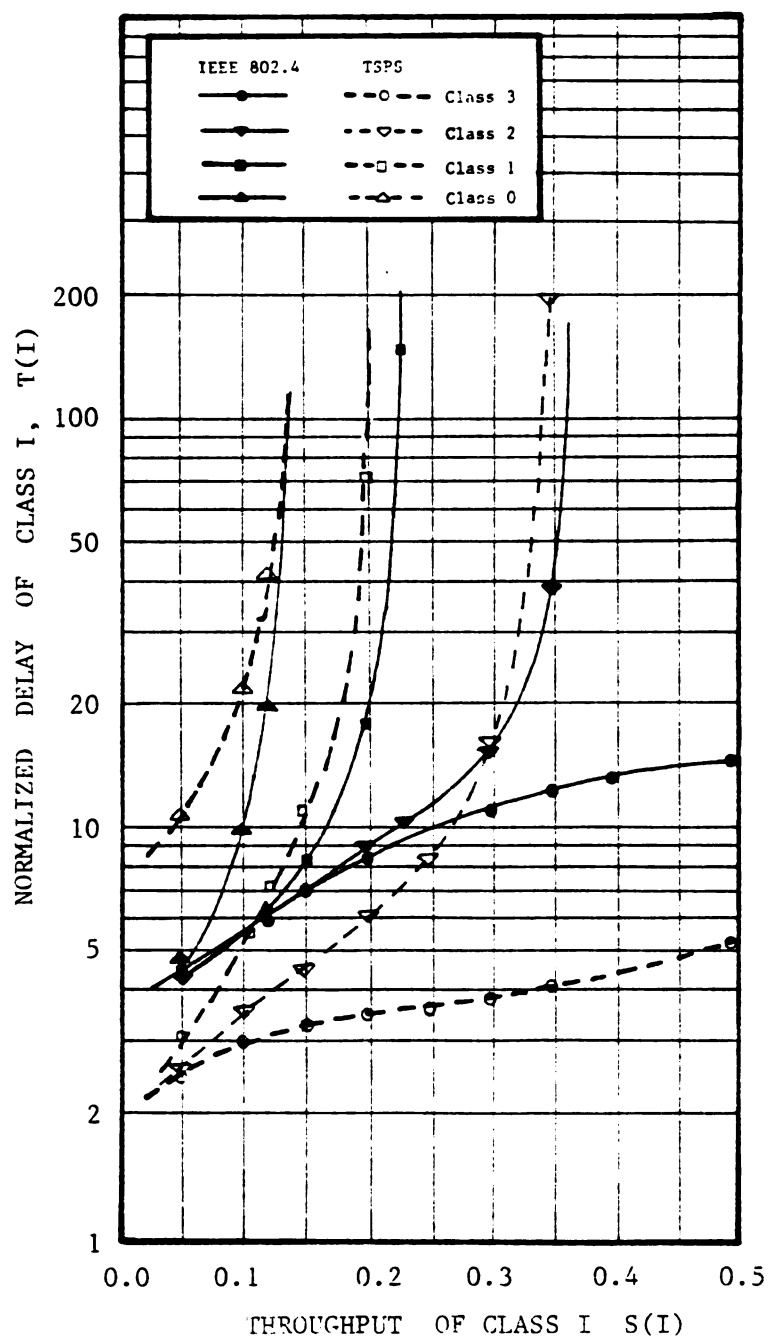


Figure 5.14 Delay-throughput characteristics [$T(I)$ vs. $S(I)$, $I=0, \dots, 3$] for the token-skipping priority scheme when the traffic is equally distributed among the priority classes [i.e., when $G(I) = G/4$, $I=0, \dots, 3$].

of the PAT in the token-skipping scheme, it is possible to reduce the delay of the lower priority classes at the expense of the delay of the higher priority class. The total throughput vs. the offered load is given in Figure 5.15. In this case, the IEEE 802.4 scheme has a throughput which is about 0.06 higher than the token-skipping scheme at high traffic intensities.

In order to study the behavior of the token-skipping priority scheme, a network where only the messages of classes 1 and 3 were present was simulated. The delay-throughput characteristics for the messages of class 3, $T(3)$ vs. $S(3)$, when the offered load in class 1, $G(1)$ is held constant is given in Figure 5.16. This shows that, independent of the load in class 1, for throughputs of less than 88%, the messages of class 3 undergo less delay than the messages of a similar standard token-passing network. Figure 5.17 shows the delay-throughput characteristics for messages of class 1, $T(1)$ vs. $S(1)$, when the load in class 3, $G(3)$, is held constant. Note that the throughput in this case includes only that of class 1 messages, whereas the actual throughput is the sum of the throughput in class 1 and that of class 3. Here the proposed token-skipping scheme shows a significant improvement over the IEEE 802.4 scheme. In both schemes, as the traffic in the higher priority class increases, the delay in the lower priority class increases due to the channel bandwidth being used by messages of higher priority.

