Analyses of
Asymmetric Prioritized Token Ring
and
Interconnection of Token Ring Networks

Bruno Gréla-M'Poko

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy at
Concordia University
Montréal, Québec, Canada

January, 1989

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

ISBN 0-315-49115-9

ABSTRACT

ANALYSES OF ASYMMETRIC PRIORITIZED TOKEN RING AND INTERCONNECTION OF TOKEN RING NETWORKS

Bruno Gréla-M'Poko, Ph. D.,
Concordia University, 1989.

This thesis is concerned with two important issues in Local Area Networks which use Token Passing as accessing technique to share a common transmission facility.

The first issue considers an asymmetric prioritized token-ring operation with nonpreemptive priority queueing and a limited service discipline at each station. The number of message priority levels and the walktime vary from one station to another. Moreover, the assumed Poisson message arrival process and the message transmission time vary not only from one station to another but also from one class to another. The analysis focuses on the observation that upon the arrival of a tagged message at a station, the server is either performing walktime, or serving a message at the station where the tagged message arrived or else serving a message at another station.

Based on these mutually exclusive cases, three special time intervals required to carry out the analysis are defined. The analysis of the mean message delay is based on characteristics of the cycle time as well as on these special intervals and in all cases the balance of flow is of particular interest.

The performance, as measured by the mean delay for any message class at any station, is derived and the numerical results validated through simulation. Interestingly, we found that in asymmetric cases the performance at a station is affected by its polling order and lower classes appear more sensitive to this position-dependent effect. The efficiency of the model is compared with the performance of other prioritized schemes available in the literature.

The second issue we consider in this thesis is the performance of a hierarchical ring network in which a number of outer-rings are interconnected through an inner-ring backbone network. The connection between outer and inner rings is provided through gateways. The Poisson arrival of messages to the outer-ring stations is assumed, with two classes of messages. Class 1 is inter ring message while class 2 is intra ring message. With respect to these traffic classes, two service disciplines are studied. In the first case, both classes of messages are treated equally while in the second case, class 1 messages are given nonpreemptive priority over class 2 messages.

By means of a certain independence assumption and under a limited service discipline, outer-ring stations are modeled as an M/G/1 queue with and without priority. The gateways are modeled as G/G/1 queues. In both types of queues, a message initiating a busy period receives special service.

Mean delay of intra and internetwork messages are derived. In order to justify the assumptions required to carry out the analysis, the system was simulated. The numerical results show excellent agreement with the results of the simulation.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisors Dr J.F. Hayes and Dr M. Mehmet-Ali for introducing me to the exciting world of research. Their knowledge and enthusiasm guided me through my research. They taught me patiently a rigorous research method and showed me how knowledge is perpetuated and enhanced in the university environment. I consider myself very fortunate to work with them. Dr Mustafa's suggestions and encouragement and particularly his continous availability to discuss scientific issues was valuable in maturing new concepts and ideas. I owe my knowledge of Local Area Networks to him.
I could fill up a few pages with names of many wonderful people who provided the encouragement and environment. It is with great pleasure that I acknowledge support and motivation provided by "la *famille Marinet de Bourg-La-Reine*" in France. Lynne and Maureen will always be with me in form of fond memories of joyous moments in the department. It is with pride that I acknowledge the support and motivation of my parents Suzanne and Antoine, my uncles Philippe, Fulbert, Joseph ... and my brothers and sisters - back home in Central Africa.

To my children Eric, Anais and Maureen I offer my deepest regret for all this period, I know, I would have spent with them. Last but not the least - To Philomène, my wife, I owe so much for her patience and understanding. Merci beaucoup.

To my children

Eric, Anais and Maureen

Table of Contents

CHAPTER I

INTRODUCTION

1.1. MOTIVATIONS

1.1.1 PRIORITY MECHANISM IN TOKEN RING

Local Area Networks (LANs) connect data sources in a limited geographical area to one another and to a central facility. Such networks have been widely used and constitute an essential component of many information processing and information gathering systems. A number of accessing techniques may be used to share the common transmission facility among the sources connected to the Local Area Networks. Among these accessing techniques is Token Passing which may be analyzed by means of a Polling model. From the viewpoint of Queueing Theory, Polling is a single server, multi-queue, cyclic service system whose congestion analysis has been the subject of many papers. The advantage of the Polling approach is that it provides graceful degradation of performance with increasing load. The performance of Token-Passing ring networks without priority structure appeared in many papers [1-11,16] which adopt models of a multi-queue, cyclic service queueing system with (server's) walktimes. The server's walktime corresponds to the signal propagation delay over a ring and the bit latency at each station. Such models were originally used in the performance analysis of polling systems in telecommunication networks (see [5] for a tutorial survey and bibliography).

However, for systems with priorities, only few treatments have been published [12-15]. The reality is that in communication networks there often exist different kinds of messages, such as real-time voice, short interactive data, long file transfer, acknowledgement and network control and it is desirable to have a priority structure among these message classes, since the different classes usually have different delay requirements. The first part of this thesis focuses on an asymmetric prioritized token ring operation with nonpreemptive priority queueing discipline at each station. Our interest is in a general and completely asymmetric topology which would allow for growth and flexibility. An asymmetric system is particularly of interest in modelling interconnected networks. In fact, even when it is assumed that the stations connected on each network are statistically identical, the bridge or gateway connecting two networks definitely has different traffic patterns from the stations on each of the two networks it is connecting [16]. Thus. in order to compute the end-to-end delay in interconnected networks, it is necessary to understand asymmetric systems. For our model, the number of priority levels and the time to transfer the control vary from one station to another. Moreover, the message arrival process and the message length distribution vary not only from one station to another but also from one class to another. Under a limited service discipline, the performance as measured by the expected delay for any arbitrary class at any station is derived. The limited service system is fair (as compared to exhaustive and gated systems) in the sense that it prevents heavily loaded stations from hogging the server even though no attempt seems to have appeared to quantify this claim [5]. The numerical results are reasonably accurate; they are within 10% of simulated values for all classes except for an overestimation for the low-priority delay at heavy utilization.

1.1.2. INTERCONNECTION OF TOKEN RING NETWORKS

Because of the wide utility of Local Area Networks and their limited area of service, it is becoming increasingly necessary to interconnect them to one another in order to offer better service. In fact, as a local area network installation grows, it may exceed the allowable values of the design parameters of an individual LAN. Restrictions such as physical extent, number of stations, performance, and media may be alleviated by the interconnection of multiple LANs. The interconnection of local area networks may be desirable for a variety of reasons, including improved performance (since each local area network contributes its own bandwidth), signal quality (compared to attaching all stations in the same establishment to the same local area network), and availability (since the breakdown of one LAN does not cause all stations on all LANs to lose service), as well as sometimes simple connectivity when individual local area networks already exist. The traditional method of providing this interconnection borrows techniques from wide area network technology, requiring the use of a common internetwork protocol or protocol translation gateways. We should note that, although LANs have been standardized in the United States [12], Europe [17], and internationally [18], the two important medium access control (MAC) standards apply to stations on a single CSMA/CD or token ring. On the other hand, there is no standard for the interconnection of LANs in the same facility or establishment into what we call a *system of interconnected LANs.*

A logical interconnection configuration is a high speed backbone network which offers a common path for all traffic between LANs. Indeed, this is the approach which has been followed for many years in the telephone network. The alternative of direct connection between LANs would be viable only for a small number of LANs. Again, our interest is in a general topology which would allow for growth and flexibility. In the second part of this thesis, a topology together with accessing technique which is a logical solution to the interconnection problem is studied. As work in the area progresses, this should be compared with other solutions.

The specific problem under study is the interconnection of LANs with ring topology. In order to simplify the development and operation of such systems, it is reasonable to use the same ring topology for the backbone network. It is of interest, if only from a historical point of view, that the first networks which were dedicated to data traffic were in the form of a hierarchy of ring networks [19,20]. The resulting topology is shown and studied in chapter V. We have a set of outer-rings, which are the LANs, connected together by an inner-ring, the backbone network. The point of contact between the LANs and the backbone network through which the internetwork traffic flows is the gateway. Since the primary function of these networks is to handle traffic which has irregular, bursty flow characteristics, buffering must be provided at these gateways. The gateway has two buffers together with the attendant circuitry. A message destinated for a distant local ring passes through a gateway to the inner-ring. The inner-ring conveys the message to the gateway of the distant ring.

From an operational point of view, each outer-ring consists of a number of station queues and a half of a gateway referred to as outer-ring gateway queue. Inner-ring queues are the halves of the gateways carrying traffic from the inner-ring to the outer. They are referred to as inner-ring gateway queues.

A fundamental assumption of the study is that Token Passing is used. Once the ring topology has been chosen, Token Passing as the accessing technique follows logically since it is an almost universal standard for rings. In the Token Passing technique, a logical token circulates in an outer-ring. The capture of the token by a station allows it to place a message on the ring. The message is accompanied by the address of the destination station. Each station reads the addresses of messages passing through and removes those with its address. In the same fashion messages from local stations are transmitted to the gateways. After a single message is transmitted (limited-to-one service), control of the ring is relinquished by passing the token on to the next station in a logical sequence.

We also assume that the gateway queues are treated in the same fashion as all the other queues in the outer-ring networks. The same protocol is followed in the inner (backbone) ring. A token circulates among the inner-ring gateway queues; its capture by an inner-ring gateway queue allows it to transmit a message to an outer-ring gateway queue.

In this thesis, the outer-ring stations are modeled as M/G/1 queues with and without prior'ty. The gateways are modeled as G/G/1 queues. In both types of queues, a message initiating a busy period receives special servire. Mean message delay for both intra and internetwork traffic for two modes of operation in the outer-ring stations is determined.

1.2 OUTLINE OF THE THESIS

This thesis presents the analyses of a generally asymmetric prioritized token-ring operation with nonpreemptive priority queueing discipline at each station and the interconnection of token-ring networks through gateways. Chapter III is concerned with the prioritized token-ring. The results from that chapter are compared with those obtained under other prioritized schemes in chapter IV. Chapter V is concerned with the interconnection of token-rings with and without priority mechanism. The performance as measured by the average delay in the prioritized token-ring and the system of interconnected token-rings is derived. The main contents of each chapter are given below.

Chapter II : Preliminaries

This chapter presents an overview of local area networks. Local area network topologies and channel accessing techniques are surveyed. A classification of polling models is given and its performance characteristics compared with the contention accessing tehnique CSMA/CD.

Chapter III : Asymmetric Prioritized Token-Ring

This chapter focuses on the analysis of a general asymmetric token-ring. The model is described, and by a mean value analysis, the mean delay of any message class at any station is derived. The stability conditions of the system are presented and the performance of the system illustrated by means of several relevant examples validated by simulation.

Chapter IV : Priority Access Schemes in LANs: A Comparison

In this chapter, we compare the results obtained from our model developed in chapter III with some results reported in the literature. The description and results of Gianini and Manfield's two-priority polling model are given and the various schemes of the prioritized versions of CSMA/CD are presented.

Chapter V : Interconnection of Token-rings

This chapter is concerned with the performance of a system of interconnected Token-rings through gateways. Thc resulting topology is described. The Poisson arrival of messages to the outer-ring stations is assumed, with two classes of messages. Class 1 is inter ring messages while class 2 messages stay on the same ring.

With respect to these traffic classes, two service disciplines are studied. In the first case, both classes of messages are treated equally while in the second case, class 1 messages are given nonpreemptive priority over class 2 messages since their path through the network is longer.

By means of a certain independence assumption and under a limited service discipline, outer-ring stations are modeled as M/G/1 queues with and without priority. The gateways are modeled as G/G/1 queues. In both types of queues, a message initiating a busy period receives special service. To model the outer-ring station queues (in the case of priority structure) and the gateway queues (in both cases), existing M/G/1† with priority and G/G/1† results are not applicable and generalization of previous results is necessary. Mean delay of local and internetwork messages are derived. When there is no priority structure, the probability distribution of the delay components is found. In order to justify the assumption required to carry the analysis through, the analytical results are validated through simulation.

Chapter VI : Conclusions and Future Work

This thesis is neither a start nor a culmination of the exciting field of analysis of local area networks, and in particular those which use token passing as channel accessing technique and which may be analyzed using polling mathematical model. It is basically an attempt to provide solutions to two increasingly important issues in token passing. Because of the complexity of these issues, originating from the complexity of the cyclic-service model, the intriguing application of probabilistic thinking with elegant mathematics may not lead to exact solutions and approximations become usefull.

Therefore the problems remain and I feel that we will need a breakthrough to solve such problems exactly. However, for such complex and general issues, our approximate approaches are very useful and may be used as reference results.

The objective of this chapter is to summarize the contributions made in this thesis. Several issues that arise from various considerations and which may lead to possible future investigations are indicated.

† In connection with queues, a convenient notation has been developed. In its simplest form it is written A/R/S, where A designates the arrival process, R the service required by an arriving customer, and S the number of cervers. For example, M/G/1 indicates a Poisson arrival of customers, a generally distributed service discipline and a single server. The M may be taken to stand for memoryless (or Markovian) in both the arrival process and the service time. Since we assume that there is no restriction on the number in a queue, a reference to the waiting room is not useful.

CHAPTER II

PRELIMINARIES

Local Area Networks (LANs) are most often described as privately owned networks that offer reliable high-speed communications channels optimized for connecting information processing equipments in a limited geographic area, namely, an office, building, complex of buildings, or campus. The *control strategy* of a local area network is decided early in the planning stages when the number and kind of participating stations are specified, as well as the nature of their interaction. The configuration of stations and links, the *topology,* is the major means of implementing this choice of control strategy. The network topology is created by the geometric arrangement of the links and stations that make up the system. A link (also called a line, channel or circuit) is the communications path between two stations. Most local area networks are based on simple structured topologies, like the ring, bus, or star. These topologies permit more effective and uniform implementation of network control strategies.

This chapter presents an overview of local area networks. Some of the basic characteristics of the local area network topologies and channel accessing strategies are outlined. Since this thesis deals with LANs which use token passing as accessing technique, it is, therefore, useful to review the relevant results on its mathematical queueing model and to compare its performance characteristics with other important channel access strategies.

The chapter is divided into four main sections. In section 2.1, the basic characteristics of LAN topologies (star, ring, bus) are outlined using a comparative approach for a better understanding. Section 2.2 is concerned with the channel accessing techniques. Non-contention (Polling) and random access (Contention) strategies are presented. Polling systems are classified and reviewed in section

2.3. Some results for commonly used service disciplines are given and the pseudo-conservation law unifying these results is presented. Finally, the efficiency of token ring is compared with the performance of its major rival CSMA/CD. This is of particular interest since the prioritized versions of these access techniques will be compared in chapter IV.

2.1 LOCAL AREA NETWORK TOPOPOLOGIES

Topoplogy determines the logical interconnection of the stations to each other. In the following, three common topologies for LANs, namely star, ring and bus are presented (see Fig.2.1(a)-(c)).

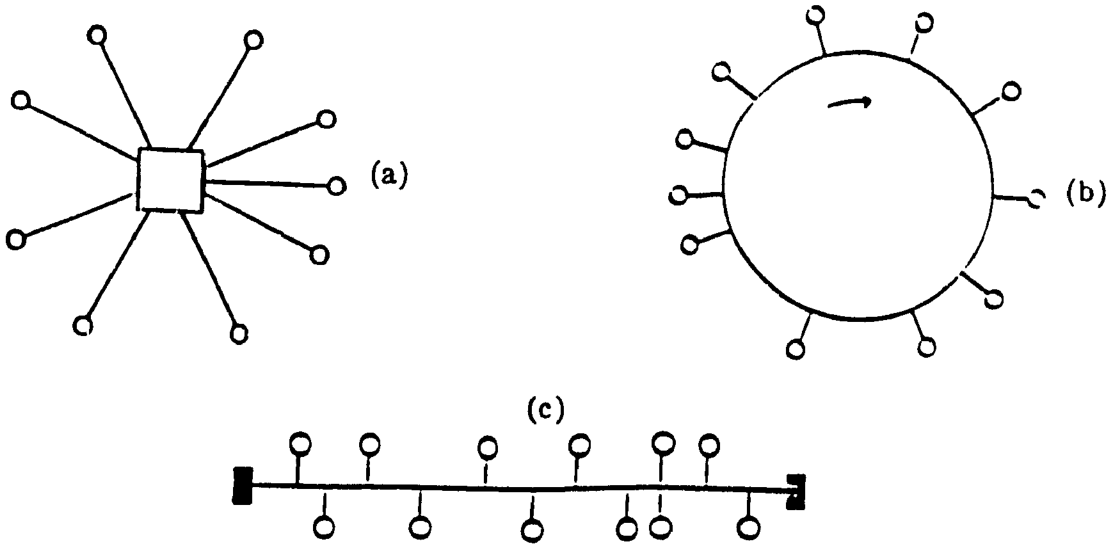


Fig.2.1 (a)-(c) Star, Ring, Bus Configuration.

2.1.1 Star Topology

The distinguishing feature of star or radial topologies is that all stations are joined at a single point as illustrated by Fig.2.1(a). The way star networks function depends on the way in which control is exercised. This usually happens in one of three ways. First, star configurations are frequently used for networks in which control of the network is located in the central node or switch. When this is the case, all routing of network message traffic, from the central node to outlying stations, between outlying stations, and from all stations to remote points, is performed by the central node. This has the effect of relieving outlying stations of control functions. Point-to point lines connect the central node and the outlying stations, eliminating the need for complex link and control requirements of other topologies, and allowing simple and usually low-cost connections to the network through the central node. A point-to-point line must be added for each station connected.

Second, a star network could be constructed so that the control of communications would be exercised by one outlying station, or third, distributed generally to all outlying stations. The central node would serve as a simple switch to establish circuits between outlying stations.

In all star networks the central station is a single point of network failure. If it goes down, so does the entire network. Thus, reliability and the need for redundancy measures are important issues in star networks. In the case of central control, the size and capacity of the network are direct functions of the power of the central station. It is important to note that star also lacks modularity and is not easy to expand.

2.1.2 Ring Topology

The distinguishing feature of ring topologies (Fig.2.1(b)) is that all the stations, which are connected by point-to-point links (links connecting two and only two stations without passing through an intermediate station), are arranged to form an unbroken circular configuration. Transmitted messages travel from station-to-station around the ring. Each station must be able to recognize its own address in order to accept the messages. In addition, each station serves as an active repeater, retransmitting messages addressed to other stations. The need to retransmit each message can make ring stations more complex than the passive stations on a bus network. However, ring stations can be less complex than the routing stations in an unconstrained topology; since message routes are determined by the topology, messages automatically travel to the next station on the ring.

When ring configurations are used to distribute control in local networks, access and allocation methods must be used to avoid conflicting demands for the shared channel. One way of doing this is by circulating a bit pattern, called a token, around the ring. A station gains exclusive access to the channel when it captures the token. It passes the right to access the channel (i.e., the token) on to other stations when it is finished transmitting. Most distributed rings operate on some variant of this idea.

Rings provide a common network channel wherein all stations are fully connected logically. When control is distributed, each station can communicate directly with all other stations under its own initiative. Ring networks with centralized control are often referred to as loops. One of the stations attached to the network controls access to and communication over the channel by other stations. Once a station is permitted by the control station to transmit a message, the message travels around the ring to its destination without further intervention of the control station. Loops may also be designed so that all message exchanges pass through the control station. In this case, the loop funtions just like a star network with a centrally located control station.

There are a number of considerations for configuring ring networks. Rings must be physically arranged so that the stations are fully connected. Lines have to be placed between any station and its two adjacent stations each time an addition is made. Thus, it is often difficult to prewire a building for ring networks in anticipation of stations to be added in the future.

Failure of a station or an active component, adding a new station, or any other break in the ring configuration will almost always cause the network to stop functioning. Steps can be taken to allow bypass of failure points in distributed rings, although this usually increases the complexity of the repeater at each station, as well as the component costs.

2.1.3 Bus Topology

The bus topology illustrated by Fig.2.1(c) funtions in the same way as a multipoint, a single line which is shared by a number of stations.