The TSPS and IEEE 802.4 priority schemes can now be compared with respect to the Rom and Tobagi's performance criteria²⁸ given in Chapter

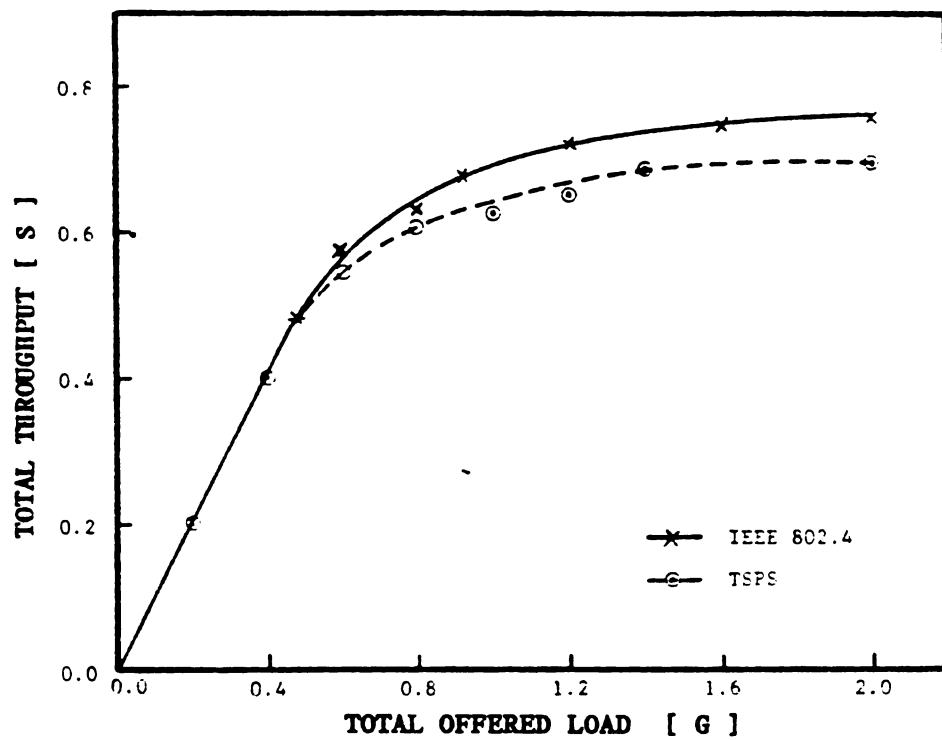


Figure 5.15 Total throughput vs. total offered load [S vs. G] for the token-skipping priority scheme when the traffic is equally distributed among the priority classes.

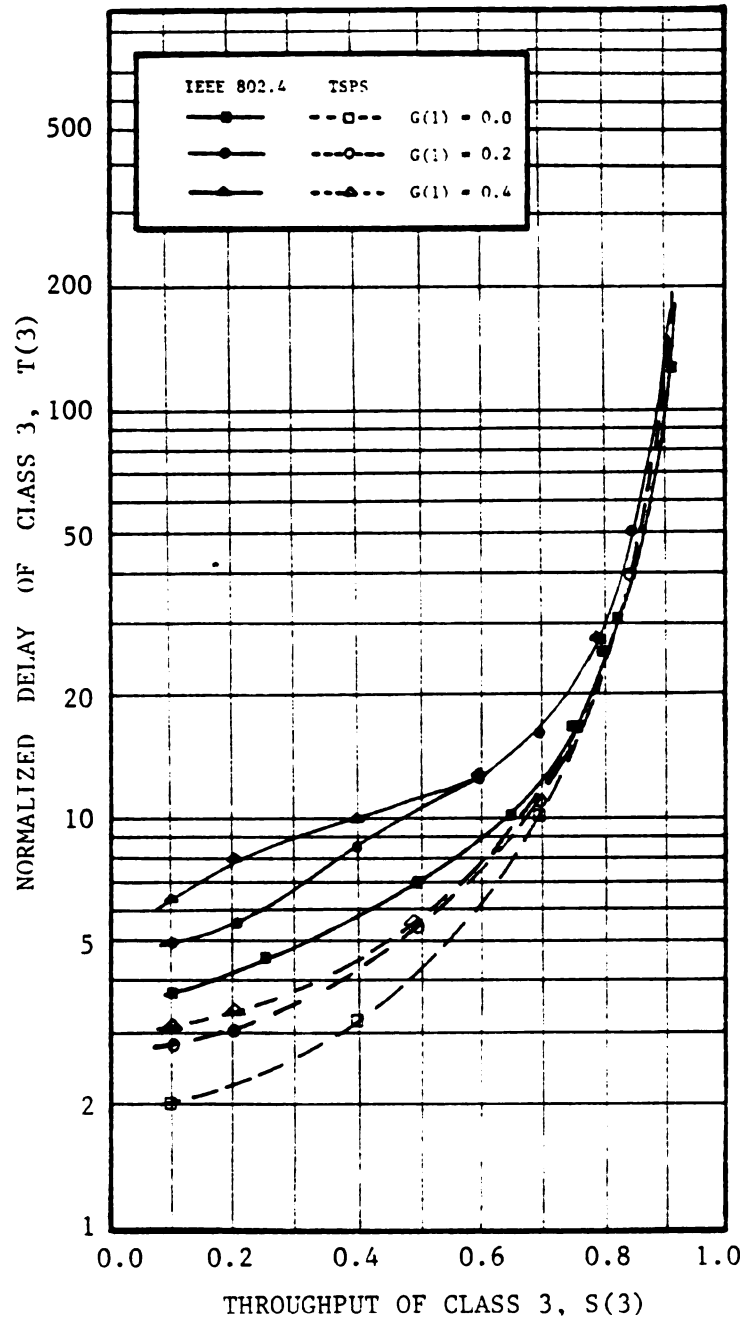


Figure 5.16 Mean delay vs. throughput of packets of priority class 3 [$T(3)$ vs. $S(3)$] for the token-skipping priority scheme when the offered load in class 1 is held constant and there is no traffic in classes 0 and 2 [i.e., $G(1) = \text{const.}$ and $G(0) = G(2) = 0$].

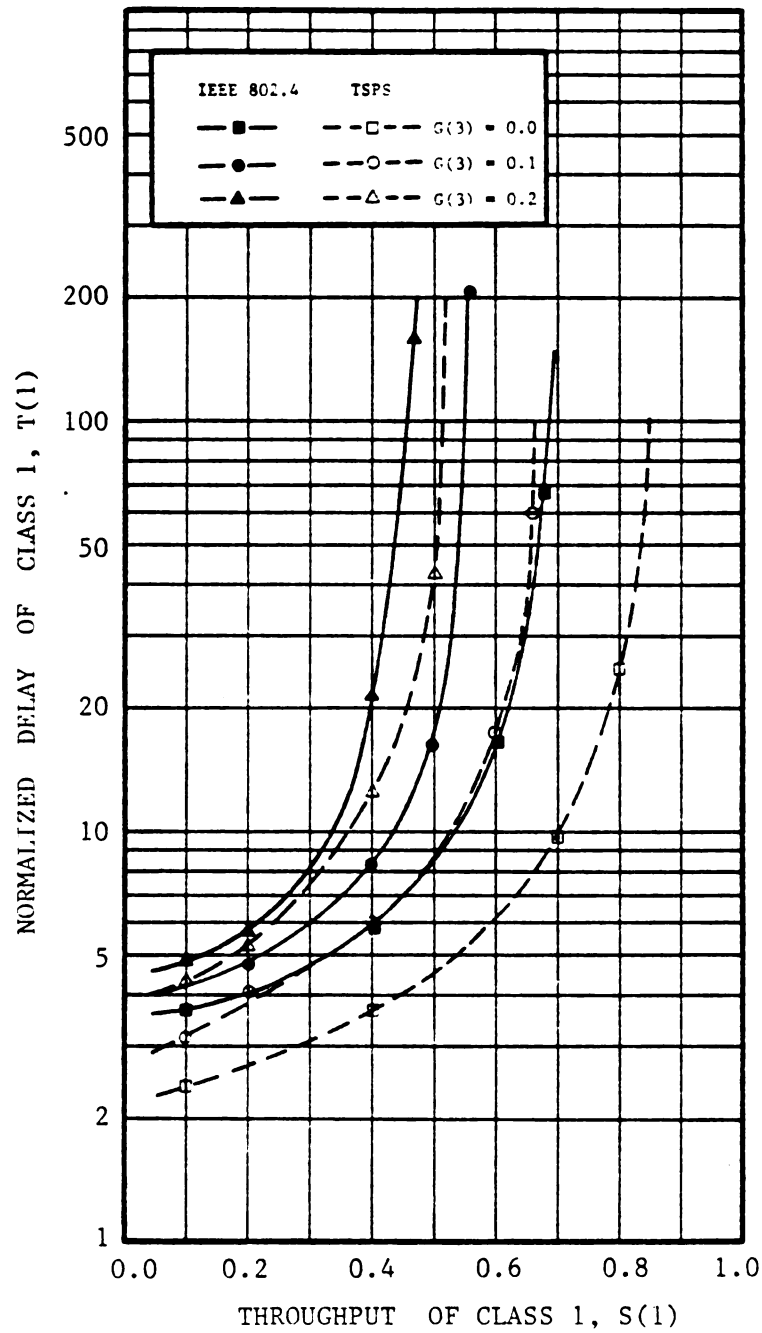


Figure 5.17 Mean delay vs. throughput of packets of priority class 1 [$T(1)$ vs. $S(1)$] for the token-skipping priority scheme when the offered load in class 3 is held constant and there is no traffic in classes 0 and 2 [i.e., $G(3) = \text{const.}$ and $G(0) = G(2) = 0$].