We have seen that in star networks, all stations are fully connected by their point-to-point links, which are joined at the central switch. Rings consist of separate point-to-point links that are fully connected in a circular arrangement. In contrast to star and ring topologies, bus stations share a single physical channel via cable taps or connectors. Although the approach is different from the star or ring, the bus also creates a fully connected shared channel.

In bus networks, messages placed on the channel are broadcast out to all stations which must be able to recognize those addressed to them. However, unlike stations on a ring, stations on a bus network do not have to repeat and forward messages intended for other stations. As a result, there is no delay or overhead associated with retransmitting messages. Because of the passive role of the stations on the bus, network operation may continue in case of station failures. This makes distributed bus networks inherently resistant to single-point failures. Bus networks are easily configured and expanded; this feature alone is often a major reason in choosing the bus topology for a local area network.

There are many distributed bus networks currently commercially available. This is attributable to the fact that the transmission media, usually coaxial cable, and transmission technologies most commonly used for the bus networks have been in use for some time by the cable TV industry.

2.2 CHANNEL ACCESS

Access techniques are the means by which stations actually gain the use of the network channel. They are used to mediate between stations that are competing for the use of the channel and are generally classified as either Polling or Random Access (Contention technique).

2.2.1 Polling

Polling techniques determine the order in which stations can take turns accessing the network, so that direct conflicts conducing to access collisions between stations are avoided. Polling is thus referred to as a noncontention method of network access. A prime application of the polling model is in local distribution where the *roll-call polling* and the *hub polling* have been most commonly adopted in practice[26]. In roll-call polling, each station is interrogated in turn by the central processor. On the arrival of a polling message, the station transmits messages waiting to the central processor. Upon completion of transmission, polling of the next station is initiated.

A big influence on performance in roll-call polling is the time expended in the dialogue between the central processor and the individual stations[26]. In systems where stations can communicate with one another this time may be reduced by the hub polling technique. Hub polling is usually easier to implement in loop configurations such that all the stations are serially connected to the central processor. The central processor initiates polling by interrogating the station at the end of the loop. This station transmits its waiting messages, if any, and the opportunity to transmit messages is passed on to the next upstream station. The latter station adds its messages followed by a polling message, etc. In this way, at the completion of the polling cycle, with all stations interrogated, the central processor regains control. We may notice that in going from roll-call polling to hub polling, there is a reduction of the role of the central processor. This decentralization of control continues to the third local distribution category which plays an important role in LANs. The two most frequently discussed methods of decentralized control of polling used in local area networks are token passing and slotted rings.

Token Passing is most often associated with ring topologies. A token ring consists of a set of stations connected serially by a transmission medium. This mechanism allows each station, in turn and in a predetermined order, to receive and to pass the right to use the channel. Information is transferred sequentially from one active station to the next. A station gains the right to transmit when it detects a free token passing on the medium. The token is a control signal, a special bit pattern usually several bits in length, that circulates around the ring from station to station following each information transfer.

Any station wishing to transmit must wait until it detects a token passing by. Upon detection of a free token, a station may capture the token and set it to busy by changing one bit in the token, which transforms it from a token to a start of message. The possession of this special bit pattern gives the station exclusive access to the network for transmitting its information (specifying the destination address), thus avoiding conflict with other stations that wish to transmit. In fact, there is no more token on the ring, so other stations wishing to transmit must wait. Stations on the ring check the message as it passes by and are responsible for identifying and accepting messages addressed to them, as well as repeating and passing on messages addressed to other stations. On completion of its information transfer (messages usually must circulate back to the sending station to confirm receipt by the destination station), the station generates a new free token which provides other stations the opportunity to gain access to the ring. We should note that the time required for the messages to circulate back to the expeditor introduces a delay and inefficiency to transmissions, since the channel cannot be used by other stations during this time.

There exist a great number of other principles either employed in existing prototype systems and products, or proposed in the litterature, which employ a variation of token passing access control in ring networks. *Slotted Ring* and *Buffer Insertion Ring* are two of these methods. In slotted rings [21,22], usually a constant number of fixed-length slots or frames circulate continuously around the ring. Each frame generally contains bits allocated for source and destination addresses, control and parity information, and data. Any station ready to transmit waits for a free or unused slot, inserts information into the appropriate field and sets a bit to indicate that the frame is full. When the sender receives back the busy slot, it changes the indicator to free, thus preventing hogging of the ring, and guaranteeing fair sharing of the bandwidth among all stations.

In buffer insertion rings, the contention between the traffic to be transmitted by a station and the data stream already flowing on the ring is resolved by dynamically inserting sufficient buffer space into the ring at each ring adapter [23-25]. In contrast to the token ring where the sender is responsible for removal of the frame it transmitted, this function is performed by the receiver. Buffer insertion rings do not allow interruption once the transmission of a message has begun [26]. Messages arriving on the line from other stations are buffered while the locally generated message is being transmitted. In buffer insertion rings, messages are kept intact and more than one station may transmit at a time. It is important to note that the flow into a station consists of two streams, locally generated traffic and through-line traffic from other stations [26]. Depending on the implementation, one or the other of these streams is given nonpreemptive priority. While a message from one of these streams is being transmitted, messages from the other are buffered. After the transmission of a message, any message of higher priority stored in the station are transmitted. When all higher-priority messages are transmitted then any lower priority messages present are transmitted [27].

The American National Standards Institute (ANSI) is working towards a new standard known as *Fiber Distributed Data Interface (FDDI)* [28,29]. It is a 100 Mbits/s token ring using fiber-optic cable to provide high-speed packet communications. Recently, FDDI is viewed as a high-performance backbone LAN to interconnect lower speed LANs. FDDI uses a medium-access control strategy based on the token-ring standard. In addition to the FDDI high data rate compared to 1 and 4 Mbps for 802.5 and the fact that FDDI uses fiber optic instead of shielded twisted pair cables two major differences between the 802.5 and FDDI are of interest. First, an FDDI station does not attempt to seize a token by "flipping" a bit. Because of the high data rate of FDDI, it was considered impractical to impose this requirement. Second, in FDDI, a multiple-token operation has been defined as opposed to the single-token operation. The FDDI method is for a station to issue the token immediately after it completes transmission of its last message (even it has not begun to receive its own transmission), whereas in the 802.5 standard the station has to wait until the messages it has sent propagate back around to itself before issuing a new token. Again, to achieve higher throughput, this technique is needed.

The techniques described above for allocation and control of a common shared channel are designed so that there are no conflicts between stations that wish to transmit. Messages are not damaged by collisions and thus do not require retransmission resulting from collision. Contention tehniques, on the other hand, anticipate conflicts and actually use them as part of the design to allocate the shared channel.

2.2.2 Contention Techniques (Random Access)

The first random access mechanism was proposed by Abramson, who was faced with the task of interconnecting various terminals dispersed in a wide geographical area to a central computer via a radio network [30]. This network, known as the ALOHA system, uses the simplest form of random access, a form of completely distributed control. A station ready to send simply sends its message, disregarding the state of the channel. Clearly, collisions may occur when more than one station send their message at the same time, in which case the terminals affected retransmit after a randomly chosen time-out. The major benefits envisioned by the inventors were elimination of the access delay owing to waiting for the polling signal and simplicity of the system. There was no central control needed, the stations operated in a purely distributed fashion. However, simulation and analytical studies showed that the performance was tarnished by a reduction of the usable bandwidth (18 percent in the case of PURE ALOHA) due to collisions. Moreover, there is a basic instability in the system. The basic ALOHA access technique can be improved by rudimentary coordination among the stations. By slotting the time and allowing stations to transmit only at the beginning of a slot the rate of collisions is reduced by half since only messages generated in the same slot interfere with one another. This doubles the throughput of the system. This modification is called *slotted ALOHA* [31].

During the late 1960's, R. Metcalfe was faced with the task of constructing a fast network linking small computers. He drew on the ideas of random access in the domain of packet radio and envisaged a packet radio network on a coaxial cable, which he termed *Ethernet.* One of Metcalfe's contributions was carrier sensing, a technique which significantly improved the achievable performance of random-access systems. This extension of ALOHA which is pa'ticularly appropriate for local area networks is called *Carrier Sense Multiple Access* with *Collision Detect (CSMA/CD).* The Multiple Access feature of CSMA/CD allows any station to send a message immediately upon sensing that the channel is free. As opposed to token passing, in which a station that wishes to transmit must wait for the token to circulate to all stations that precede it on the channel, CSMA/CD contains no such delay, and therefore, can allow quicker access.

Carrier Sense is the ability of each station to detect any traffic on the channel. This is also called listen-before-talking. Whenever a station senses that there is traffic on the channel, it defers transmission. However, because of the propagation delay (time it takes for a signal to travel across the network), two stations could detect that the channel is free either at or close to the same time, since each will not yet have detected the signal of the other. Consequently, a collision between two messages will occur. Collisions are detected by stations while they are transmitting by monitoring the electrical signal level of the channel, which is changed by message collisions. This is called listen-while-talking. Collision Detect is the ability of a transmitting station to sense a change in the signal level of the channel and to interpret it as a collision.

Upon detecting a collision, each station involved aborts its transmission, backs off for a brief interval, and attempts to transmit again after a fixed or a random period. It is obvious that the selection of random intervals before retransmission is more effective in avoiding further collisions. If there are successive collisions, the stations involved back off and wait for a longer period each time.

To ensure that all the stations are able to hear a collision on the bus channel, frames (a frame represents the entity transmitted by a station when it has access to the medium) must be of a minimum length. This size, slightly greater than the round trip propagation delay, is called slot time. In fact, once a slot time has passed in the transmission of a frame, there is no longer possibility for collision since all stations have had the opportunity to detect traffic on the channel. The transmitting station is said to have acquired the channel and the remainder of the frame is sent contention free. The critical parameter that determines the performance of the CSMA/CD bus is the ratio of the slot time to the average duration of transmitted frames. Since the propagation delay (slot time) is independent of the data rate, this ratio increases with the data rate. Theory [63,32] shows that a CSMA/CD bus behaves ideally as long as this ratio is sufficiently low. If, for reasonable traffic loads, it exceeds .02-.05, the increasing collision frequency will cause significant performance degradation. It follows that the most efficient use of the CSMA/CD mechanism is when frames are larger; the longer the frames that are broadcast, the fewer are the collisions relative to the total time the channel is used.

2.3. REVIEW OF POLLING MODEL

The main purpose of this section is to review some relevant results on polling and not to provide an exhaustive survey of the vast literature that exist on this model. The interested reader is referred to [5] for a tutorial survey.

Polling is one of the techniques which allows messages generated by different stations to share the same tranmission medium. Polling models can be used to study a variety of systems including Token-Passing. A queueing model of a Polling system is a *single server, multiqueue system* with a *cyclic service discipline:* a server (the single communication channel of the ring network) is shared by customers or stations (e.g. terminals), and the cyclic service discipline specifies how this sharing is accomplished. The time needed to switch service from one station to another is called a *walktime* (sometimes known as *switchover time*, or *reply interval*).

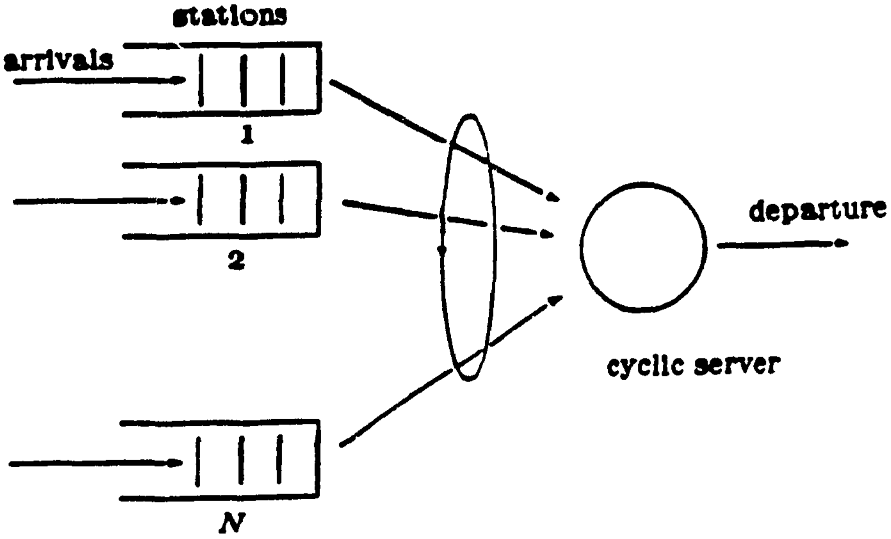


Fig. 2.2: Schematic view of a Polling system with N stations.

The entity for service may be a fixed length *packet* or a variable length *message* (several packets). When all statistical specifications for message length, message arrival process and walking time are the same for all stations, it is said the system has identical stations; otherwise, it corresponds to the case of nonidentical stations.

The time may be continuous or discrete; in the latter, the packet transmission time is generally chosen as the (fixed-length) slot size.

Figure 2.2 illustrates a schematic view of a polling system consisting of N stations.

Polling systems can be classified by the *buffer capacity* at each station. Two buffer capacity models have been studied extensively; these are *single message buffer* and *infinite buffer*. There has been little work on the *general finite buffer* system.

Single Message Buffer

In the conventional one-message buffer system, each station has a buffer which can store at most one outstanding message. New messages arriving at an occupied buffer are lost, or new messages are generated at each station only after the service completion of the previous message.

The analysis of such a polling system has a long and interesting history[6]. For example, there is an analogy between polling systems and machine patrolling [26,33] in which a repairman examines N machines in a fixed sequence. If a machine is broken, he pauses to make repairs. This is analogous in polling systems to message transmission. The overhead that is incurred is the time required to walk between machines.

The performance measures are the *average polling cycle* defined as the time interval beginning at the polling instant of station 1 (for example) and ending with the completion of the reply interval from station N to station 1, and the *mean message delay.* We should note that in all surveys of previous works concerning the analysis of polling systems with single message buffers, the mean message delay (response time) in the case of general $R_i(s)$ and $M(s)$ (the Laplace - Stieltjes transforms of the distribution functions for the walktime to station i and the message transmission time respectively) has not been previously available.

In [5] Takagi considered a symmetric case where general distributions $R(s)$ and $M(s)$ are identical for all stations. His model and Mack's model in [3] are complementary in the sense that Takagi's can handle variable walk times (Mack's cannot) but Takagi's cannot be applied to asymmetric systems (Mack's can in the case of constant walk times). The case of constant message transmission time and constant reply intervals is explicitly solved. Even the distribution of the waiting time is explicitly given. However, when the message transmission time and/or reply interval are generally distributed, Takagi has to solve $O(2^N)$ equations, even if the distribution functions are identical for all stations.

Another variation of the single buffer system is the *buffer relaxation* system. In the buffer relaxation system, a newly arriving message can be stored in its buffer after the previous message's transmission has been started. An exact analysis of asymmetric single buffer relaxed system can be found in [4].

Infinite Buffer Systems

In this case, the buffer can hold any number of messages or packets without loss. Three basic types of service disciplines have been identified and intensively studied. These are *Exhaustive* service or "*Come right in*" discipline, *Gated* service or "*Please wait*" discipline, and *limited* service.

In the exhaustive (e) service, the server will continue to serve a station until its buffer becomes empty. Messages or packets arriving at a station while the server is present are transmitted in the same service period (i.e. in the same cycle). The earliest work in the area involved just two stations with zero walktime [34-36]. Such a particular case (N=2) has been called *alternating priority discipline* in the queueing theory.

The case of arbitrary number of stations is analysed in [6] and [7] where a continuous-time model with Poisson message arrivals and zero overhead (this is a drawback) are assumed. Exhaustive service with nonzero walktimes for the case $N=2$ was studied in [37] and [38]. For a long time, the only work on an arbitrary number of stations with nonzero overhead was by Liebowitz[8], who suggested the famous independence approximation by assuming that the contents of different stations are independent. In fact, independence among queues tends to hold for high overhead and light loading. Liebowitz first found an expression for the cycle time and based on this expression Hayes and Sherman[39] obtained an expression for the delay. The problem is that an analysis of cycle time in this case is not trivial, that is why an approximation was used. In 1972, both Hashida[9] and Eisenberg[10] separately published results on exhaustive service systems with an arbitrary N and nonzero walktime.

The analysis for the case of non identical stations for a continuous time model may be found in [11] and in [5] where $O(N^2)$ equations have been derived.

In the completely symmetric case (i.e. all the stations have exactly the same characteristics), by cleverly exploiting the symmetry, it is possible to obtain the mean waiting time without determining expressions for waiting time or queue length distributions. For this case of identical stations, we have [5,9]

$$E[W]\Big|_{exhaustive} = \frac{\delta^2}{2r} + \frac{N[\lambda m^{(2)} - r(1+\rho)]}{2(1-N\rho)} \tag{2.1}$$

where $\rho = \lambda m$.

N denotes the number of stations, λ the mesage arrival rate, m the mean message transmission time, $m^{(2)}$ the second moment of the message transmission time, r the mean walktime and δ^2 the variance of the walktime.

In the gated (g) service ("Please wait") discipline, the server continues to serve only those messages or packets which are waiting at the station when it is polled. Those messages or packets which arrive during the service time are set aside to be served at the next polling. The gated service system with N nonidentical stations with zero reply intervals is considered in [6,7]. An early approximate treatment for the case of nonzero intervals is in [8] and [40]. Exact analysis of the gated service models with an arbitary number of nonidentical stations and nonzero walktimes may be found in [5] where $O(N^2)$ equations required to solve the problem are derived. For the case of identical stations, we have

$$E[W]\Big|_{gated} = \frac{\delta^2}{2r} + \frac{N[\lambda m^{(2)} + r(1+\rho)]}{2(1-N\rho)} \tag{2.2}$$

In the limited, also called non-exhaustive (ne), service system, a station is served until either the buffer is emptied or a specified number of messages or packets are served, whichever occurs first.

Analysis of this system for the case where $N=2$ and at most one message is served from each station at a time with zero switchover time is in [41], where the term *alternating service discipline* is used. A similar system ($N=2$) with nonzero reply intervals is considered in [42]. The mean message waiting time for N identical stations with nonzero walktimes where, at most one message is served from each station at a time, is available in [5]. Approximate analyses for N nonidentical station case are given in [43],[44] and [45].

We should note that the only limited service case for which we can evaluate the mean waiting time is the case of identical stations with "limited-to-one" service. In this case of identical stations, we have

$$E[W]\Big|_{limited} = \frac{\delta^2}{2r} + \frac{N[\lambda m^{(2)} + r(1+\rho)] + \lambda\delta^2}{2[(1 - N(\rho + \lambda r)]} \tag{2.3}$$

We do not have an explicit solution to any asymmetric system, including the case of two stations.

An approximate treatment for a system of N stations with zero reply interval, where at most K_i packets are served from station i is in [46] where the term *chaining* is introduced.

Comparing the three service disciplines we have presented, we get

$$E[W]\Big|_{exhaustive} \leq E[W]\Big|_{gated} \leq E[W]\Big|_{limited}$$

for the symmetric, continuous time systems.

Pseudo-conservation laws in cyclic-service systems

The principle of work conservation has proven to be very useful in the analysis of queueing systems with a non-First In First Out (FIFO) service discipline. When no work is created or destroyed within the system, the amount of work present should not depend on the order of service, and hence should equal the amount of work in the corresponding system with FIFO service discipline.

Priority systems with switchover times between different stations do not possess the work-conservative property, because the server is forced to stay idle although work may be present within the system. We should notice that the introduction of reply intervals can be interpreted as creation of additional work within the system. A basic example of such systems is the single-server multiqueue model with cyclic service and walk times. We have seen that this model plays an important role in the analysis of polling schemes, and hence finds an interesting application in local area networks with ring topology employing a medium access protocol based on token passing.