2. The IEEE 802.4 scheme cannot be made to function in a completely hierarchically independent way through parameter selection. A certain degree of hierarchical independence can be achieved by using very low values for target-token-rotation times. However, this will drastically affect the delay-throughput characteristics of messages of lower priorities. In the token-skipping priority scheme presented here, complete hierarchical independence can be achieved by setting IPAT equal to the packet time. This will cause the highest priority present in the network to be determined following each message transmission. In the simulations, IPAT was assumed to be equal to 2.5 packet times. Hence, it is not completely hierarchically independent. For example, from Figure 5.16 it is seen that at a throughput $S(3) = 0.2$, a change in load of class 1 from 0.0 to 0.4 causes a change in delay of class 3 by 1 packet time. For IEEE 802.4, it changes by almost 4 packet times. Even with such a difference in the degree of hierarchical independence, the delay of both class 1 and class 3 messages of the IEEE 802.4 scheme are much worse than that of the TSPS.

The IEEE 802.4 scheme is not fair for the messages belonging to the lower classes of priority. In extreme cases, it is possible for messages of one class at a particular node to get blocked while messages of the same class at another node get transmitted. However, fairness is built into the token-skipping scheme, since, for a given class, the access is always passed to each node along the logical ring.

The priority scheme is robust since no errors in the state information used by the priority scheme can drastically affect the

system performance. The priority scheme itself guards against errors in TCON flags. Also, any error in PPR will only result in the token being transferred to the incorrect layer. This will last only until the PAT expires. Errors in MW and MT flags will result in the token being transferred a little later or little before the token is due to be transferred between layers, affecting only the fairness of the scheme.

The last of Rom and Tobagi's performance criteria for priority schemes is that the overhead has to be kept at a minimum. At first, it appears that the priority assessment frame and circulation of token in each layer separately in the token-skipping method, causes an increase in the overhead, in terms of the bandwidth used. However, in the IEEE 802.4 scheme, it may happen that the token has to pass through a large number of nodes without transmitting the messages of lower priorities present in those nodes. This overhead is reduced in the token-skipping scheme by allowing the token to skip the nodes, which do not have messages of the current priority. It is not possible to say that one scheme has a lower overhead compared to the other. The overhead will depend, among others, on the parameters selected and the distribution of load among nodes and classes. As seen from simulation results for the case where all the four classes have the same traffic intensity, the net throughput of the token-skipping scheme is lower by about 0.06. However, in the case when there are only two classes of messages in the network, the token-skipping scheme handles messages of both classes at a lower delay compared to the IEEE 802.4 scheme. The TSPS will,

however, require additional hardware/software at the nodes to implement the priority scheme.

In this chapter, we compared the performance of the token-skipping scheme, and its priority implementations with the proposed IEEE 802.4 channel-access scheme. The modified priority scheme, which has the IEEE 802.4 scheme as a special case, and is upward compatible with the proposed standard seems to provide a significant improvement in performance. The token-skipping priority scheme, under certain circumstances, sacrifices throughput in order to achieve hierarchical independence and fairness.

CHAPTER 6

SUMMARY AND CONCLUSIONS

This research work was aimed at providing a better understanding of the capabilities and the limitations of a token-bus local area network in handling different classes of traffic and to develop a scheme which would allow it to handle different classes of traffic efficiently. The specific tasks addressed were as follows:

1. identify the limitations of the token-bus network in handling different classes of traffic;
2. devise a priority mechanism, that would allow the token-bus to meet the requirements of different classes of traffic efficiently;
3. investigate the performance of the token-bus architecture for different classes of traffic.

The work completed under each of these tasks is summarized below.

A major limitation of the token-bus architecture for local area networks can be identified as the overhead associated with passing the token through idle nodes. This is especially significant at light loads and when the load distribution among nodes is asymmetric. The proposed IEEE 802.4 standards¹³ for token-bus networks define an optional priority mechanism for token-passing bus networks. In this

scheme, for each class of messages, each node measures the time the token takes to circulate around the logical ring and return to it. But, the transmission of packets of lower priorities are allowed only if this time is less than a predetermined value called the target-token-rotation time (TRT). This may cause the token to pass through a large number of stations without transmitting any messages, thus degrading the performance of the network. If the overhead required to pass the token through those nodes, that would not be able to transmit messages, can be reduced, we should be able to enhance the performance of this network architecture.

The token-skipping channel-access scheme developed in this research is aimed at reducing the overhead involved in passing the token, while retaining the advantages of the token-passing channel-access scheme. It makes use of the broadcast nature of the bus to allow the token to skip most of the nodes which would not make use of the token to transmit packets. In Section 3.2, the token-skipping channel-access scheme was used to modify the proposed IEEE 802.4 priority scheme. This results in a significant improvement in performance under a wide variety of load conditions. It is possible to implement this priority scheme while ensuring upward compatibility with the IEEE 802.4 standard. In Section 3.3, a new priority scheme called the token-skipping scheme was presented. This scheme possesses a high degree of hierarchical independence in performance among different classes of packets, and fairness among packets of the same class, two major attributes expected from a priority scheme. However, it

sacrifices throughput under certain load conditions in order to achieve these attributes.

The third task was to study the performance of the token-passing bus for different classes of traffic. We compared the performance of the proposed schemes, i.e., the token-skipping channel-access scheme, the modified priority scheme and the token-skipping priority scheme with the IEEE 802.4 scheme using simulation studies. Simulation of the token-skipping scheme shows, for example, that a network with 100 nodes, with Poisson arrival of messages and a symmetric traffic distribution among the nodes has a mean delay of less than half that of a similar network using the standard token-passing scheme. Simulations also show that the modified priority scheme uses the bandwidth of the bus more efficiently than does the IEEE 802.4 priority scheme. The modified scheme handles packets of each of the priority classes with a lower mean delay and has a higher throughput under a wide variety of load conditions. The token-skipping priority scheme makes use of the bus to exchange information required to achieve hierarchical independence and fairness. This may cause the throughput to degrade under certain load conditions. The simulation of a network with 50 nodes and four classes of packets, with traffic equally distributed among the nodes and the classes, show the maximum throughput achieved with the token-skipping priority scheme to be lower than that with the IEEE 802.4 priority scheme. However, when the traffic belongs to only one of the four classes, it provides a better service in terms of the throughput and the delay.

The analytical model developed in Chapter 4 can be used to predict the performance of the token-skipping scheme when the following conditions are satisfied: Poisson arrival of messages, symmetric traffic distribution among the stations, constant message size and exhaustive service discipline. Although these assumptions restrict its use, the model still may be used to approximate a more general network. For example, it has generally been observed that the difference due to the service discipline, i.e., exhaustive vs. nonexhaustive, is very small if the traffic is uniformly distributed among the nodes⁴. This model yields accurate delay-throughput characteristics when the ratio of maximum skip distance to the number of nodes is low. This, however, corresponds to the values of the maximum skip distance, which are practical, as a high value for maximum skip distance results in a degradation of performance except under very light offered loads. The analytical model allows the analysis of the network for a variety of parameter settings at a relatively low cost compared to simulations. It also provides a qualitative relationship between different parameters and variables.

The schemes developed in this research work offers a means of enhancing the performance of the token-bus networks while retaining its advantages, such as its deterministic nature and reliability. It requires additional hardware/ software complexity at the individual nodes. Further study is required to compare the cost of this overhead with the improvement in performance. This investigation considered only the noiseless operation of the network. It also assumed a static

set of nodes. Further investigations could be made as to how this scheme will behave when noise is present and when the set of nodes in the network is not static. Moreover, the modified priority scheme and the token-skipping priority scheme should be compared with the IEEE 802.4 priority scheme under more practical offered loads, such as a typical load in a factory floor computer network. This would allow the advantages gained by this new scheme to be compared with the cost of implementing it. If the benefits outweigh the cost, the next step could be the actual implementation of this protocol.

APPENDIX

APPENDIX

FORMAL DESCRIPTION OF THE TOKEN-SKIPPING PROTOCOL AND THE PRIORITY SCHEMES

A formal description of the token-skipping channel-access scheme, the modified priority scheme and the token-skipping priority scheme is given below. The following notation is used:

TS -- Address of this station;
PS -- Address of the previous station on the logical ring;
RS -- Address of the next station on the logical ring;
CS -- Address of a station along a selected logical chord;
CS' -- Address of the station at the other end of the extreme chord,
this has to be initialized by an initialization procedure;
SA -- Source address of the token;
DA -- Destination address of the token;
R -- Ring flag;
M -- Message flag.

I. THE TOKEN-SKIPPING SCHEME

The token-skipping channel-access mechanism can be implemented by four concurrent procedures at each station. The procedures CHORD-SELECTOR and TOKEN-HANDLER are invoked by the transmission of a token by any station. The procedure RING-SELECTOR, invoked by a message transmission by any station, resets the R flag following a message transmission. MASTER, invoked by TOKEN-HANDLER is responsible for message transmission. TOKEN-HANDLER takes care of token transmission and reception. The CHORD-SELECTOR is responsible for

updating the logical chords using the M flag of the token. The four procedures are described below. The value of CS', is determined by an initialization procedure.