Because of the importance of this cyclic-service model, and moreover the complexity of its mathematical analysis, the recent discovery of *pseudo-conservation* laws, expressions for a weighted sum of the mean waiting times at the various queues of the cyclic system [11,47], has attracted considerable attention. The basic idea is that the amount of work in the cyclic- service system at an arbitrary epoch in a service interval is distributed as the sum of two independent quantities: the amount of work in the "corresponding" M/G/1 queue at an arbitrary epoch in a service interval and the amount of work in the cyclic-service system at an arbitrary epoch in a switching interval. This concept is proved as a theorem in [48] where a stochastic decomposition for the amount of work in cyclic-service systems with mixed service strategies is derived. The obtained decomposition result (natural extension of Kleinrock's conservation law [49]) is then used to obtain a pseudo-conservation law for such systems. The four service disciplines considered are: exhaustive (e), gated (g), limited-to-one or non-exhaustive (ne) and decrementing or semi-exhaustive (se) services. The decrementing (semi- exhaustive) service system is defined as follows: when there is at least one message at a station, the service for that station continues until the number of queued messages decreases to one less than the number present upon the arrival of the server at that queue.

If $E[W_i]$ is the mean waiting time at station i, the pseudo-conservation law may be written as [48]:

$$\sum_{i\in e}\rho_i E[W_i] + \sum_{i\in g}\rho_i E[W_i] + \sum_{i\in ne}\rho_i[1-\frac{\lambda_i s}{1-\rho}]E[W_i] + \sum_{i\in se}\rho_i\left[1-\frac{\lambda_i s(1-\rho_i)}{1-\rho}\right]E[W_i]$$

$$= \rho\sum_{\forall i}\frac{\lambda_i m_i^{(2)}}{2(1-\rho)} - \sum_{i\in se}\frac{\lambda_i^2 m_i^{(2)}\rho_i s}{2(1-\rho)} + \rho\frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)}\left[\rho^2-\sum_i\rho_i^2\right]$$

$$+\frac{s}{2(1-\rho)}\sum_{i\in g,ne}\rho_i^2 \qquad (2.4)$$

where

$$\rho_i = \lambda_i m_i,\ \rho = \sum_{i=1}^{N}\rho_i \text{ and } s = \sum_{i=1}^{N} r_i .$$

λ_i represents the message arrival rate at station i, m_i the mean message transmission time at station i, $m_i^{(2)}$ the second moment of the message transmission time at station i, and r_i the mean walktime to station i.

The result expressed by the above equation is really interesting for two important reasons. The first reason is that allowing mixed service strategies enables Boxma to unify the proof for mean waiting time results of the four cyclic service systems with the same service discpline at all queues. The second reason has more interest in the interconnection of local area networks. In fact, in a system of interconnected token passing networks where several rings are connected to each other by gateways for example, the queues which represent the gateways may have higher priority than the other queues on the ring. The service at an ordinary station usually is limited. The internetwork messages in such system have to pass through the gateways. Since their path is much longer compared to the one of local messages, one may consider other service disciplines to model the preferential treatment received by these queues.
This expression seems also to be very useful in other respects. It is useful for obtaining approximations for individual mean waiting times. Such approximations are badly needed in analytically untractable cases but also in analytically tractable cases; the latter because, when the number of queues is large, the numerical computation of the exact formulas may become very cumbersome[48]. This expression can also be used to study asymptotics, yielding information about what happens when the number of queues becomes very large or when the offered traffic at a particular queue approaches its stability limits[48].

2.4 PERFORMANCE CHARACTERISTICS OF ACCESS PROTOCOLS

The delay-throughput characteristic of the medium-access control schemes, and the system behavior at the load saturation point, constitute the two major performance aspects of primary interest. The most important performance results of Polling systems which may be used to analyze Token-Passing rings have been given in the previous section. Before discussing the delay-throughput characteristics of the accessing techniques, it is important to note that there exist many performance analyses of bus systems. Metcalfe and Boggs first derived a simple formula for prediction of the capacity of finite-population and fixed message length Ethernets[60]. This is of interest because of its simplicity and because it was presented along with the first published description of the Ethernet implementation. The topology is assumed to be a balanced star, i.e., the propagation delay between every pair of stations is assumed to be τ. The channel is slotted in time. Packets arriving during a slot wait until the start of the next slot at which all ready stations simultaneously begin transmission. The slot duration is chosen to be at least 2τ so that any collision can be detected and transmission aborted within one slot. If N is the number of stations, message arrivals at each station, both new and retransmissions, are assumed to be such that each station attempts to transmit with optimum probability of $1/N$ in each slot. Later, more sophisticated stochastic analyses of delay and throughput appeared.

The original CSMA protocol was investigated by Tobagi and Kleinrock[49,61] for packet radio communication. Tobagi and Hunt[62] applied the method of embedded Markov chains to extend the results of CSMA to cover the case of collision detection (CSMA/CD). Tobagi and Hunt first present a model for estimating the maximum throughput of a CSMA/CD network under the assumption of an infinite population. The topology is assumed to be balanced star. The channel is assumed to be slotted in time, with slot duration τ, to permit the formulation of a discrete-time model. Retransmission delays are assumed to be arbitrarily large since the aim is to obtain the maximum throughput. Hence, the arrivals of new and retransmitted messages can be assumed to be

independent and to form a Poisson process. Next, a delay-throughput analysis is presented for finite population system. It is shown that for sufficiently large population sizes, e.g., 50 stations (at the ratio of propagation delay and mean message transmission time equal to 0.01), the maximum throughput predictions of the finite and infinite population analyses converge, relatively insensitive to the number of stations. Lam[63] used a single-server queueing model to approximate the distributed protocol of the Ethernet. The analysis is attractive as the expressions for delay and throughput are simpler than those obtained by Tobagi and Hunt. The balanced star is also assumed. New messages are assumed to arrive from an infinite population of users in a Poisson process. As in the Metcalfe and Boggs model, the channel is assumed to be slotted with a slot duration of 2τ. Unlike the Ethernet, the retransmission algorithm is assumed to be such that the probability of a successful transmission in each slot is $1/e$.
In most cases, the classic approaches to modelling CSMA/CD LANs are Markovian models for the medium with the assumptions of a Poisson arrival process. In these models, an infinite population is assumed and an incoming message always arrives at an empty station[62]. Recently, the classic Markovian model is extended[64] to deal with a finite population with dissimilar arrival rates among stations. Since the local concentration at a station will reduce medium contention, it should improve the overall performance of a CSMA/CD LAN.

In the following, we use some of the results reported in [50] and [51] to compare the efficiency of the two most commonly used medium accessing techniques: CSMA/CD and token passing. The result used by Bux[51] for his comparison is the solution given by Lam[63]. This solution has been derived under the assumption that the system is stabilized by using a suitable adaptive algorithm. A slight modification has been made by Bux. A consequence of the assumption of slotting the channel with slot length 2τ is that even if the utilization approaches zero, messages have to wait for the time τ on the average before they are transmitted. Therefore, Bux heuristically modified the delay formula in[63] by reducing the mean delay by τ. With this modification, the mean delay (transfer time) d of the bus with CSMA and collision detection is given by

$$d = \frac{\lambda\{\overline{m^2} + (4e + 2)\tau\overline{m} + 5\tau^2 + 4e(2e-1)\tau^2\}}{2\{1 - \lambda(\overline{m} + \tau + 2e\,\tau)\}} + \overline{m}$$

$$+ 2\tau e - \frac{(1 - e^{-2\lambda\tau})(\frac{2}{\lambda} + \frac{2\tau}{e} - 6e)}{2\left\{M(\lambda)e^{-\lambda\tau}\frac{1}{e} - 1 + e^{-2\lambda\tau}\right\}} + \frac{\tau}{2} \qquad (2.5)$$

where λ denotes the aggregate arrival rate, $M(s)$ the Laplace transform of the probability density function of the message transmission time, $\overline{m}$ its mean and $\overline{m^2}$ its second moment.

Before commenting on the results for CSMA/CD and Token ring, it is important to see how Bux obtained his results on Token ring. Even though it is only a matter of substitution, this introduces some concepts used in practice that are not obvious from equations (2.1-3). In Bux's notation [50]

$S =$ number of stations $\qquad \lambda =$ total message arrival rate

$T_p =$ message transmission time $\qquad \rho = \lambda E[T_p]$ channel utilization

$\tau =$ ring round-trip propagation delay

τ is given by

$\tau(\text{sec}) = \text{P(sec/km)} \times l(\text{km}) + N \times L_R(\text{bits}) \times C^{-1}(\text{sec/bit})$

where P represents the signal propagation delay, l the length of the ring , L_R the bit latency at each station and C (bits/sec) the transmission on the ring. Bux results are for exhaustive service. Thus, by substitution

$$N \rightarrow S,\ \tau \rightarrow \frac{\tau}{S},\ \delta^2 \rightarrow 0,\ \lambda \rightarrow \frac{\lambda}{S},\ m^{(2)} \rightarrow E[T_P{}^2],\ \rho \rightarrow \frac{\lambda}{S}E[T_p] = \frac{\rho}{S}$$

in eq. (2.1), we get

$$E[W] = \frac{\tau(1-\rho/S)}{2(1-\rho)} + \frac{\rho E[T_p{}^2]}{2(1-\rho)E[T_p]} \tag{2.6}$$

Adding to the above equation for mean waiting time the mean message transmission time $E[T_p]$ and the mean propagation delay to the destination (assumed to be uniformly distributed over the ring) $\tau/2$, we have the mean message delay d as

$$d\Big|_{exhaustive} = \frac{\tau(1-\rho/S)}{2(1-\rho)} + \frac{\rho E[T_p{}^2]}{2(1-\rho)E[T_p]} + E[T_p] + \tau/2$$

For the other types of service, we obtain by the same way

$$d\Big|_{gated} = \frac{\tau(1+\rho/S)}{2(1-\rho)} + \frac{\rho E[T_p{}^2]}{2(1-\rho)E[T_p]} + E[T_p] + \tau/2$$

$$d\Big|_{limited} = \frac{\tau(1+\rho/S)}{(1-\rho-\lambda\tau)} + \frac{\rho E[T_p{}^2]}{2(1-\rho-\lambda\tau)E[T_p]} + E[T_p] + \tau/2$$

Figures 2.3(a) and (b) illustrate the delay-throughput relation of token ring and CSMA/CD bus for two data rates: 1 Mbps and 10 Mbps. From these results, the following conclusions can be deduced:

1) both systems perform equally well at a data rate of 1 Mbps;

2) token ring has better performance characteristics over a wide range of parameters, when the data rate is increased to 10 Mbps.

The validity of the remarks earlier about the ratio of slot time (propagation delay) to average message transmission time, is supported by Fig 2.3(b). In

Fig.2.3(b), all parameters are the same as before except for the length of the cable, which is now 10 km instead of 2 km. The curve for the CSMA/CD bus at 10 Mbps illustrates the impact of the propagation delay, and confirms the important sensibility of CSMA/CD to the ratio of propagation delay to mean message transmission time. As a practical consequence, all CSMA/CD systems specify a maximum distance less than 10 km.

The maximum throughput dependency on transmission speed is demontrated in Fig. 2.4(a) and (b) for the CSMA/CD bus and the token ring, respectively. Figure 2.4(a) shows the maximum throughput for different values of the mean information-field length. It can be seen that the efficiency of the CSMA/CD protocol decreases significantly with increasing speed and decreasing information-field lengths. From Fig. 2.4(b), we can notice that the efficiency of the token protocol is much less sensitive to increased transmission rates and decreasing information-field length compared to CSMA/CD. The single token operation performs reasonably well even though the multiple-token operation is more efficient at high data rates. The token ring standard† defines data signaling rates of 1 and 4 Mbps. Consequently, token passing access on rings is open to significantly higher transmission rates than the ones currently standardized and in use.

† This standard specifies the formats and protocols used by the Token-Passing Ring medium access control (MAC) sublayer, and the means of attachment to the token-passing ring physical medium.

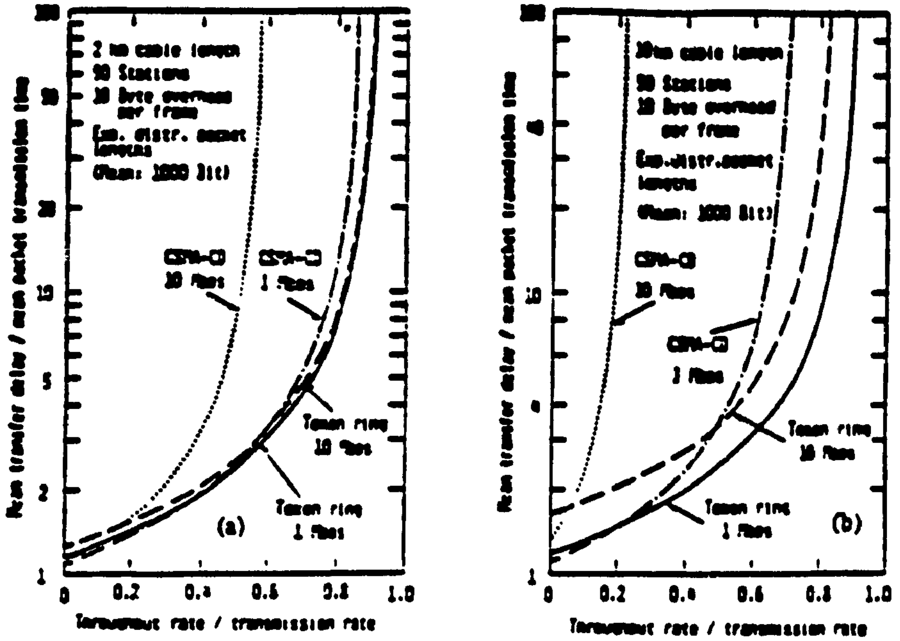


Fig. 2.3: Delay-Throughput characteristic for CSMA/CD bus and Token ring[51]

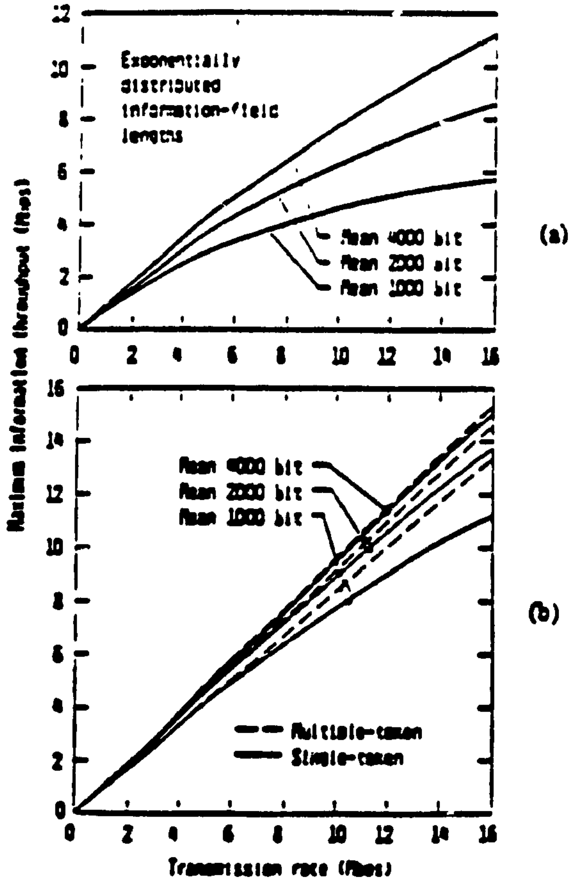


Fig. 2.4: Maximum information throughput versus transmission rate[51].
(a) CSMA/CD. (b) Token ring.

2.5 CONCLUDING REMARKS

Since there is no predetermined order of access by stations in CSMA/CD, there is no guaranty of a maximum waiting time before getting access to the channel. However, since each station randomly bids for the channel, a *statistical* guaranty can be calculated. In many local area network CSMA/CD implementations, the fair distribution of access to the network is combined with a high-speed, high-bandwidth channel, so that the channel is virtually always available to participating stations. Polling networks, on the other hand, offer a *deterministic* guaranty of access to the channel by each station. Access is prede.ermined by a schedule that truly guarantees each station a chance to transmit at a specific point in the polling order. The advantage of the polling approach is that it provides graceful degradation of performance with increasing load.

The performance limitations of CSMA/CD-type contention systems and the fact that these systems do not easily scale up to significantly higher speeds, led to new approaches, the so-called unidirectional broadcast bus systems. Two prominent examples are, Expressnet[65] (modified version of the Unidirectional Broadcast System-Round Robin, UBS-RR) and Fasnet[66]. The approach used in Expressnet is to fold a unidirectional cable onto itself so as to create two channels, an outbound channel onto which the stations transmit messages and an inbound channel from which stations receive messages. The Expressnet achieves a conventional round robin discipline in that stations are given the chance to transmit in a predescribed order which is determined by their location on the unidirectional bus. If a station has no message, it declines and waits for the next round before getting another turn. The basic link configuration of Fasnet consists of two unidirectional buses with signals propagating in opposite directions. In Fasnet, stations are still ordered according to their position on the bus; however, following a transmission, the next station to transmit is always the most

upstream one which has a message and has not yet transmitted in the current round. In this case the order of transmissions within the round may vary depending on the instant at which messages arrive to each station.
Given a unidirectional bus to which stations are connected, an implicit ordering exists among stations of which the access control schemes under consideration make use. The basic operation of these access protocols is cyclic, and each station has the opportunity to transmit once in a cycle. The access method in either case is closely related to a ring protocol and may be regarded as a variant of implicit token passing. The information flows in only one direction on the medium, unlike the usual CSMA/CD configurations, but like CSMA/CD, the essential passivity of the medium is retained.

Optical fibers enjoy a number of "charming" properties that make them natural candidates for high-capacity transmission systems. The well-known characteristics of fiber links such as their capability to allow very high data rates over distances required in real systems, their immunity to electromagnetic interference and their small physical dimensions make them attractive for use in LANs. The fact that the FDDI ring will employ optical-fiber links was mentioned earlier. The first optical fiber networks, Fibernet(optical version of Ethernet)[67] and Hubnet[68], were built in the star configuration. More recently, examples of systems using the ring topology have appeared in the literature[69]. Some recent proposals have been folded bus[65] and double bus structures[66]. Despite the very desirable features of optical fibers, there are still unsolved problems and debates pertaining to the optical components to be used which have prevented a pervasive use of optical fibers in LANs. The tools and procedures for field fault isolation and repair are only in the early stages of development and not widely available. Issues such as optical tap losses (bus systems) or losses in optical bypass relays (ring systems) still deserve much attention.

CHAPTER III

Asymmetric Prioritized Token-Ring

We have seen from the previous chapter that polling is one of the channel accessing techniques which allow messages generated by different stations to share the same tranmission medium. The advantage of the polling approach is that it provides graceful degradation of performance with increasing load. The polling model can be used to study a variety of systems including Token-Passing. The queueing model of a Polling system is a single server, multiqueue system with a cyclic service discipline. Polling systems can be classified by the buffer capacity at each station. Two buffer capacity models have been studied extensively; these are single message buffer and infinite buffer. There has been little work on the general finite buffer system.

In the conventional one-message buffer system, each station has a buffer which can store at most one outstanding message. New messages arriving at an occupied buffer are lost, or new messages are generated at each station only after the service completion of the previous message. Analysis of the conventional one-message buffer systems can be found in [1,2,3,5] for example. Another variation of the single buffer system is the *buffer relaxation* system. In the buffer relaxation system, newly arriving message can be stored in its buffer after the previous message's transmission has been started. An exact analysis of asymmetric single buffer relaxed system can be found in [4].

In the case of infinite buffer systems, the buffer can hold any number of messages without loss. Three basic types of service disciplines have been identified. These are: exhaustive service discipline, gated service discipline, and limited service. In the exaustive service scheme, the server will continue to serve a station until its buffer becomes empty before polling the next station. In the gated service discipline, the server continues to serve only those messages which are waiting at the station when it is polled. Those messages which arrive during the service time are set aside to be served at the next polling. And in the limited service scheme a station is served until either the buffer is emptied or a specific number of messages are served, whichever occurs first. Analyses of infinite buffer systems can be found in [5-11,16].

Most of the papers which appeared about polling systems considered only models without priority structure. However, for systems with priorities, only few treatments have been published [12-15]. The reality is that in data communication networks there often exist different kinds of messages, such as real-time voice, short interactive data, long file transfer, acknowledgement and network control. Therefore, it is desirable to have a priority structure among these messages classes, since the different classes usually have different delay requirements.