Procedure CHORD-SELECTOR

```
{ Invoked by the appearance of a token on the bus}
{ Changes the chords so that they point to the closest station }
{ with messages of the highest priority within a distance  $S_m$  }
Begin
  If ( SA = CS ) then
    If ( M = 0 ) then CS := CS'
  else
    If ( SA lies between TS and CS ) then
      If ( M = 1 ) then CS := SA
End.
```

Procedure MASTER

```
{Invoked by the TOKEN HANDLER }
Begin
  Send up to N packets
  { N is the max. no of packets
    allowed per station per token}
End.
```

Procedure RING-SELECTOR

```
Begin
  {Invoked by the appearance of a message on the bus}
  If Message on bus then R:=0;
End.
```

Procedure TOKEN-HANDLER

{Invoked by the appearance of a token on the bus }
 {Implements explicit and implicit token passing along}
 {ring and explicit token passing along chords.}
 Begin

 If (TS is between SA and DA) then

 Begin

 If (buffer not empty) then send burst
 else listen;

 If (resp. window idle) then terminate;

 R:= true;

 If (PS <> SA) then terminate;

 If (buffer not empty) then

 Begin

 MASTER;

 R:=false; SA:=TS; DA:= CS

 End

 else

 Begin

 SA:=TS; DA:=RS

 End;

 If (buffer not empty) then M:=1

 else

 M:=0;

 Send Token

 End

 else

 Begin

 If (TS<>DA) then terminate;

 If (PS<>SA) then

 If (Resp. window busy) then terminate;

 If (buffer not empty) then

 Begin

 R:=false;

 MASTER

 End;

 SA:=TS;

 If R then DA:=NS

 else

 DA:=CS;

 If (buffer not empty) then M:=1

 else

 M:=0;

 Send token

 End

End.

II. THE MODIFIED PRIORITY SCHEME

The modified prioritized token-passing access scheme can be explained by the following five procedures implemented by each of the nodes. Procedures **TOKEN-HANDLER** and **CHORD-SELECTOR** are invoked by a token transmission by any node. Procedure **MASTER** is called by the **TOKEN-HANDLER** when a station receives the token. **RING-SELECTOR** is invoked by a message transmission by any station. The procedure **CLOCK** updates the timers used in the priority scheme. The following notation is used:

P -- The number of priority classes;
HPHT -- High-priority token-hold time;
TOL(i), $i=0, \dots, P-2$ -- Tolerance for class i ;
TRT(i), $i=0, \dots, P-2$ -- Target token-rotation time for class i ;
TRTIM(i), $i=0, \dots, P-2$ -- Token-rotation timer for class i .

Procedure MASTER

{ Invoked by the **TOKEN-HANDLER** upon receiving the token}

Begin

{ Service queues starting with the highest priority}
 { and sends the token }

THT := **HPHT**;

For **J** := **P**-1 to 0 do

Begin

While (**THT** > 0 and Packets of class **J** present) do

Send packets of class **J**;

If (**J** > 0) then **THT** := **TRTIM(J-1)**;

TRTIM(J) := **TRT(J)**

End;

{Send the token to the next node}

SA := **TS**;

If (Packets of Class **P**-1 present) then **M** := 1

else

M := 0;

If **R** then **DA** := **RS**

else

DA := **CS**;

Transmit Token;

R := false

End.

Procedure TOKEN-HANDLER

{Invoked by the appearance of a token on the bus }

{Implements explicit and implicit token passing along}

{ring and explicit token passing along chords.}

Begin **If** (TS is bypassed by the token) **then** **Begin**

Active := false;

If (Packets of Class P-1 present) **then** Active := true **else** **For** I := 0 to P-2 **do** **If** (Packets of Class I present and TRTIM(I) > TOL(I)) **then**

Active := true;

If (Active) **then** Send a burst **else**

Listen;

If (Response Window not busy) **then** **For** I := 0 to P-2 **do**

TRTIM(I) := TRT(I)

else

R := true;

If (PS = SA) **then** **Begin**

Accept Token;

MASTER

End **End** **else** **If** (TS = DA) **then** **If** (PS = SA) **then** **Begin**

Accept the token;

MASTER

End **else** **Begin**

Listen;

If (Response window not busy) **then** **Begin**

Accept the token;

MASTER

End **End****End.**

Procedure RING-SELECT

{Invoked by the appearance of a data packet on the bus}

Begin

R := false

End**Procedure CLOCK**

{Invoked by a clock every octet time}

Begin

If (THT > 0) then THT := THT - 1;

For I := 0 to P-2 do

If (TRTIM(I) > 0) Then TRTIM(I) := TRTIM(I) - 1

End.**Procedure CHORD-SELECT**

{ Invoked by the appearance of a token on the bus}

{ Changes the chords so that they point to the closest station }

{ with messages of the highest priority within a distance S_m }**Begin**

If (SA = CS) then

If (M = 0) then CS := CS'

else

If (SA lies between TS and CS) then

If (M = 1) then CS := SA

End.**III. TOKEN-SKIPPING PRIORITY SCHEME**

This scheme can be implemented using six procedures. The procedures TOKEN-HANDLER and CHORD-SELECTOR are similar to those described for the token-skipping scheme, except now they operate similarly on P different layers. P is the number of priority classes of messages. These two procedures, together with the procedure MASTER facilitates the token passing within each layer. TOKEN-HANDLER and CHORD-SELECTOR are invoked by the appearance of a token on the bus and the procedure RING-SELECTOR by the appearance of a message on bus. The

procedure MASTER is called by the TOKEN-HANDLER once a node receives the token and becomes the bus master. It takes care of message transmission as well as the transfer of the token between layers. Two other procedures, PAF-HANDLER triggered by the appearance of a PA frame on the bus and FLAG-UPDATE, invoked by a successful token pass along the bus, update timers and the flags. The following notation is used:

M(i), i=0,...P-1 -- Message flag for class i;
 MW(i), i=1,...P-1 -- Messages waiting flag for class i;
 MT(i), i=1,...P-1 -- Messages transmitted flag;
 N(i) -- Maximum number of packets of class i that are allowed to be transmitted per station per token;
 P -- Number of priority classes of messages;
 PAF -- Priority assessment frame;
 PAT -- Priority assessment timer;
 TP -- Priority of the token;
 TCON(i), i=1,...P-1 --Token convert flag indicates the station that converted the token to class i;
 PPR(i), i=1,...P-1 -- The priority of the token before it was converted to priority i.

Procedure FLAG-UPDATE

{ Invoked by a successful token pass}
 Begin
 If (TS=SA or TS is bypassed by the token) then
 MW(TP):=false;
 MT(TP):=false;
 else
 For I:=1 to P-1 do
 If M(I) then MW(I):=true
 End.

Procedure RING-SELECTOR

{Invoked by the appearance of a message on the bus}
 Begin
 R(TP):= false;
 MT(TP):= true
 End.

Procedure MASTER

{Invoked by the TOKEN-HANDLER upon receiving a token}

Begin

Send packets of Class TP until

[Packets of class TP are exhausted
 OR N(TP) packets are transmitted
 OR (TP<>P-1 AND PAT has expired)];

If [TP<>P-1 AND PAT has expired] then

Begin

{Find the highest priority of messages
 present, H by transmitting a PAF}

Send PAF;

If H>TP then

{Transfer the token to the new layer}

TCON(H):=true;

PPR(H):=TP;

TP:=H

End**else**

If [TCON(TP) AND ((NOT MW(TP)) OR (NOT MT(TP)))
 OR (NOT MT(TP) AND NOT MW(TP))] then

{If the token is back at the station which converted
 it to the current class and (no messages are waiting
 or no messages were transmitted)
 or (no messages are waiting and
 no messages were transmitted) }

Begin

If (TCON(TP) OR (NOT MT(TP) AND NOT MW(TP))) then

Begin

{Transfer the token to previous layer}

TCON(TP) := false;

X:= TP;

TP:= PPR(X);

PPR(X):=1;

End;

{Find the highest priority of messages
 present, H by transmitting a PAF}

Send PAF;

If H>TP then

{Transfer the token to the new layer}

TCON(H):=true;

PPR(H):=TP;

TP:=H

End**End.**

```

Procedure PAF-HANDLER
{Invoked by the appearance of a PAF on the bus}
Begin
  {Update PAT}
  PAT:=IPAT;
  For I:= P-1 down to TP do
    If messages of class I present then
      Begin
        Send burst;
        terminate
      End
    else
      Begin
        Listen
        If response window busy then
          terminate;
        End
      End
    End
  End

```

BIBLIOGRAPHY

BIBLIOGRAPHY

1. Allen A.O., "Queueing Models of Computer Systems," Computer, Apr. 1980, pp. 13-24.
2. Arthurs E. and Stuck B.W., "A Theoretical Performance Analysis of Polling and Carrier Sense Collision Detection Communication Systems," Local Computer Networks, edited by P.C. Ravisio et al., North Holland, 1982, pp. 415-437.
3. Burr W.E., "An Overview of the Proposed American National Standards for Local Distributed Data Interfaces," Communications of the ACM, Vol. 6, No. 8, Aug. 1983, pp. 554-561.
4. Bux W., "Local Area Networks: A Performance Comparison," Local Networks for Computer Communications, edited by A. West and P. Janson, North Holland, 1981, pp. 157-180.
5. Cherukuri R., Li L. and Louis L., "Evaluation of Token-Passing Schemes in Local Area Networks," Proc. Computer Networking Symposium, Dec. 1982, pp. 57-68.
6. Chien J.Y., "Performance Analysis of the 802.4 Token Bus Media Access Control Protocol, Presented at the Factory Floor Communications Workshop, GM Tech. Center, MI, Sept. 1984.
7. Chlamtac I. and Franta W.R., "BRAM: The Broadcast Recognizing Access Method," IEEE Trans. Communications, Vol. Com. 27, No. 8, Aug. 1979, pp. 1183-1189.
8. Clark, D.D., Pogran, K.T. and Reed, D.P., "An Introduction to Local Area Networks," Proc. of the IEEE, Vol. 66, No. 11, Nov. 1978, pp. 1497-1517.
9. Dahod A.M., "Local Networks Respond to Changing System Needs," Computer Design, Vol. 23, No. 6, June 1, 1984, pp. 61-68.
10. DEC, Intel and Xerox Corp., The Ethernet: A Local Area Network Data Link and Physical Layer Specifications, Version 1.0, Sep. 1980.