This chapter is an attempt to introduce priority mechanism in single server cyclic service disciplines. Our work is based on [16] in which Ibe gave an approximate analysis of asymmetric single-service polling systems without priority structure. Here, we present a mean value analysis of asymmetric prioritized token-ring operation with nonpreemptive priority queueing discipline at each station. At a station, higher class messages are given non preemptive priority over lower classes. The number of priority levels and the walktime vary from one station to another. Moreover the message arrival process and the message transmission time distribution vary not only from one station to another but also from one class to another. Under a limited service dislpline, the performance, as measured by the expected waiting time for each message class at each station is obtained. As mentioned earlier, the limited service system is fair (as compared to exhaustive and gated systems) in the sense that it prevents heavily loaded stations from hogging the server. More interesting characteristics of this service discipline such as station and system stability will be discussed later on in this chapter.

A related study of a two priority model appears in [13]. In this two-priority model, a station can be polled at high or low priority: a low-priority poll always begins with a low-priority service, and occurs only if there is no high-priority messages present anywhere in the ring at the polling instant; otherwise, the station is polled at high priority, and only high-priority messages are transmitted. Within a given priority level, each station is sequentially polled for messages. When all messages at a given priority are exhausted(from the system), the priority of the server drops to the highest priority level of available messages, and polling continues at this new level. Conversely, if while polling at a given level, a message of a higher priority arrives somewhere in the system, then the server priority jumps nonpreemptively up to the priority of this new message, and polling continues at that level. This requires that all the stations have the exact knowledge about the priority level of all the messages in the system at the end of a message transmission time. Clearly, this is physically impossible and such model is not realistic.

We should note that the IEEE 802.5 token ring protocol[12] provides multiple levels of message priority by specifying a reservation scheme in which the token indicates the global ring priority level. The token contains a 3-bit field which indicates the lowest priority that may be currently transmitted; the protocol allows a maximum of eight priority levels. A station may capture the token and transmit messages only if it has at least one enqueued message whose priority is equal to or greater than the priority of the token. If all of a station's enqueued messages have priorities less than the token's priority, the station may bid for a token of lower priority by attempting to set a 3-bit reservation field in the token. The transmitting station always emits a reservation field with a zero value in message headers, thus ensuring that all stations have a chance to bid for the token. Upon completion of its transmission, the transmitting station emits the token at the highest requested priority, saving the previous priority if it was lower. If a station raises the priority of the token from X to Y, and later receives a token of priority Y, with no reservations higher than X, it returns the priority of the token to X. It is important to notice that since the transmitting station issues a token at the higher priority, stations of lower priority cannot seize the token, so it passes to the requesting station or an intermediate station of equal or higher priority with messages to send. Even though the effect of these steps is to sort out competing claims and allow the station with the highest priority message to seize the token as soon as possible, this has the disadvantage of allowing stations with higher priority messages to monopolize the token.

The layout of the chapter is as follows. The description and analysis of the system are given in Section 3.1. The stability conditions for the system are discussed in section 3.2. The simulation flow diagram and results based on relevant examples are given in Section 3.3. and 3.4 respectively.

For most of the development in this and subsequent chapters, the notation formalized in the appendix should suffice. However, where necessary, we will introduce further notation.

3.1. Description and Analysis

We consider a multi-queue system with N stations numbered from 1 to N. Figure 3.1a shows a queueing model of the system and the priority mechanism is described in Figure 3.1b. Each station i ($i = 1,...,N$) has an infinite buffer and receives a certain fixed number of independent Poisson† message arrival streams which belong to one of a set of l_i ($i = 1,2,..,N$) different priority classes. The

† For the Poisson distribution, the probability of k message arrivals in T seconds is $P_k = (\lambda T)^k \exp(\lambda T)/k!$ $k = 0,1,2...$ where λ is the average arrival rate. The interarrival time is exponentially distributed.

number of priority classes varies from one station to another. Our convention is that the smaller the index associated with a class is, the higher is the priority of that class. At a station, higher class messages are given nonpreemptive priority over lower classes. The distribution of the message transmission time varies from one station to another and also from one class to another. We consider a limited service model in which server administers service to one message at a time, if any, from one station and with a finite *walktime,* and then it goes to inspect the next station. The walktime varies from one station to another. We are interested in the mean waiting time (the mean time from the arrival instant to the beginning of transmission) for any arbitrary message of priority class j_c at any station i .

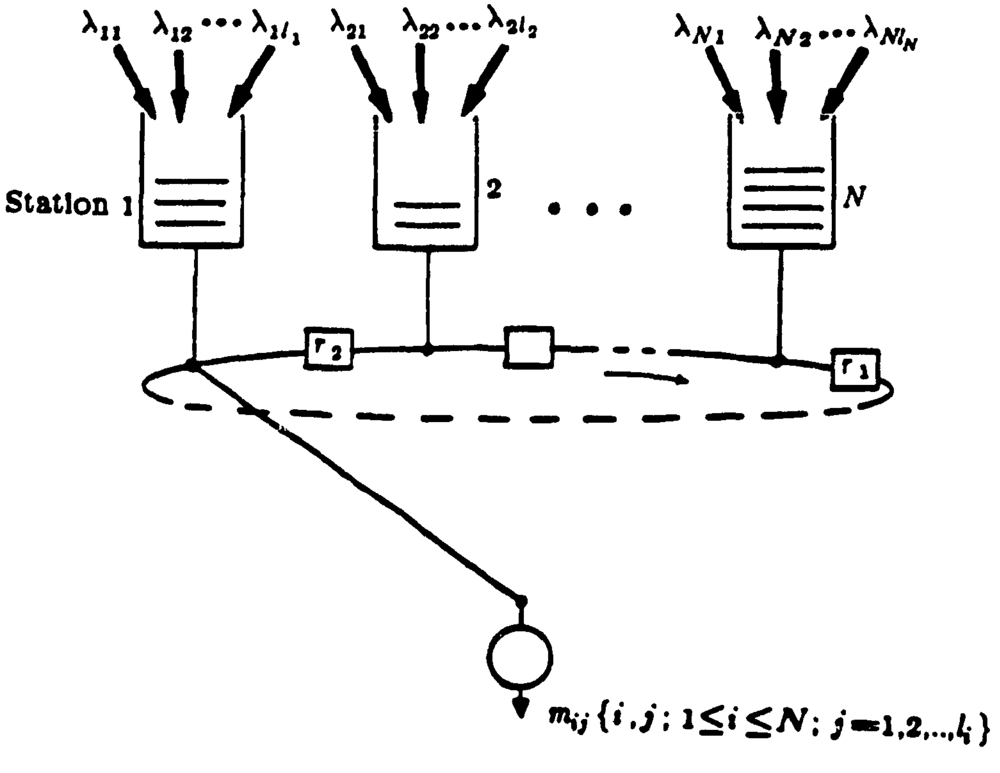

λ_{11} $\lambda_{12} \cdots \lambda_{1l_1}$
λ_{21} $\lambda_{22} \cdots \lambda_{2l_2}$
λ_{N1} $\lambda_{N2} \cdots \lambda_{Nl_N}$
Station 1
2
• • •
N
r_2
r_1
$m_{ij} \{ i,j;\ 1 \leq i \leq N;\ j = 1,2,\ldots,l_i \}$

Fig. 3.1a: Queueing Model.

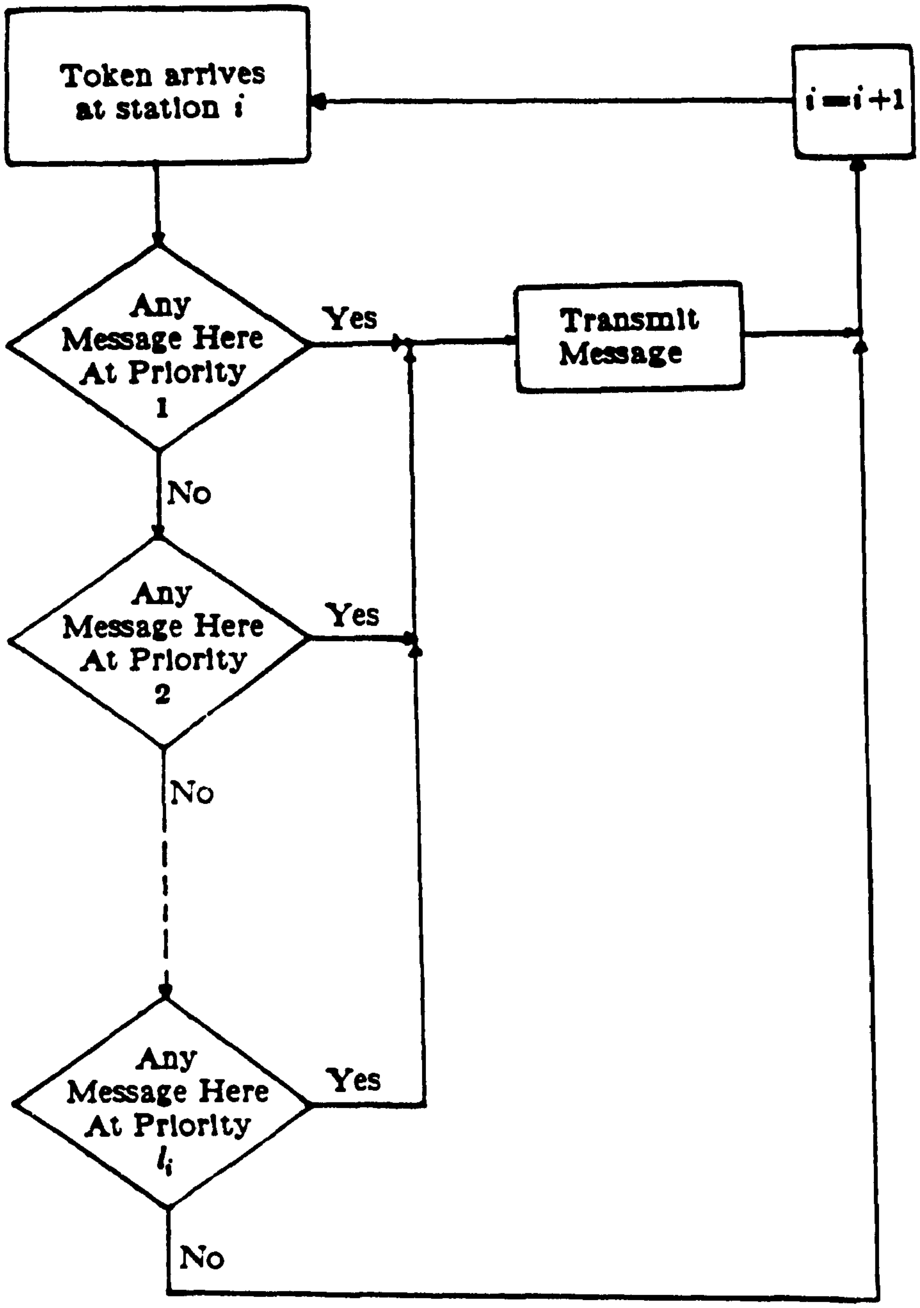

Token arrives at station i
$i = i+1$
Any Message Here At Priority 1
Yes
Transmit Message
No
Any Message Here At Priority 2
Yes
No
Any Message Here At Priority l_i
Yes
No

Fig. 3.1b: Priority Mechanism.

Next, we give some useful terminology. For any strictly cyclic-service system, we can define a *cycle time* as the time between two successive arrivals of the server at a station. The *passage time* from station j to station i given event A is the time from the instant event A occurs to the time the server completes serving station i. The *visit time* of the server at station i is the time between the arrival of the server at station i and its subsequent departure from that station. The *intervisit time* of station i is the time from the instant the server leaves station i to the next time it completes walking to station i. The *walktime* of station i is the time between the instant the server leaves station $i-1$ to the instant it starts service at station i.

We consider a class j_c tagged message which arrives at station i. A well-known attribute of point processes is that the distribution of the cycle in which a random message arrives and the general distribution of cycles is not the same. An arriving message selects longer cycles. An exact study is complicated since the analysis of these cycles is not easy. In order to simplify matters, we assume that the general cycles and the cycles in which our tagged message arrives have identical distributions.

Let W_{ij_c} be the waiting time of this class j_c message at station i. The message will see the system in one of the following mutually exclusive situations :

[1] The server is performing a walktime,

[2] The server is serving a class l message at station i, $l=1,2,...,l_i$

[3] The server is serving a class l message at station j, $j \neq i$; $l=1,2,...,l_j$.

We shall take advantage of the powerful technique of conditioning used so often in probability theory; this technique permits one to write down the probability associated with a complex event by conditioning that event on enough conditions, so that the conditional probability may be written down by simple inspection. The unconditional probability is obtained by multiplying by the probability of each condition and summing over all mutually exclusive and exhaustive conditions.

Before our class j_c tagged message which arrives at station i gets transmitted, not only messages belonging to the same class j_c at station i seen by the message when it arrives should be served but also all higher class messages seen or arrived at the station while it is waiting should be served. The conditional mean waiting time of the class j_c tagged message at station i is given by :

$$E[W_{ij_c} / N_{ij_c}=n_{j_c}, N_{il}=n_l \, , Nw_{il}=nw_l \;\; (l=1, 2, \ldots, j_c\text{-}1)]$$

$$= n_{j_c}\{m_{ij_c} + E[U_{ij_c}]\} + \sum_{l=1}^{j_c-1} n_l\{m_{il} + E[U_{il}]\} + \sum_{l=1}^{j_c-1} nw_l\{m_{il} + E[U_{il}]\}$$

$$+ \sum_{j=1}^{N}\sum_{l=1}^{l_j} Pm_{jl}\{mo_{jl} + E[V_{jil}]\} + \sum_{j=1}^{N} Pr_j\{ro_j + E[V_{ji}]\} \qquad (3.1)$$

where (see Appendix A):

$N_{ij_c} \triangleq$ Number of class j_c messages seen by the class j_c tagged message at station i when it arrives.

$Nw_{il} \triangleq$ Number of higher priority messages which arrive during the waiting time of the class j_c tagged message ($l=1,2,\ldots,j_c\text{-}1$).

$N_{il} \triangleq$ Number of higher class messages seen by the class j_c tagged message at station i when it arrives ($l=1,2,\ldots,j_c\text{-}1$).

$l_i \triangleq$ Number of priority classes at station i

$\lambda_{il} \triangleq$ Arrival rate of class l messages at station i , $l=1,2,..,l_i$.

$\lambda_i \triangleq$ Total arrival rate at station i. $\lambda_i \triangleq \sum_{l=1}^{l_i} \lambda_{il}$

$m_{il} \triangleq$ Mean transmission time of class l message at station i.

$mo_{il} \triangleq$ Mean residual life of a class l message at station i.

$ro_i \triangleq$ Mean residual life of the polling time at station i.

$Pm_{jl} \triangleq$ Probability that the server is serving a class l message at station j when the tagged message arrives.

$Pr_j \triangleq$ Probability that the server is polling station j when the tagged message arrives.

$V_{jil} \triangleq$ Passage time from station j to station i, given that the tagged message arrived when the server was serving a class l message at station j. This time is measured from the time service to the class l message at station j ends.

$V_{ji} \triangleq$ Passage time from station j to station i, given that the tagged message arrived when the server was walking to station j. This time is measured from the time the walktime to station j ends.

$U_{il} \triangleq$ The intervisit time for station i, given that a class l message is served at station i.

By definition $E[V_{ii}] = 0$.

It is important to note that in reality the intervisit times for station i are not independent. There is also a correlation between U_{il} and the conditioned variables. The same dependency can be observed for V_{jil} and V_{ji}.

Let:

$P_{in_{j_c}} \triangleq$ Prob. that n_{j_c} class j_c messages are waiting at station i at the arrival time of the tagged message.

$P_{in_l} \triangleq$ Prob. that n_l ($l = 1,2,...,j_c-1$) messages are waiting at station i.

$P_{inw_l} \triangleq$ Prob. that nw_l class l ($l=1,2,...,j_c-1$) messages have arrived at station i while the tagged message is waiting.

Using the above definitions, the unconditional mean waiting time of a class j_c message at station i may be written as:

$$E[W_{ij_c}] = \sum_{n_{j_c}=0}^{\infty} P_{in_{j_c}} \cdot n_{j_c}\{m_{ij_c} + E[U_{ij_c}]\}$$

$$+ \sum_{l=1}^{j_c-1} \sum_{n_l=0}^{\infty} P_{in_l} \cdot n_l\{m_{il} + E[U_{il}]\}$$

$$+ \sum_{l=1}^{j_c-1} \sum_{nw_l=0}^{\infty} P_{inw_l} \cdot nw_l\{m_{il} + E[U_{il}]\}$$

$$+ \sum_{j=1}^{N} \sum_{l=1}^{l_j} Pm_{jl}\{mo_{jl} + E[V_{jil}]\}$$

$$+ \sum_{j=1}^{N} Pr_j\{ro_j + E[V_{ji}]\} \qquad (3.2)$$

But

$$\sum_{n_{j_c}=0}^{\infty} P_{in_{j_c}} \,.\, n_{j_c} = E\,[N_{ij_c}] \tag{3.3a}$$

$$\sum_{n_l=0}^{\infty} P_{in_l} \,.\, n_l = E\,[N_{il}] \tag{3.3b}$$

and

$$\sum_{nw_l=0}^{\infty} P_{inw_l} \,.\, nw_l = E\,[Nw_{il}] \tag{3.3c}$$

From Little's formula, the average number of class j_c messages waiting at station i may be related to the mean waiting time of class j_c messages at station i as follows:

$$E\,[N_{ij_c}] = \lambda_{ij_c} E\,[W_{ij_c}] \tag{3.4}$$

A similar relation exists between the average number of higher class messages (l = 1,2,..,j_c-1) and their mean waiting time, that is:

$$E\,[N_{il}] = \lambda_{il} E\,[W_{il}] \tag{3.5}$$

The newly arrived higher class messages which receive service before the tagged message are those which arrived while it is waiting, and therefore on average, we have :

$$E\,[Nw_{il}] = \lambda_{il} E\,[W_{ij_c}] \tag{3.6}$$

Using equations (3.3), (3.4), (3.5) and (3.6), equation (3.2) becomes

$$E\,[W_{ij_c}] = \{m_{ij_c} + E\,[U_{ij_c}]\}E\,[N_{ij_c}]$$

$$+ \sum_{l=1}^{j_c-1} \{m_{il} + E[U_{il}]\} E[Nw_{il}] + \sum_{l=1}^{j_c-1} \{m_{il} + E[U_{il}]\} E[N_{il}]$$

$$+ \sum_{j=1}^{N} \sum_{l=1}^{l_j} Pm_{jl} \{mo_{jl} + E[V_{jil}]\} + \sum_{j=1}^{N} Pr_j \{ro_j + E[V_{ji}]\} \tag{3.7}$$

Thus the average waiting time of a class j_c message at station i is given by the following recursive relation :

$$E[W_{ij_c}] = \frac{\sum_{l=1}^{j_c-1} \{m_{il} + E[U_{il}]\} \lambda_{il} . E[W_{il}]}{1 - \sum_{l=1}^{j_c} \lambda_{il} \{m_{il} + E[U_{il}]\}}$$

$$+ \frac{\sum_{j=1}^{N} \sum_{l=1}^{l_j} Pm_{jl} \{mo_{jl} + E[V_{jil}]\} + \sum_{j=1}^{N} Pr_j \{ro_j + E[V_{ji}]\}}{1 - \sum_{l=1}^{j_c} \lambda_{il} \{m_{il} + E[U_{il}]\}} \tag{3.8a}$$

which becomes after successive substitutions

$$E[W_{ij_c}] = \frac{\sum_{j=1}^{N} \sum_{l=1}^{l_j} Pm_{jl} \{mo_{jl} + E[V_{jil}]\} + \sum_{j=1}^{N} Pr_j \{ro_j + E[V_{ji}]\}}{\left[1 - \sum_{l=1}^{j_c-1} \lambda_{il} \{m_{il} + E[U_{il}]\}\right] \left[1 - \sum_{l=1}^{j_c} \lambda_{il} \{m_{il} + E[U_{il}]\}\right]} \tag{3.8b}$$

Voilà!. This simple closed form expression for mean waiting time of any class of message at any station, is the major result of this chapter.