11. Draft IEEE Standard 802.2: Logic Link Control, Draft D, IEEE Computer Society, Nov. 1982.
12. Draft IEEE Standard 802.3: CSM/CD Access Method and Physical Layer Specifications, Draft D, IEEE Computer Society, Dec. 1982.
13. Draft IEEE Standard 802.4: Token-Passing Bus Access Method and Physical Layer Specifications, Draft E, IEEE Computer Society, July 1983.
14. Fisher P.D., "Local Area Networks: A Basic Definition," Industrial Technology Institute, File No. ITI/ LANG/ 83/01, Apr. 1983.
15. Franta W.R. and Chlamtac I., Local Networks, Lexington Books, 1981.
16. Ikeman H., Lee E.S. and Boulton P.I.P., "High-speed Network uses Fiber Optics," Electronics Week, Oct. 22 1984, pp. 95-100.
17. Kleinrock L., Queueing Systems, Vol. 1: Theory, John Wiley and Sons, 1975.
18. Kleinrock L., Queueing Systems, Vol. 2: Computer Applications, John Wiley and Sons, 1976.
19. Konheim A.G. and Meister B., "Waiting Lines and Times in a System with Polling," Journal of the Assoc. Computing Machinery, Vol. 21, No. 3, July 1974, pp. 470-490.
20. Kuehn P.J., "Multiqueue Systems with Nonexhaustive Cyclic Service," The Bell Systems Technical Journal, Vol. 58, No. 3, Mar. 1979, pp. 671-698.
21. Li L. and Hughes H.D., "Definition and Analysis of a New Shortest Delay Protocol," Proc. 6th Conference on Local Computer Networks, Oct. 1981, pp. 21-29.
22. Liu M.T., Hilal W. and Groomes B.H., "Performance Evaluation of Channel-Access Protocols for Local Computer Networks," Proc. COMPCON, Fall 1982, pp. 417-426.
23. Liu T.T., Li L. and Franta R., "A Decentralized Conflict-free Protocol, GBRAM for Large Scale Local Networks," Proc. Computer Networking Symposium, 1981, pp. 39-54.
24. Liu T.T., Li L. and Franta R., "The Analysis of a Conflict-free Protocol Based on Node Clusters," Proc. 6th Conference on Local Computer Networks, 1981, pp. 61-72.

25. Pack C.D. and Whitaker B.A., "Multipoint Private Line Access Delay under Several Interstation Disciplines," IEEE Trans. Communications, Vol. Com-24, No. 3, March 1976, pp. 339-348.
26. Perlman R., "Incorporation of Service Classes into a Network Architecture," Proc. 7th Data Communications Symposium, Oct. 1981, pp. 204-210.
27. Reiser M., "Performance Evaluation of Data Communication Systems," Proc. of the IEEE, Vol. 70, No. 2, Feb. 1982, pp. 171-196.
28. Rom R. and Tobagi F.A., "Message Based Priority Functions in Local Multiaccess Communication Systems," Computer Networks, North Holland, 1981, pp. 273-286.
29. Sauer C.H. and Chandy K.M., Computer Systems Performance Modelling, Prentice Hall, 1981.
30. Silvester J.A. and Lee T., "Performance Modelling of Multi-Access Computer Communication Networks," Proc. COMPCON, Fall 1982, pp. 377-382.
31. Stack T.R. and Dillencourt K.A., "Protocols for Local Area Networks," Proc. IEEE Trends and Applications : Computer Network Protocols, May 1980, pp. 83-93.
32. Stuck B.W., "Calculating the Maximum Mean Data Rate in Local Area Networks," Computer, May 1983, pp. 72-76.
33. Tannenbaum A.S., Computer Networks, Prentice Hall, 1981.
34. Tannenbaum A.S., "Network Protocols," ACM Computing Surveys, Vol. 13, No. 4, Dec. 1981, pp. 453-489.
35. Tobagi F.A., Gerla M., Peebles R.W. and Manning E.G., "Modelling and Measurement Techniques in Packet Communication Networks," Proc. of the IEEE, Vol. 66, No. 11, Nov. 1978, pp. 1423-1447.
36. Tobagi F.A. and Kleinrock L., "Packet Switching in Radio Channels: Part III - Polling and (Dynamic) Split Channel Reservation Multiple Access," IEEE Trans. Communications, Vol. Com-24, No. 8, Aug. 1976, pp. 832-845.
37. Zimmermann H. and Pouzin L., "The Standard Network Architecture Developed by ISO," Data Communication and Computer Networks, North-Holland, 1981.

MICHIGAN STATE UNIV. LIBRARIES



31293107741005