From renewal theory it is known that

$$mo_{jl} = m_{jl}^{(2)}/2m_{jl} \tag{3.9a}$$

and

$$ro_j = r_j^{(2)}/2r_j \tag{3.9b}$$

Adding to equation (3.8b) for the mean waiting time of the class j_c message at station i the mean transmission time m_{ij_c} and the mean propagation delay to the destination (assumed to be uniformly distributed over the ring) $\tau/2$, the mean delay of our tagged message is easily obtained. Hence, the mean waiting time of class j_c message at station i and consequently its mean delay may be known once Pm_{jl}, Pr_j, $E[V_{jil}]$, $E[V_{ji}]$ and $E[U_{il}]$ are known. The determination of these quantities is the objective of the following pages.

Before determining these unknowns, it is important to remark that when $l_i=1$, $(i=1,...,N)$ we have an asymmetric single-service polling system without priority structure and eq. (3.8) reduces to eq. (3) in [16]. Moreover when $N=1$ eq. (3.8) reduces to the average waiting of a class j_c message in M/G/1 with l_1 prioity classes and with some form of server vacations. Let us now proceed to find the unknowns.

Let P_j be the probability that a station j has at least one message in queue at steady state when it is polled. P_j can be evaluated as follows:

The mean cycle duration C is given by:

$$C = \sum_{k=1}^{N} \{r_k + P_k [\sum_{l=1}^{l_k} \frac{\lambda_{kl}}{\lambda_k} m_{kl}]\}. \tag{3.10}$$

By the Poisson assumption, the mean number of messages arriving at station j in one cycle is $\lambda_j C$; and the mean number of messages served in one cycle at station j is P_j . For the station to be stable, the flow of messages in and out of the station during a cycle should balance; that is we must have :

$$\lambda_j \sum_{k=1}^{N} \{r_k + P_k [\sum_{l=1}^{l_k} \frac{\lambda_{kl}}{\lambda_k} m_{kl}]\} = P_j \quad j=1,2,..,N \tag{3.11}$$

The solution to this set of N equations with N unknowns is :

$$P_j = \frac{\lambda_j \sum_{j=1}^{N} r_j}{1 - \sum_{k=1}^{N} \sum_{l=1}^{l_k} \lambda_{kl} m_{kl}} \tag{3.12a}$$

If P_{jl} denotes the probability that the message to be served at station j when it is polled is of class l, then

$$P_{jl} = \frac{\lambda_{jl}}{\lambda_j} P_j \tag{3.12b}$$

The above result may be derived from eq. (3.11) by replacing P_j, λ_j by P_{jl}, λ_{jl}. For notation simplicity, we define two intermediate parameters m_j which represents the overall mean message transmission time at station j,

$$m_j = \sum_{l=1}^{l_j} \frac{\lambda_{jl}}{\lambda_j} m_{jl} \tag{3.13}$$

and Pm_j, the probabilty that the server is serving a message at station j when the tagged message arrives.

The balance of flow in a cycle shows also that :

$$Pm_j . C = P_j . m_j \tag{3.14}$$

The term (Pm_j) represents the portion of the time (cycle) the server is busy serving a message at station j. From eq. (3.14) :

$$Pm_j = P_j \frac{m_j}{C} = \lambda_j m_j \tag{3.15a}$$

Thus, the total rate of messages served at station j which must be equal to the mean number of messages served at that station per unit of time (cycle), i.e., the throughput of the station j, is given by $\lambda_j m_j$.

From eq. (3.12b), it may be seen that, given that station j is busy at a random point, the probability that the next message to be served is of class l is λ_{jl}/λ_j. Now assuming that the tagged message arrives when the server is serving station j, the probability that a class l message is in service is like picking up an interval among l_j discrete possibilities, each one occuring with probability (λ_{jl}/λ_j) and mean length of m_{jl}. The probability that the server is busy serving a class l message at station j when the tagged message arrives Pm_{jl} is then given by

$$Pm_{jl} = \frac{\frac{\lambda_{jl}}{\lambda_j} m_{jl}}{\sum_{l=1}^{l_j} \frac{\lambda_{jl}}{\lambda_j} m_{jl}} Pm_j = \lambda_{jl} m_{jl} \tag{3.15b}$$

which represents the portion of the time (cycle) the server is busy serving a class l message at station j.

The probability that the server is performing walktime to station j when the tagged message arrives is given by :

$$Pr_j = \frac{r_j}{C}$$

$$= r_j \frac{1 - \sum_{k=1}^{N} \sum_{l=1}^{l_k} \lambda_{kl} m_{kl}}{\sum_{k=1}^{N} r_k} \tag{3.16}$$

We consider now the mean intervisit time $E[U_{il}]$:

$E[U_{il}]$ is not an ordinary intervisit time, but an intervisit time in which a class l message is served at station i. Let q_{jil} be the probability that station j has a message to serve in a cycle during which a class l message is served at station i.

Then the mean length of such a cycle is :

$$C_{il} = m_{il} + \sum_{k=1}^{N} r_k + \sum_{k \neq i} q_{kil}\, m_k \tag{3.17}$$

and the number of messages served in such a cycle at station j is given by q_{ji}. Thus, for the flow to balance in this cycle, we must have :

$$\lambda_j \{ m_{il} + \sum_{k=1}^{N} r_k + \sum_{k \neq i} q_{kil}\, m_k \} = q_{jil},\ \ j=1,2,..,N;\ l=1,2,..,l_i;\ j \neq i \tag{3.18}$$

Equation (3.18) represents a system of N-1 equations with N-1 unknowns which may be solved easily. The solution to this set of equations is :

$$q_{jil} = \frac{\lambda_j \{ m_{il} + \sum_{k=1}^{N} r_k \}}{1 - \sum_{k \neq i} \lambda_k\, m_k},\ \ l=1,2,..,l_i \tag{3.19}$$

Thus, the mean intervisit time, given that a class l message is served at station i, is given by :

$$E\,[U_{il}] = \sum_{k=1}^{N} r_k + \sum_{k \neq i} q_{kil} \cdot m_k\,,\ \ l=1,2,..,l_i \tag{3.20}$$

where q_{kil} is given by equation (3.19).

Next, we consider the mean passage time from station j to station i .

$E\,[V_{jil}]$ is a *special passage time*. It is subject to the fact that the tagged message arrived when a class l message was being served at station j.

Let qu_{kjl} be the probability that a station $k \neq j$ has a message to be served in a cycle during which the tagged message arrived when a class l message was being served at station j. We should notice that the arrival time of the tagged message breaks the current transmission at station j into two parts. At the time the tagged message arrives some messages already arrived at station k during the first part of the transmission at station j which corresponds to its age. Station k will continue receiving messages during the residual life of the transmission at station j. From renewal theory, the total mean transmission time of the class l message being transmitted at station j when the tagged message arrived is the mean residual time plus the mean age of its transmission time. Since the mean residual time and the mean age of its transmission time are equal, the total mean transmission time of this class l message at station j is $2mo_{jl}$.

By the same flow balance argument we have used before,

$$\lambda_k \{ \sum_{n \neq j} qu_{njl}\, m_l + 2.mo_{jl} + \sum_{n=1}^{N} r_l \} = qu_{kjl}, \quad k=1,2,..,N,\ k \neq j \tag{3.21}$$

The solution is :

$$qu_{kjl} = \lambda_k \frac{\{2mo_{jl} + \sum_{l=1}^{N} r_l \}}{1 - \sum_{n \neq j} \lambda_n m_n}, \quad k \neq j;\ l=1,2,...,l_j \tag{3.22}$$

Thus :

$$E\,[V_{jil}] = \sum_{k=j+1}^{i} r_k + \sum_{k=j+1}^{i-1} qu_{kjl}\, m_k, \quad l=1,2,...,l_j \tag{3.23}$$

where we have adopted the same conventions as in [16]; that is when the subscript $k > N$, then $k = k$ (modulo N). Also, when $i < j+1$, then the first summation will be taken from $j+1$ to N, and then from 1 to i; and when $i-1 < j+1$, then the second summation will be taken from $j+1$ to N, and then from 1 to $i-1$.

Finally to evaluate E [V_{ji}] , we use the same method and obtain :

$$qv_{kj} = \lambda_k \frac{\{2ro_j + \sum\limits_{l \neq j} r_l\}}{1 - \sum\limits_{l=1}^{N} \lambda_l m_l} \qquad (3.24)$$

where qv_{kj} is defined as the probability that station k has a message to be served in a cycle in which the mean walktime to station j is ro_j, the mean residual walktime.

Thus,

$$E[V_{ji}] = \sum_{k=j+1}^{i} r_k + \sum_{k=j}^{i-1} qv_{kj} m_k, \quad j \neq i \qquad (3.25)$$

This concludes the determination of all the unknowns and therefore the average waiting of a class j_c at station i may be easily evaluated from eq.(3.8).

Before considering the flow diagram of our simulation and some examples, it is important to discuss the *stability* conditions for this system.

3.2. SYSTEM STABILITY

Studies on stability conditions for polling systems deal mainly with system stability. Station stability is particularly useful in limited service polling systems because the interesting feature of this service discipline is that a station can still be stable even when all other stations in the system are unstable. However, in an unlimited service polling system, when one station becomes unstable then all stations become unstable. This is so because the server spends an infinitely long period of time at the unstable station thereby causing the queue length of each station to increase infinitely. But in a limited service polling system, the bounded number of customers that can be served at each station prevents any station from hogging the server. Also, due to the bounded visit *(sojourn)* time at each station, when priority mechanism is introduced, some classes may be unstable while others remain stable.

We define a station in a polling system to be stable if the mean message waiting time is finite [16,53]. In [53] it is shown that for the limited-to-one service polling system, the condition under which a station i is stable is

$$\lambda_i \sum_{j=1}^{N} r_j + \sum_{j=1}^{N} Min(\lambda_i, \lambda_j) m_j < 1. \tag{3.26}$$

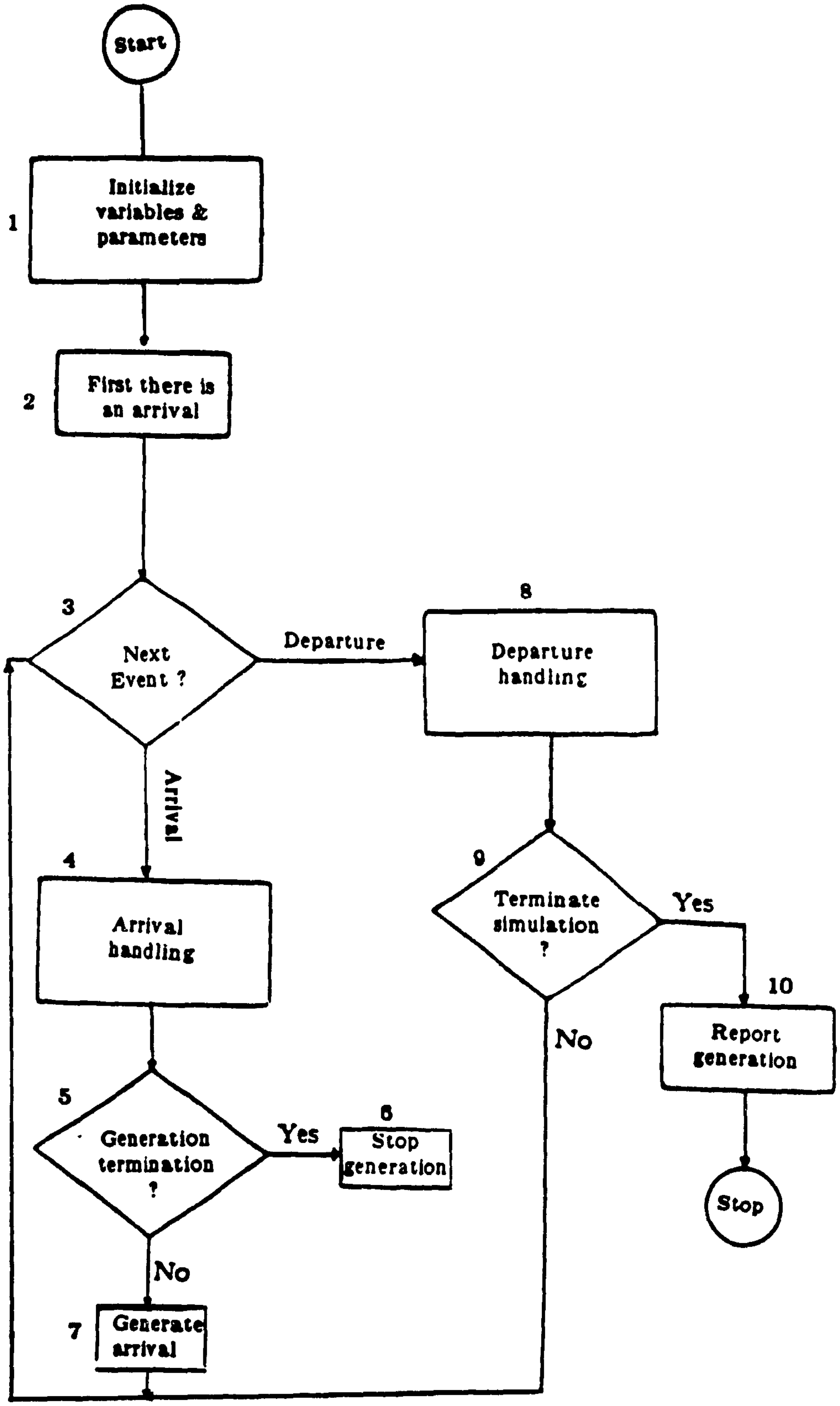
In particular, the stability condition for the station i with the highest arrival rate is given by

$$\lambda_i \sum_{j=1}^{N} r_j + \sum_{j=1}^{N} \lambda_j m_j < 1. \tag{3.27}$$

Since $Min(\lambda_i, \lambda_j) m_j < \sum_{j=1}^{N} m_j$, every station will be stable if the station with the highest arrival rate in the system is stable.

3.3. SIMULATION FLOW DIAGRAM

Before considering the examples, it is important to note that the simulations, in this thesis, make possible the systematic studies of the problems since exact solutions are not available and experiments on the real systems are not available. Our simulation models (see appendix C for the prioritized token ring and appendix D for the interconnection of token rings to be analyzed in chapter V) describe the operation of the systems in terms of individual events of the individual elements in these systems. The interrelationships among the elements (such as queues) are built into the models. Each simulation involves a description of the real way in which the system changes over some period of time; then each model allows the computing device to capture the effect of the elements' actions on each other as a dynamic process. The flow diagram of the simulation of the prioritized token-ring shown below is composed of ten blocks.



Simulation flow diagram for token ring.

Block 1 : Initialization of the system variables and parameters

Here, we make necessary declarations, set the initial conditions of the system, and read data cards. The number of stations in the system and the walk-time duration are given. The minimum and maximum number of messages to be generated are given. Four starting values for random number generators are inputs. They are for the aggregate message arrival process, the message transmission time, the message source station and finally the determination of the message class. The portion of each message class is given and all the variables for statistic gathering are initialized.

Block 2 : First there is an 'arrival

The simulation really starts with the arrival of the first message. The aggregate message arrival to the system is generated according to a Poisson process. It is necessary to supply the expected number of arrivals in a chosen interval and only one uniformly distributed random number. It is important to note that two types of event may occurs in the system. These are arrival of a message and departure of a message. To ensure that the first event to consider is an arrival, it is necessary to assign a large value to the departure event in the initialization block.

Block 3 : Arrival handling

Since we need to keep track of message's movement in the system, a message table is necessary. Conceptually, the message table is a list of messages that are generated during the simulation run. Associated with each message are message's arrival time, message source queue, message class, and message transmission time. This block allows us to associate an arrival time, a source queue and a class to each message. A message source queue is determined using discrete mappings of standard uniform variates. All that is necessary is to divide

up the range of the uniform variate into portions, one for each station source queue. The width of each portion corresponds to the probability that a particular station will be chosen. The rationale behind this scheme is that since the random variates are uniformly distributed over the entire range, then the number of times a random variate falls in a given portion is determined by the width of the portion. We can notice that, for symmetric system, the portions should have the same width. To perform these mappings, it is only necessary to check the value of the random variates. Since bifurcations of a Poisson process are themself Poisson processes, the message arrival process to a station is Poisson. For the aggregate message arrival to the system, we have seen that it is necessary to supply the expected number of arrivals in a chosen interval and only one uniform random number. A message class is determined by the same way and the number of portions needed (at each station) is equal to the number of priority levels at that station. This block increases by one the number of messages in the system and the number of arrivals to the system. Since the message source queue and class are known, the number of messages in that station queue and the number of messages of that class at that station are also increased by one.
The portion of arrivals of that class is computed by dividing the number of arrivals of that class by the total number of arrivals to the system.

Block 5 : Test for terminating generation of message

This test determines when enough messages have been generated. The following two blocks constitute the possible situations which may arise.

Block 6 : Stop generation of message

To terminate generating new messages, one of the following two conditions must be satisfied. The first condition is when the number of messages generated so far is less than the maximum number of messages we have given. In this case, not only the calculated portion of each message class should be equal to the given value, but also the total number of arrivals to the system should at least equals the minimum number of messages we have given. The second condition is when the number of messages generated is equal to the maximum number we have given.

Block 7 : Next arrival generation

The generation of the next arrival is done as in block 2. All that is necessary is to supply the expected number of arrivals in a chosen interval and only one uniformly distributed random number.

Block 8 : Departure handling

This block allows us not only to associate with each message a transmission time, but also to handle the dynamic of the system by the cyclic movement of the server. Initially, the server is at the station where the first message arrived. At each station visited by the server, the priority mechanism depicted in Fig. 3.1b applies. If the station visited is empty, the scheduled server arrival time to the next station in the polling order is simply obtained by adding a walktime to the server arrival timc at the previous station. Otherwise, it is obtained by adding a walktime and a transmission time to the server arrival time at the previous station.

For generating exponentially distributed message transmission time, it is necessary to supply a mean and only one uniformly random number. If the station visited is not empty and after transmission is accomplished, the number of messages in the system and the number of messages in that station are decreased by one. The number of messages of that class which was served is also decreased by one. On the other hand, the number of departures from the system and from that class are increased by one. The remaining messages in that class station queue move one step forward. This structure allows us to save memory space for the message table since the table entries are created and destroyed as messages arrive and depart, respectively.

Block 9 : Test for terminating simulation run

This test determines when the simulation run should be terminated. Since block 4 ensures that enough messages have been generated, the simulation run is terminated when all the generated messages depart from the system.

Block 10 : Report generation

When the simulation run is terminated, the report generation block allows us to print out summary statistics of the simulation run.

After being constructed, the simulation models are driven by generating input data (probabilistic simulations) and then simulate the actual dynamic behavior of the systems. As any technique of conducting sampling experiments on the model of the system, the simulated experiments should be regarded as ordinary physical experiments, and should be based on sound statistical techniques. For the data analysis, we have used the method of replicating the simulation experiment. We run the experiment six times using different and independent streams of random numbers for different runs.

The simulation program ERIC-MAUREEN in Fortran/77 runs in a VAX-UNIX environment, but is easily transportable to other machines. The minimum and maximum number of messages to be generated are given as inputs and in all the independent runs, more than 100,000 messages were generated. Six replications of the simulation were used in order to obtain each point. Thus, a t-distribution with five degrees of freedom† were used.

† For calculation of degree of freedom see appendix B.

3.4. EXAMPLES

We consider, in this section, some relevant examples with comparisons to computer simulations to illustrate the performance of our system; in each example, the values as calculated using our analysis, are compared with the results of a simulation, given as 95% confidence intervals† . In all cases, the maximum error† remains well below 10%. The simulation program can be found in appendix C.

Examples are given for a fifty-station token ring, and for two and three priority levels at the stations. The extension of the conclusions to more than three classes is straightforward. Message transmission time is assumed to be exponentially distributed or deterministic, identical for all priority classes and with a mean as an integer multiple of the total propagation delay τ which is taken as one unit. For all the examples, mean message transmission time is chosen to be 100 (in units of τ) and the walktime is constant at $r = 1/N$ (in units of τ) for all stations. These values are reasonable for practical systems. For the cases with two priority levels, the arrival rate of class 1 messages at a station represents 50 percent of the total arrival rate at that station. For the cases with three priority levels at a station, each message class takes one third of the total arrival rate to that station.

Example 1 : Fifty stations symmetric system with two priority levels

$\lambda_i = \lambda \quad i = 1,2,...,50$

Fig.3.2 shows the simulated and approximate mean delay for each class for different values of the total throughput. The values obtained from the simulation are summarized by table 3.1. The decimal parts are truncated since they are

† For definition and calculation of confidence intervals and maximum error, see appendix B.

negligible.

Example 2 : Fifty stations asymmetric system with one heavily loaded station

$\lambda_1 = 4\lambda$ and $\lambda_i = \lambda$, $i = 2,..,50$.

The approximate mean delays and the simulation result are illustrated in Fig.3.3 as functions of the total thoughput. The simulation values for mean delays at stations 2 and 50 are also summarized by table 3.2.

Example 3 : Fifty stations asymmetric system with five heavily loaded stations

$\lambda_1 = \lambda_{11} = \lambda_{21} = \lambda_{31} = \lambda_{41} = 4\lambda$ and $\lambda_i = \lambda$, $i \neq 1,11,21,31,41$.

The approximate mean delays at station 1 are plotted in Fig.3.4a and for stations 2 and 50, the mean delays are illustrated in Fig.3.4b. Simulation values for mean delays in sets 1 and 2 are summarized by tables 3.3a and 3.3b respectively.

Example 4 : Fifty stations symmetric system with three priority levels

Fig.3.5 shows the approximate average delay for each class for different values of the total throughput.

Example 5 : Fifty stations, One heavily loaded station and three classes

The approximate mean delays for each class at stations 2 and 50 are plotted in Fig.3.6 as functions of the total throughput.

Throughput	Class 1 mean delay	Class 2 mean delay
.6	243 ± 11	247 ± 12
.7	320 ± 15	329 ± 17
.8	464 ± 25	401 ± 27
.9	840 ± 53	965 ± 72
.95	2120 ± 132	2515 ± 186

Table 3.1 : Simulation mean delays. Two-priority symmetric system.

Throughput	Class 1 mean delay	Class 2 mean delay
.6	248 ± 16	252 ± 21
.7	324 ± 22	335 ± 29
.8	488 ± 37	517 ± 46
.9	889 ± 74	1012 ± 83
.95	1685 ± 118	2207 ± 147

Table 3.2 : Simulation mean delays at station 2. One heavily loaded station.

Throughput	Class 1 mean delay	Class 2 mean delay
.5	202 ± 14	208 ± 17
.6	246 ± 18	260 ± 21
.7	335 ± 31	367 ± 36
.8	500 ± 38	596 ± 52
.9	953 ± 71	1280 ± 104
.94	1740 ± 132	2695 ± 218

Table 3.3a : Simulation mean delays in set 1. Five heavily loaded stations.

Throughput	Class 1 mean delay	Class 2 mean delay
.6	242 ± 18	245 ± 22
.7	321 ± 14	332 ± 19
.8	470 ± 32	518 ± 46
.9	826 ± 53	907 ± 72
.96	2189 ± 162	2460 ± 217

Table 3.3b : Simulation mean delays in set 2. Five heavily loaded stations.

From the above examples we make some interesting observations with respect to the average delays.

1) In all cases, a deterministic message transmission time provides less delay than an exponential one with the same mean.

2) Only in the symmetrical situation the stations have the same mean delay for each class.

3) In an asymmetric system, identical stations (stations with the same arrival rate, same transmission and walking times) do not necessarily have the same average delay. The latter depends on the distance of the stations relative to the heaviest loaded station i.e the order in which the stations are polled affects the performance. We notice that if stations $i-1$ and $i+1$ are statistically identical and station i has a higher arrival rate than these two stations, then the mean waiting time of each class at station $i+1$ will be slightly larger than at station $i-1$, if the polling order is $\ldots, i-1, i, i+1, \cdots$ Similarly, all stations that are polled directly after station i will have mean delays for each class that decrease as their "distance" from station i increases. This effect is emphasized by the second and fifth examples and illustrated in Fig.3.3 and Fig.3.6 respectively. We notice that if station i has the highest arrival rate and all other stations have the same arrival rate (smaller than that of station i), then each class mean delay decreases from station $i+1$ to $i-1$. The reason for this phenomena may be seen intuitively. Suppose a tagged message arrives at station $j (j \neq i)$ and has to wait. If station j is just the successor of station i in the polling order (i.e $j = i+1$), from the instant the tagged message arrived, *most likely* station i will be between the stations to be polled before station j; and since station i has the higest arrival rate, most probably it will hold the token for transmission. The situation changes with the position of station j in the polling order. The farther away station j is from station i in the polling order, the *less likely* station i will be between the stations to be polled before station j. Since the probability of the heaviest station to be polled before station j decreases, the waiting time of the tagged message at station j decreases consequently.

It is important to note that, in general, a message delay (from its arrival to its service), consists of a residual cycle, plus a number of complete cycles. The complete cycles are identical for all stations. However, the residual cycle is larger for the stations following the the most heavily loaded station in the polling order since "their" residual cycle is more likely to include the most heavily loaded station. This is the reason for the position dependency of the delay at a station in nonsymmetric systems.

In the nonsymmetrical cases, we notice also that lower priority classes appear more sensitive to the polling order. From the fifth example, we notice (at saturation 99% load) that the difference in mean delays for each class between station 2 and 50 is approximately 4 times a mean message transmission time for class 1, 8 times for class 2 and 19 times for class 3. This is also to be expected since a priority class blocking probability at a station decreases as its distance from the heaviest station increases. The third example further emphasizes all the above observations. In this example with 5 heavy stations, we can note some element of symmetry. In fact, stations 2, 12, 32, and 42 are identical in the sense that all of then are immediate successors of heavy stations with the same rate. Under this concept, we can identify the following ten sets of identical stations with the set number specified by its first element:

{1,11,21,31,41}, {2,12,22,32,42}, {3,13,23,33,43}, {4,14,24,34,44},
{5,15,25,35,45}, {6,16.26,36,46}, {7,17,27,37,47}, {8,18,28,38,48},
{9,19,29,39,49}, {10,20,30,40,50}.

Mean delays for each message class are identical for stations belonging to the same set.

3.5. CONCLUDING REMARKS

We have extended Ibe's result to asymmetric polling systems in which each station receives different priority classes of messages. The number of priority classes vary from one station to another, and at a station higher class messages are given nonpreemptive priority over lower class messages.

Several examples of asymmetric systems have been also considered. In these cases the approximate analysis is compared with simulation results. We also observe that each message class delay at a station is affected by the order in which that station is polled. A similar observation was made by G.B. Swartz [54] who noticed that although the average cycle time is independent of the station it is measured from, the variance of the cycle time, on the contrary, is dependent on the station and on the polling order. This second-order effect causes the delay to be position- dependent. At heavy loads, the model seems to underestimate high priority classes and overestimate low priority delays. This reflects the fact that at these loads, since queue lengths are no more stable over time, low priority messages are blocked more often than predicted for steady state network behavior, resulting in a reduction of load for high priorities.

We showed that operation of the priority mechanism is not free. There is a well-defined cost associated with its implementation. This cost is an easily computable function of various network parameters. The results obtained are very simple and the expressions "sympatiques"†. This is not always the case. The Gianini and Manfield (G&M) analysis[13] of the two-priority polling system is rather complex even under symmetry. In the next chapter, we compare our results with some others reported in the literature. The description and results of the G&M model are given and the various schemes of the prioritized versions of CSMA/CD[70] are presented.

† Sympatique is a french word. Its meaning here is pleasant, attractive to see and easy to understand.

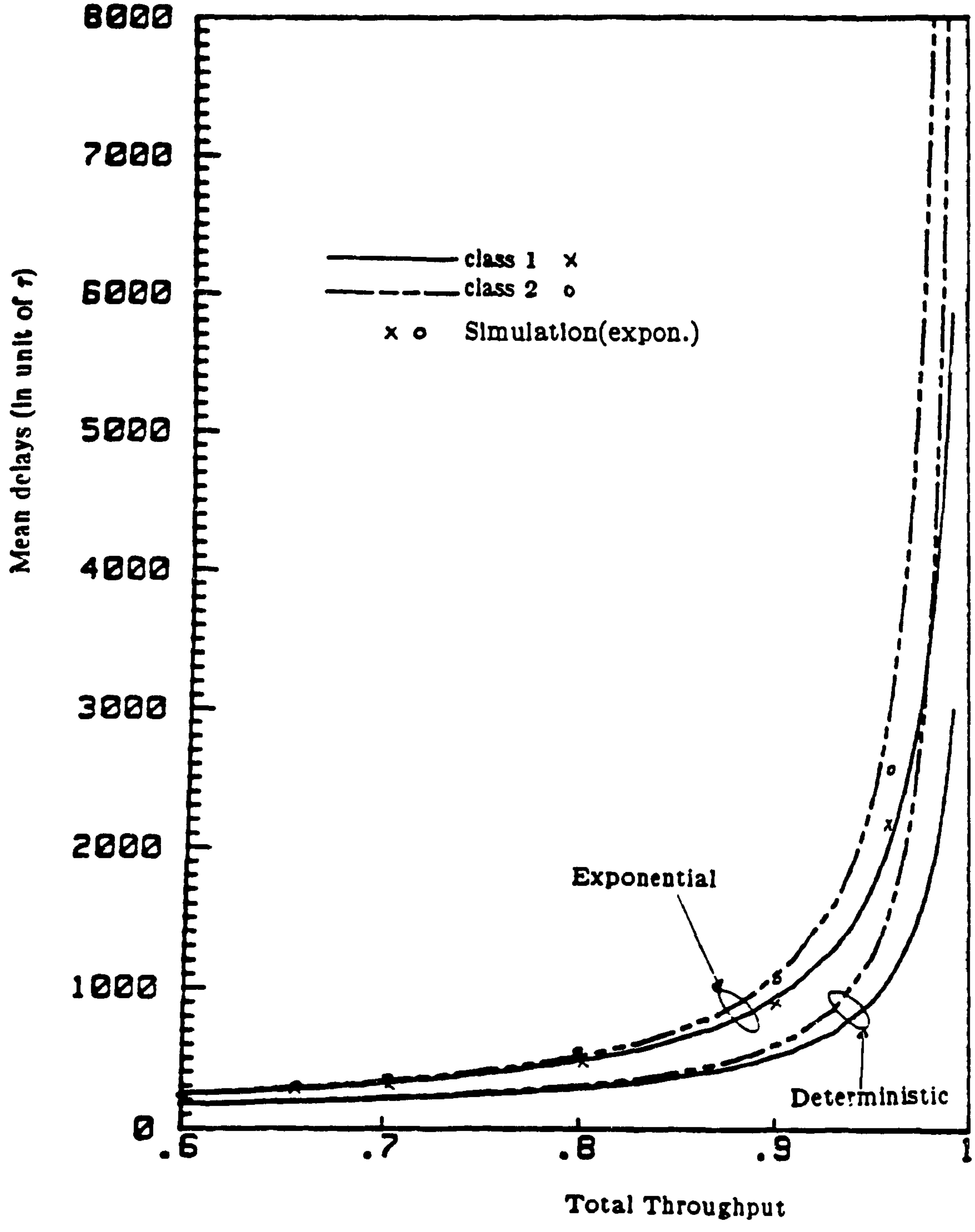


Fig.3.2: Mean delays as function of total throughput
Two-priority symmetric system

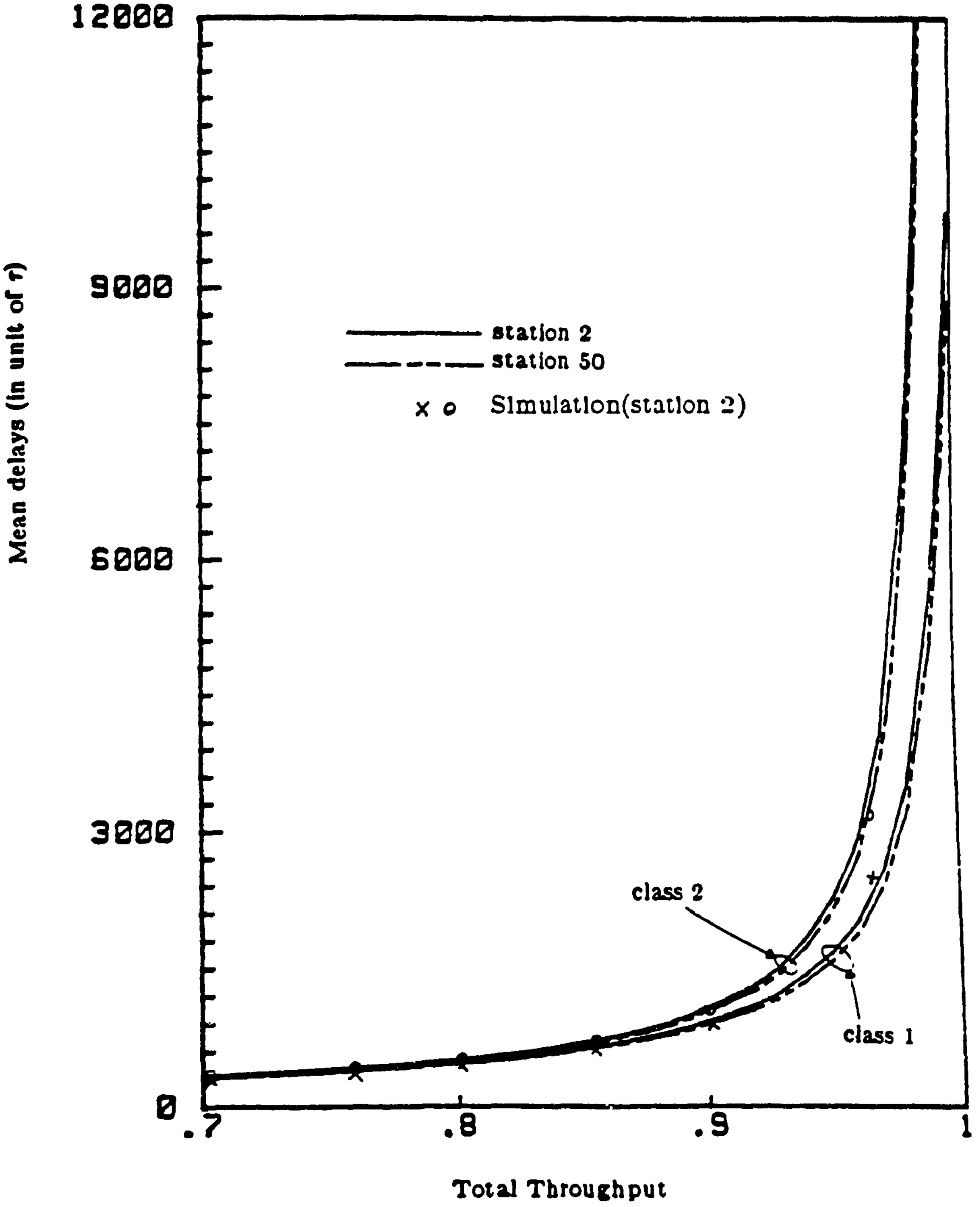


Fig.3.3: Mean delays at station 2 and 50 as function of total throughput
Two-priority One heavy station (station 1)
Exponential message transmission time

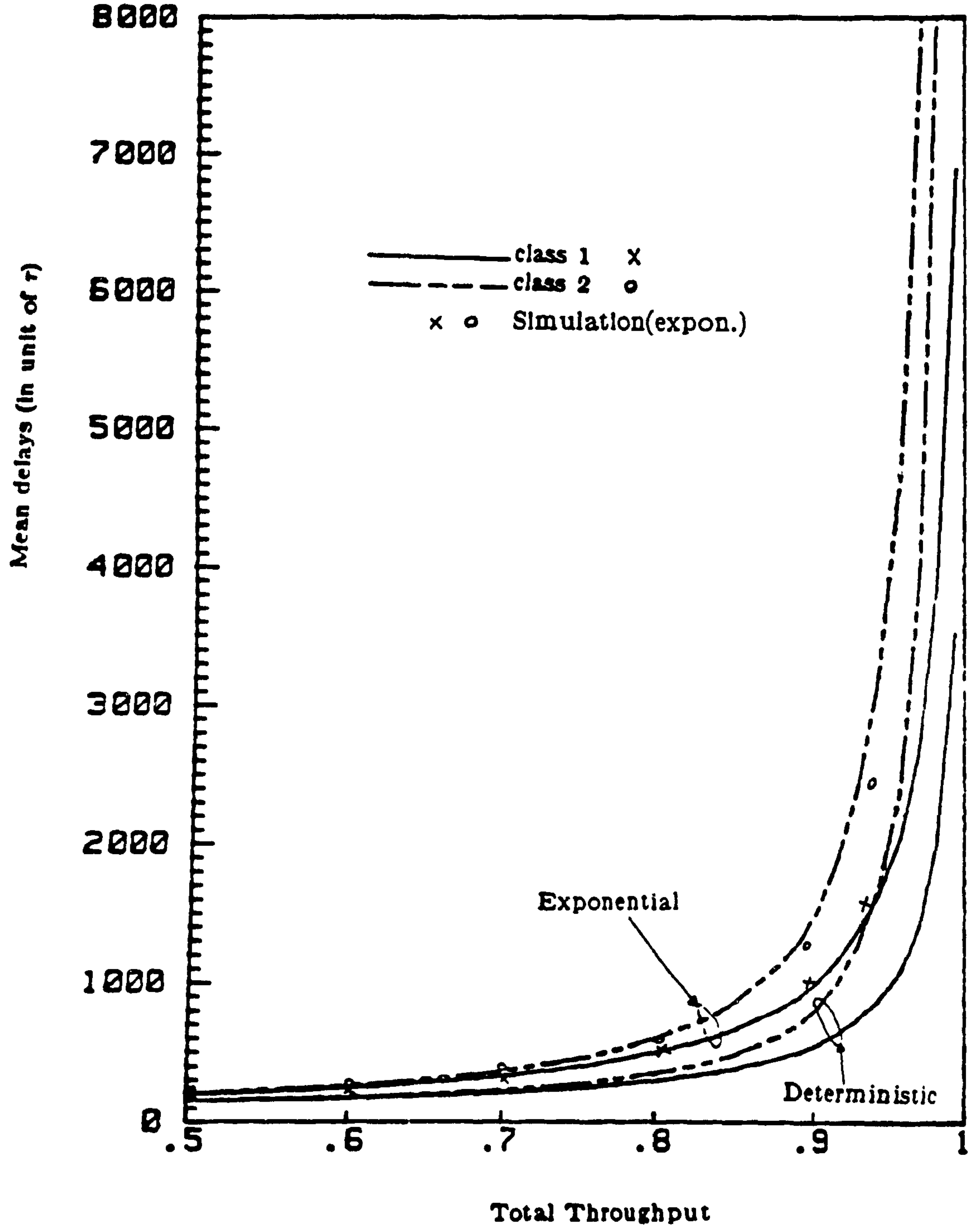


Fig.3.4a: Mean delays in set 1 as function of total throughput
Two-priority Five heavy stations (set 1)

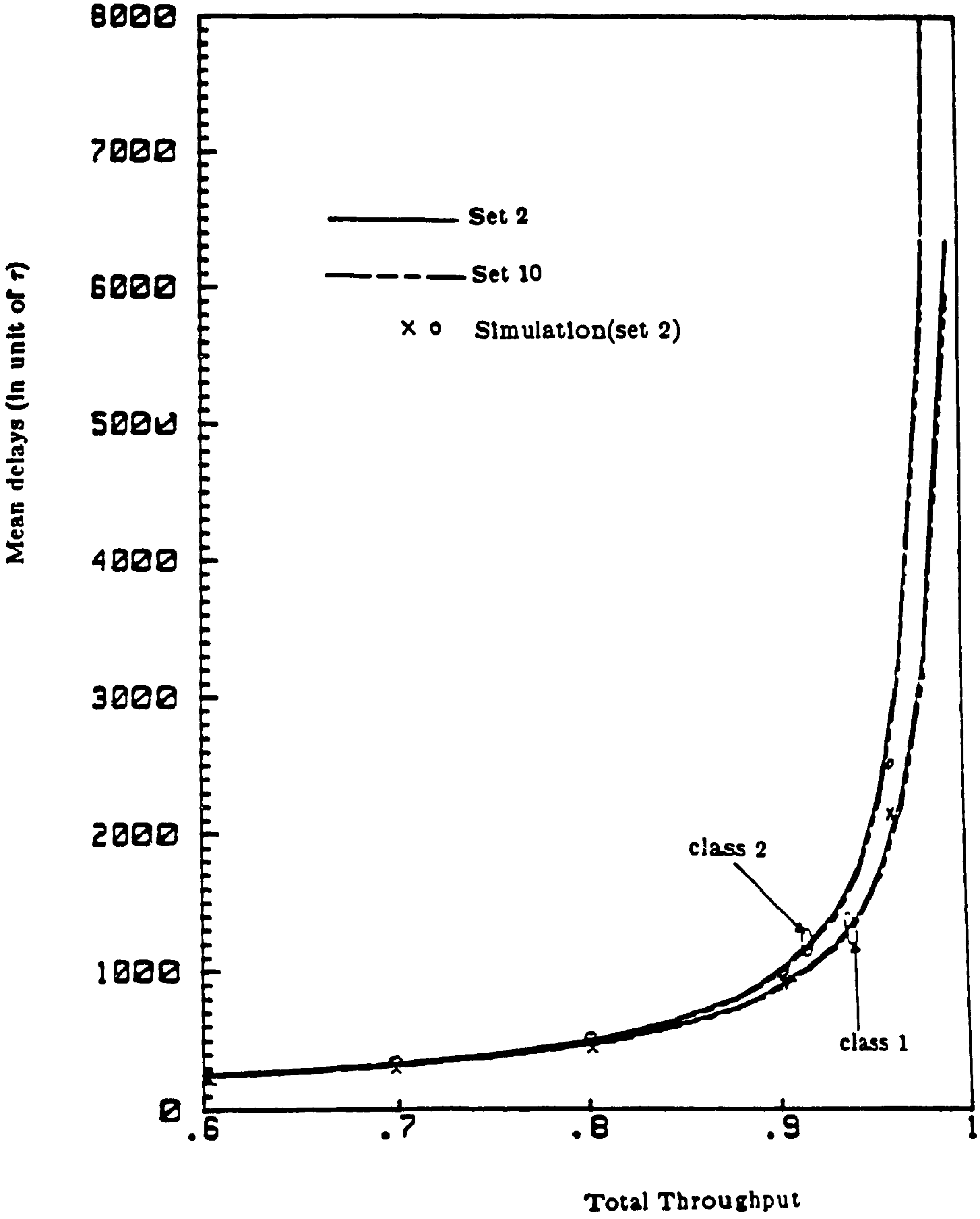


Fig.3.4b: Mean delays in sets 2 and 10 as function of total throughput
Two-priority Five heavy stations (set 1)
Exponential message transmission time

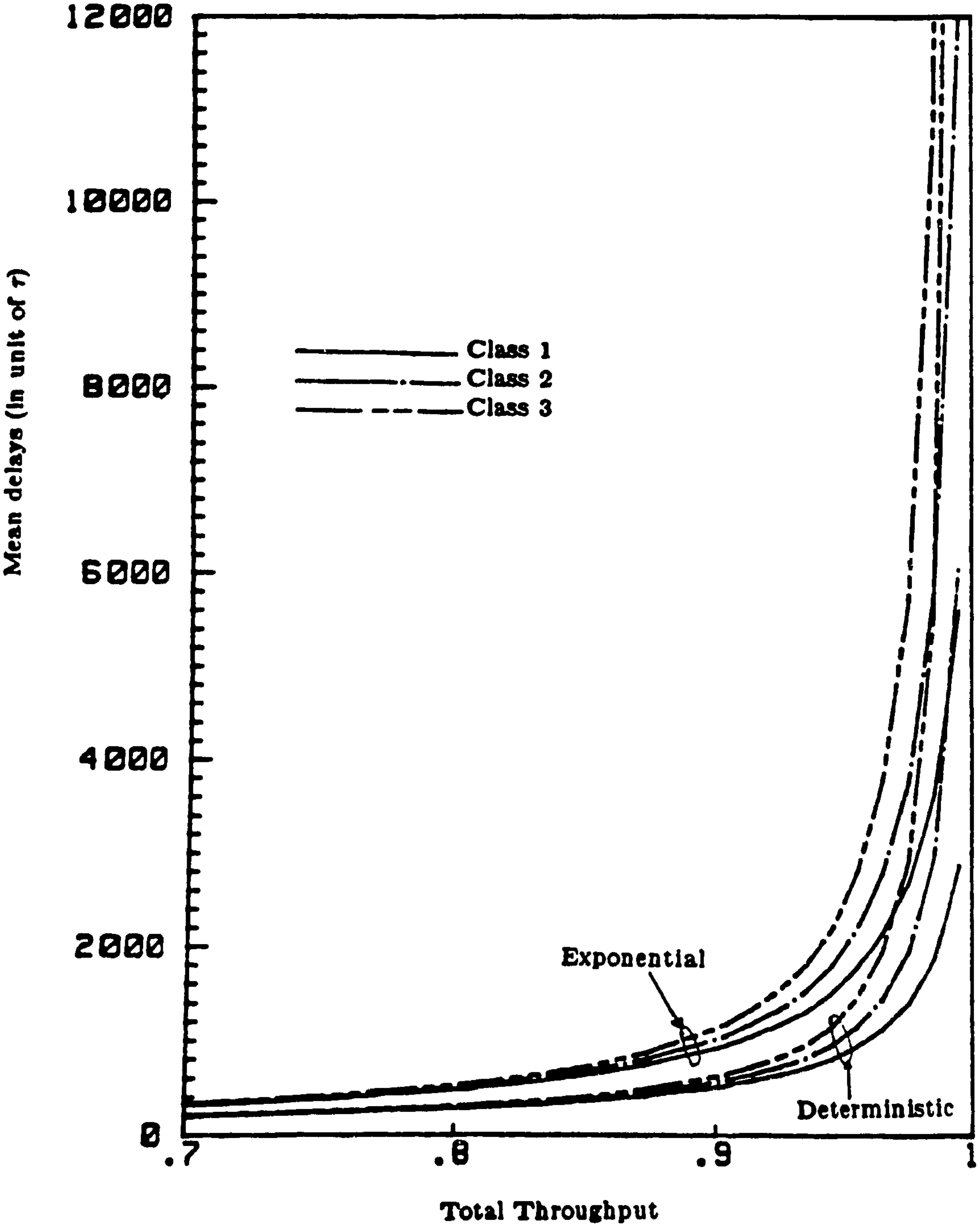


Fig.3.5: Mean delays as function of total throughput
Three-priority symmetric system

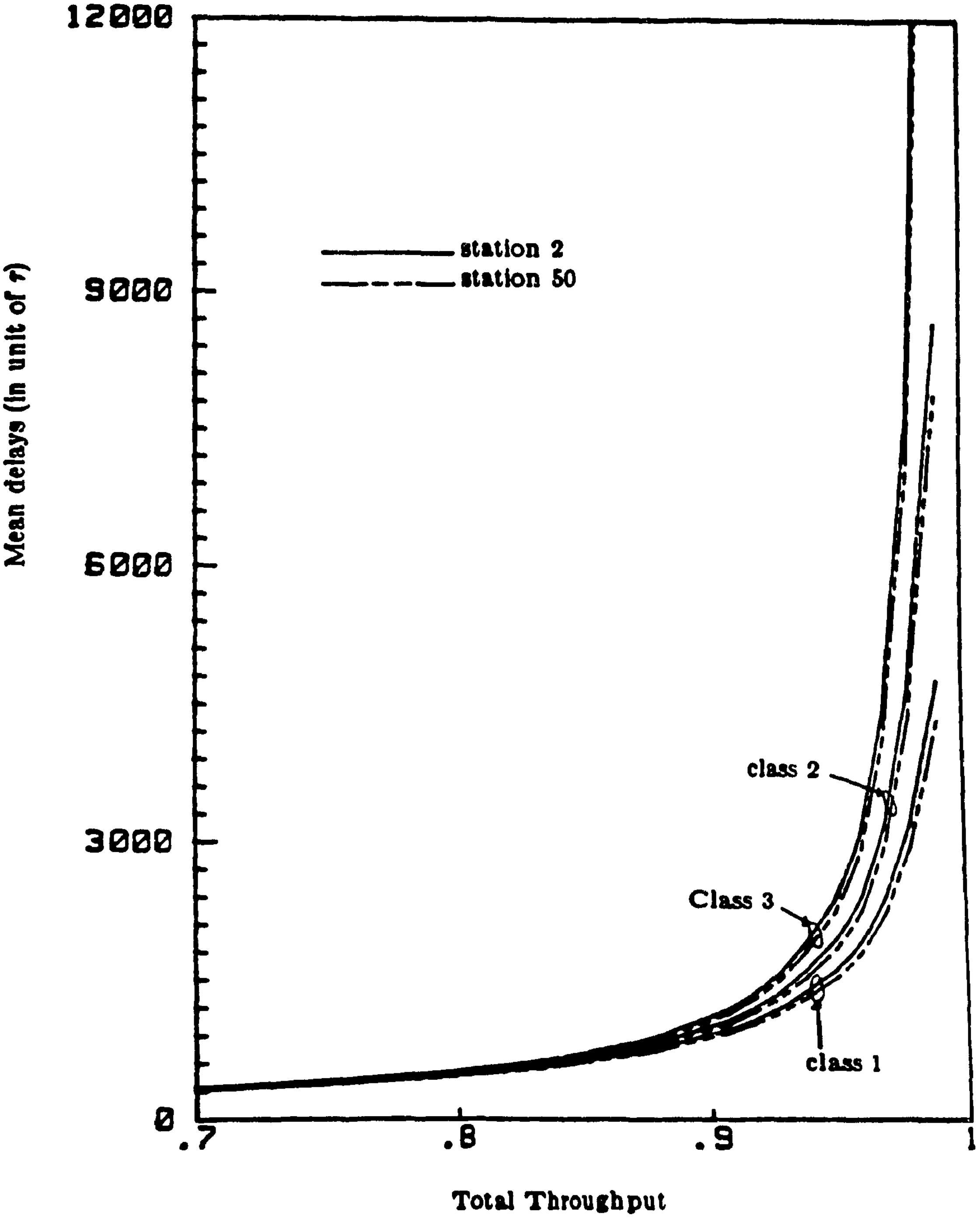


Fig.3.6: Mean delays as function of total throughput
Three-priority One heavy station (station 1)
Exponential message transmission time

CHAPTER IV

Priority Access Schemes in LANs: A Comparison

We have seen that the introduction of priority mechanism in LANs was motivated by different delay requirements among different types of messages which exist in data communication networks. Even though this diversity of messages in data communication is not new, the concept of priority structure among different types of messages does not have a long history in LANs. This explains why only a few prioritized schemes in LANs have been reported in the technical literature. In 1985, Choudhury and Rappaport [70] developed a technique to approximate prioritized versions of CSMA/CD, and have applied the technique to a number of nonpreemptive and preemptive schemes.

By the same time, D.R. Manfield [71] presented the queueing analysis of the transmission delays for the two-way data traffic between a central processor and a set of peripheral controllers, where the communication is based on a complex polling cycle giving effective priority to outgoing messages from the central control. The incoming (low-priority) queues are served non-exhaustively while the outgoing (high-priority) queues are served exhaustively.

More recently, Gianini and Manfield studied the two-priority model[13] where a station can be polled at high or at low priority: a low-priority poll always begins with a low-priority service and occurs only if there are no high-priority messages present anywhere in the system at the polling instant; otherwise, the station is polled at high priority, and only high-priority messages are transmitted. In Gianini and Mansfied model[13] the decision of transmitting a message at a station is based on the priority level of the whole system. Even though our model is much fairer for stations and classes, their model (even not physically realizable) is closer to the one of IEEE standard than ours in which the decision of transmitting a message is localized at the station.

In this chapter, we compare the results obtained from our model (developed in the previous chapter) with some of the results reported in [13] and [70]. The delay-throughput characteristic of the schemes and the system behavior at load saturation point, constitute the two major performance aspects of primary interest.

The layout of the chapter is as follows. The description and results of Gianini and Manfield's two-priority model are given in section 4.1. Some of their results reported in [13] are compared with ours. The prioritized versions of CSMA/CD are presented in section 4.2 and the results compared with those obtained from the prioritized token passing model we have analyzed.

4.1 GIANINI and MANFIELD's TWO-PRIORITY MODEL [13]

In [13] Gianini and Manfield (G&M) consider a multiqueue system with N stations. Each station transmits variable-length messages at two priority levels. The convention in [13] is that the smaller the index associated with a class is, the lower is the priority of that class. Each station has two queues: a low-priority queue (priority 1) and a high priority queue (priority 2). Arrivals at station i, priority j $(i=1,2,..,N, j=1,2)$ are independent Poisson processes with parameters λ_{ij}; message transmission times are arbitrarily distributed random variables with LST $M_{ij}(s)$. The overhead time needed to switch from station i to station $i+1$ is modelled by a random variable with LST $R_i(s)$. A station can be polled at high or at low priority. A low-priority poll always begins with a low-priority transmission, and occurs only if there are no high-priority messages present anywhere in the ring at the polling instant; otherwise, the station is polled at high priority, and only high-priority messages are transmitted. The priority mechanism of the exhaustive system is described in Figure 4.1.

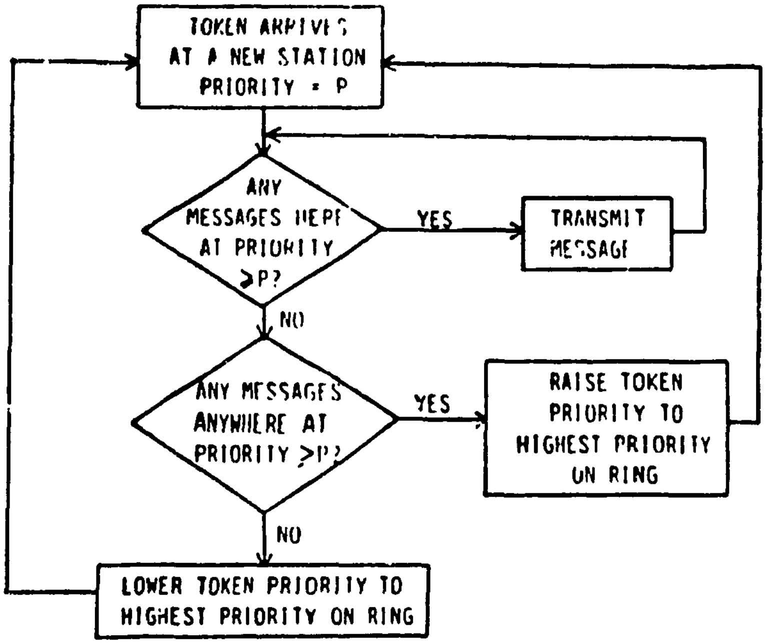

TOKEN ARRIVES AT A NEW STATION PRIORITY = P
ANY MESSAGES HERE AT PRIORITY >P?
YES
TRANSMIT MESSAGE
NO
ANY MESSAGES ANYWHERE AT PRIORITY >P?
YES
RAISE TOKEN PRIORITY TO HIGHEST PRIORITY ON RING
NO
LOWER TOKEN PRIORITY TO HIGHEST PRIORITY ON RING

Fig. 4.1: Priority Mechanism of the Exhaustive Service System[13].

As in our case and in cyclic queues in general, the mean delay analysis of the two-priority model is related to the study of cycle times. However, in the two-priority model, since a station is polled at low priority only if there are no high-priority messages enqueued anywhere on the ring at the polling instant, the time between low-priority polls at a given station, a *supercycle time* is defined. The analysis of the mean delay for low-priority messages is based on characteristics of the supercycle time much in the same way that mean delay analysis in the single-priority model is based on a study of the cycle time. Gianini and Manfield considered separately the exhaustive and the gated service disciplines. For each of these service disciplines, an approximation to the mean delay for both types of priority messages at each station is found. Obtaining the results requires long and tedious algebraic manipulations. Our objective is not to derive these results; however, even the expressions are strongly dependent, we give here the main results obtained in the case of exhaustive service discipline. The same approach was followed for the case of gated service.

We define

$$\rho_{ij} = \lambda_{ij} m_{ij}, \quad i=1,...,N, \quad j=1,2$$

$$\rho_i = \rho_{i1} + \rho_{i2}, \quad i=1,...,N$$

$$\rho = \sum_{i=1}^{N} \rho_i$$

During a high-priority poll, transmission continues at station i until the high-priority queue is empty. The amount of service generated by a single high-priority message enqueued at the polling instant in the high-priority queue of station i is a busy period B_i with LST given by

$$B_i(s) = M_{i2}\left(s + \lambda_{i2} - \lambda_{i2} B_i(s)\right) \tag{4.1}$$

The first two moments of B_i are given by

$$E(B_i) = \frac{m_{i2}}{1 - \rho_{i2}}, \quad E(B_i{}^2) = \frac{m_{i2}{}^{(2)}}{(1 - \rho_{i2})^3} \tag{4.2}$$

The duration of a poll at high priority is the sum of as many busy periods, B_i, as there are high-priority messages enqueued at station i at the polling instant.

During a low-priority poll, however, transmission continues until both priority queues are empty. The service discipline in each queue is FIFO, but high-priority messages have nonpreemptive priority over low-priority messages. The amount of service generated by a single low-priority message enqueued at station

i at a low-priority polling instant is a modified busy period, which is initiated by a low-priority message, but in which all arriving messages at either of the two priority classes are transmitted until both priority queues are empty. Let $\hat{B}_i$ be the length of this modified busy period and S_i the length of an extended service at low priority. We may relate the LST of the modified busy period ($\hat{B}_i(s)$) to the LST of the extended service at low priority ($S_i(s)$) by

$$\hat{B}_i(s) = S_i\left(s + \lambda_{i1} - \lambda_{i1}\hat{B}_i(s)\right) \tag{4.3}$$

with

$$S_i(s) = M_{i1}\left(s + \lambda_{i1} - \lambda_{i2}B_i(s)\right) \tag{4.4}$$

where $B_i(s)$ is as defined in (4.1); the mean and second moment of $\hat{B}_i$ can be obtained as follows:

$$E(\hat{B}_i) = \frac{m_{i1}}{1-\rho_i}, \quad E(\hat{B}_i^{\,2}) = \frac{(1-\rho_{i2})m_{i1}^{(2)}}{(1-\rho_i)^3} + \frac{\lambda_{i2}m_{i1}m_{i2}^{(2)}}{(1-\rho_i)^3} \tag{4.5}$$

The duration of a low-priority poll is the sum of as many modified busy periods as there are low-priority messages enqueued in the station i at the low-priority polling instant of that station.

It is important to notice that for the gated service discipline, only the high-priority messages already enqueued in the station i at a polling instant are transmitted during a high-priority poll, whereas during a low-priority poll, only the low-priority messages already enqueued in station i at the polling instant are transmitted. The amount of work generated by a single high-priority message during a high-priority poll has thus a LST simply given by $M_{i2}(s)$, and the work generated by a single low-priority message during a low-priority poll has a LST given by $M_{i1}(s)$

To find an approximation to the mean delay, first the joint probability generating function of all queue lengths at polling instants is determined. The first two moments of the lengths of both priority queues at station i are then used to calculate the mean delay of messages in station i.

Let $L_{ij}(t)$ denote the number of class j ($j = 1,2$) messages at station i at time t and define the joint and marginal generating function F_i for [$L_{ij}(t), i=1,...,N; \; j=1,2$] at time $t=\tau_i(m)$ (where m is sufficiently large for steady-state to have been achieved), i.e. the time when station i is polled. The equation governing the two-priority model may be obtained by relating F_i to F_{i+1}. We have

$$F_{i+1}(z_1,..,z_N, z_1', ..., z_N')$$

$$= R_i \left[\sum_{j=1}^{N} \left(\lambda_{j2}(1 - z_j) + \lambda_{j1}(1 - z_j') \right) \right]$$

$$\cdot \left[F_i \left(z_1,...,z_{i-1}, B_i \left(\sum_{j \neq i} \lambda_{j2}(1 - z_j) + \sum_{j=1}^{N} \lambda_{j1}(1 - z_j') \right), z_{i+1},...,z_N, z_1' ..., z_N' \right) \right.$$

$$- F_i(0,...,0, z_1' ..., z_N')$$

$$\left. + F_i \left(0,...,0, z_1' ..., z_{i-1}' , \hat{B}_i \left(\sum_{j \neq i} \lambda_{j2}(1 - z_j) + \sum_{j=1}^{N} \lambda_{j1}(1 - z_j') \right), z_{i+1}' ..., z_N' \right) \right] \quad (4.6)$$

We define

$$g_i^{(0)}(j,k) = \left. \frac{\partial^2 F_i(0,..,0, z_1' ,..., z_N')}{\partial z_j' \, \partial z_k'} \right|_{z_1' = \cdots = z_N' = 1} \quad (4.7a)$$

$$f_i(j,k) = \left. \frac{\partial^2 F_i(z_1..,z_N, z_1' ,..., z_N')}{\partial z_j \, \partial z_k} \right|_{z_1 = \cdots = z_N = z_1' = \cdots = z_N' = 1} \quad (4.7b)$$

and

$$g_i(j) = \frac{\partial F_i(z_1..,z_N,z_1' ,...,z_N')}{\partial z_j'}\Bigg|_{z_1= \cdots =z_N=z_1' = \cdots =z_N' =1} \tag{4.7c}$$

To find an expression for the mean, the queueing system in station i is modelled as an M/G/1 with server vacation time. It is proven by G&M that in the symmetrical case, the mean delay for class j messages at station i $(E(D_{ij}))$ can be expressed as a function of $g_i^{(0)}(i,i)$ and $f_i(i,i)$. For low-priority messages, we have

$$E(D_{i1}) = \frac{1-\rho_{i2}}{1-\rho_i}\frac{g_i^{(0)}(i,i)}{2C\lambda_{i2}^2} + \frac{\lambda_{i1}m_{i1}^{(2)}}{2(1-\rho_i)(1-\rho_{i2})} + \frac{\lambda_{i2}\rho_{i1}m_{i2}^{(2)}}{2(1-\rho_i)(1-\rho_{i2})^2} \tag{4.8}$$

and for high-priority messages, we have

$$E(D_{i2}) = \frac{1}{1-\rho_i}\frac{f_i(i,i)}{2C\lambda_{i2}^2} + \frac{\rho_{i1}}{1-\rho_{i2}}\frac{m_{i1}^{(2)}}{2m_{i1}} + \frac{\lambda_{i2}m_{i2}^{(2)}}{2(1-\rho_{i2})} \tag{4.9}$$

where

C is the expected cycle length given by

$$C = \frac{\sum_{i=1}^{N} r_i}{1-\rho}, \tag{4.10}$$

$$g_i^{(0)}(i,i) = \frac{\lambda^2{}_{i1}(1-\rho_i)^2}{(1-\rho_{i2})(1-\rho)}[(1-N\rho_{i2})(1-\rho_i) + (N-1)\rho_{i1}\rho_{i2}]^{-1}$$

$$\Bigg[Nr_i^{(2)} + (N-1)r_i C - 2Nr_i(1-N\rho_{i2})(1-\rho_i)C + \frac{2Nr_i(1-N\rho_{i2})}{\lambda_{i1}}g_i(i)$$

$$+ N\lambda_{i2}(1-\rho_{i2})^2(1-\rho_i)CE(B_i^2)$$

$$+ (N - 1)\lambda_{i1}[1 - \rho_{i2} - \rho_i(1 - N\rho_{i2})]\frac{1 - \rho_i}{1 - \rho_{i2}}CE(B_i^{\,2})\Bigg] \qquad (4.11)$$

and

$$f_i(i,i) = \lambda_{i2}\frac{1 - \rho_{i2}}{1 - N\rho_{i2}}\Bigg[Nr_i^{(2)} + (N - 1)r_i\,C + (N - 1)\lambda_{i2}(1 - \rho_{i2})(1 - \rho_i)CE(B^2{}_i)$$

$$+ (N - 1)\frac{m_{i1}^{(2)}}{(1 - \rho_i)^2}(1 - \rho_{i2})g_i^{(0)}(i,i)\Bigg] \qquad (4.12)$$

The computation of mean delay experienced by each message class would have been exact if $g_i(i)$, the expected number of low-priority messages at an arbitrary polling instant of station i was known exactly. Unfortunately, the equations Gianini and Manfield obtained do not suffice to determine this quantity. Based on regenerative points, an approximate expression for $g_i(i)$ is obtained, making their results approximate.

Results

Here, we present some numerical examples for a system with two priority levels. In each example, the delay-throughput characteristics for each message class, as calculated under our model are compared with the results in [13]. The first example(Fig.4.2) is for a system consisting of 8 stations; the walktimes are constant and equal to unity; message transmission times are exponentially distributed with mean at both priorities equal to 10. The low-priority utilization is equal to 40% of the total utilization; and the service disciplines are exhaustive in G&M model and limited in ours. The second example(Fig.4.3) is for the gated service discipline in G&M model. All parameters, as well as the distributions of walktimes and of message transmission times, are as in the first example. For the third example(Fig.4.4), we consider a system of 20 stations with the same parameters as in the previous two examples. Both exhaustive and gated service disciplines in G&M model are considered.

From these results, the following conclusions can be deduced.

1) In all cases, high priority messages perform better under G&M model compar:d to the limited service of our model.

2) At low load, low priority messages in G&M model(exhaustive or gated service) perform better than all classes under limited service. The ranges in which this phenomenon is observed appear larger under exhaustive service than under the gated one.

From Fig.4.2 where exhaustive service is considered, we can see that low priority messages in G&M model show lower delay than class 1 and class 2 messages in limited service for load less than .5 and .6 respectively; these ranges are reduced when the service is gated (see Fig.4.3). In fact, at low traffic level, a priority mechanism does not show a significant effect and the discrimination between classes is not significant. The time between low-priority polls at a given station, a supercycle time, is not so different from a normal cycle. The system behaves just like a normal token ring with exhaustive, gated or limited service discipline and it is well known that the exhaustive and gated services perform better than the limited one.

3) The limited service discipline (under our model) exhibits lower discrimination among priorities than exhaustive or gated service disciplines (under G&M model). This is emphasized in Fig.4.4. Because of the large number of stations, combined with the long walktimes (the time between polls of station i when no message is sent during a cycle is equal to 20, or two times the mean message transmission time), the low-priority delays in exhaustive or gated services are extremely large, even at relatively low utilizations. As mentioned in [13], the approximation appears quite accurate in all cases except for low-priority messages at 80% utilization; however, at that point, the low-priority delay is so huge

(approximately 200 times the mean message transmission time) that it would be unacceptable in some applications. Under the limited service and at the same utilization(80%), low-priority delay is only 20 times the mean message transmission time.

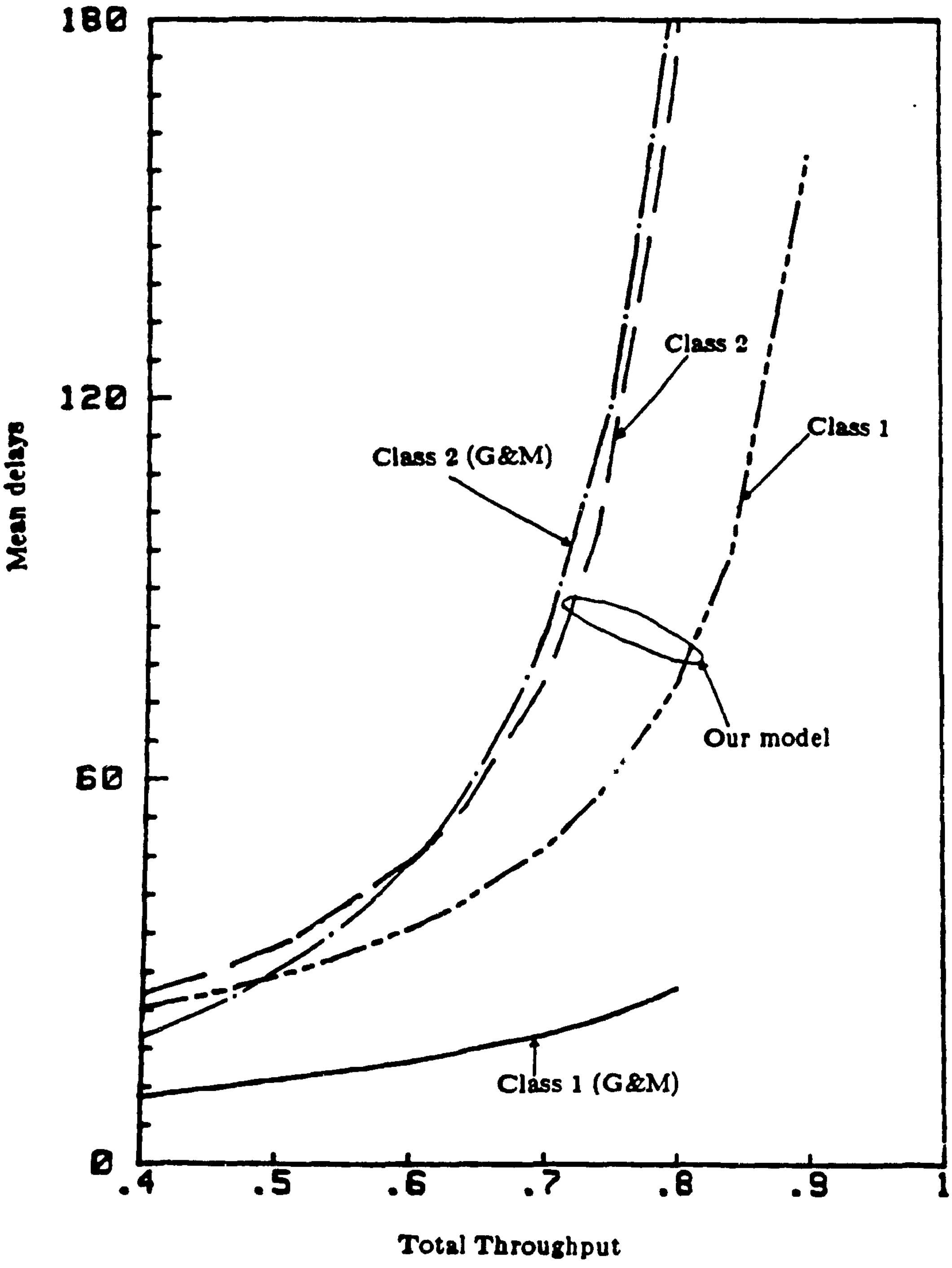


Fig.4.2: Delay comparisons for our model and G&M under Exhaustive service.
Two priority 8-station symmetric systems
Exponential message transmission time

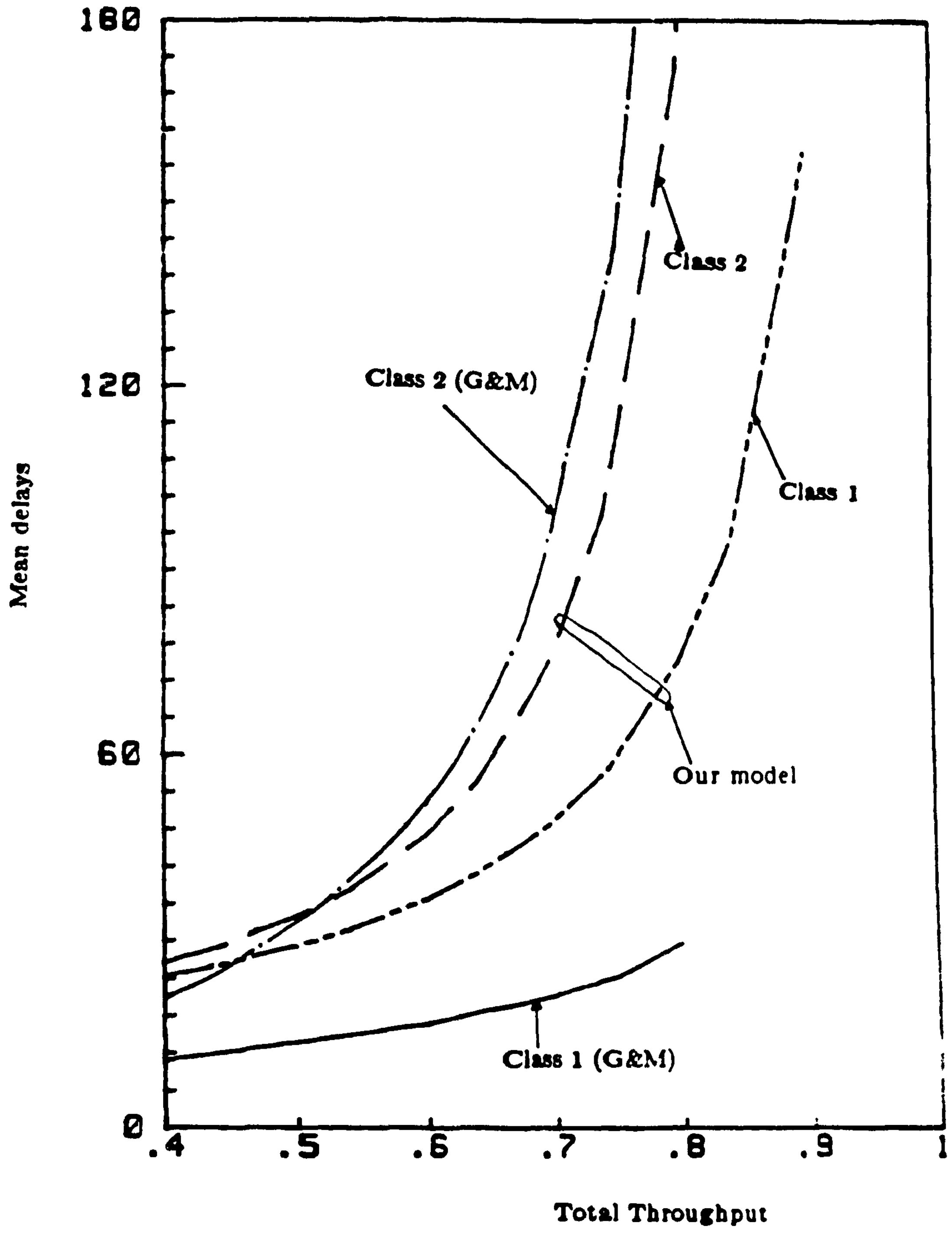


Fig.4.3: Delay comparisons for our model and G&M under Gated service
Two priority 8-station symmetric systems
Exponential message transmission time

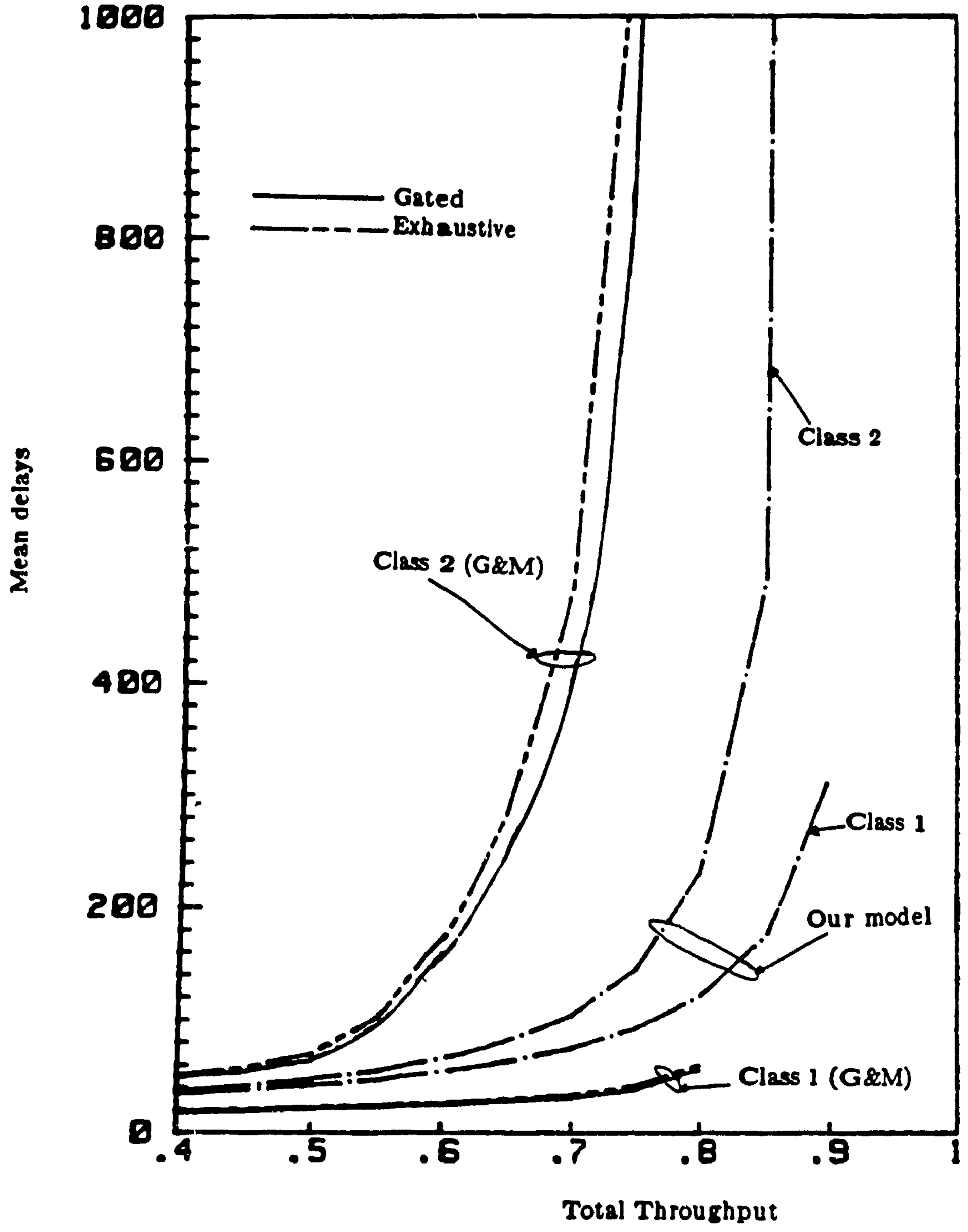


Fig.4.4: Delay comparisons for our model and G&M.
Two priority 20-station symmetric systems
Exponential message transmission time

4.2 PRIORITY ACCESS SCHEMES USING CSMA/CD [70]

In [70], an approximate technique for analyzing prioritized versions of CSMA/CD is developed and applied to a number of nonpeemptive and preemptive schemes. The priority mechanism is introduced during bus access, and at the same time a general retransmission strategy is used and the retransmission delay is a function of priority as well as the number of previous attempts. An arbitrary number of priority levels g is assumed, with the convention that a smaller priority index indicates higher priority. The channel is slotted in time with slot duration (τ) equal to the maximum propagation delay between any two stations. Messages (with transmission time assumed to be generally distributed) arriving during a slot time wait until the start of the next slot at which all ready stations simultaneously attempt transmission. The stability of the system depends crucially on the average message rescheduling delay d_{ij}, for the ith priority after j unsuccessful transmission attempts. The system exhibits a stable operating point and only the average delay for each class at this point is considered. For purposes of analysis, the rescheduling delays are assumed arbitrarily large; hence, the arrivals of new and retransmitted messages can be assumed to be independent and to form a Poisson process.

For simulation purposes in [70], the new message arrival process to the channel is assumed to be Poisson. The retransmitted messages are generated according to the rescheduling algorithm being simulated. A finite population of stations is simulated. New messages are considered to be generated only by those stations not having a message outstanding.

Three schemes have been analyzed. These are:

Scheme 1: Nonpreemptive Priority. All messages transmitted in One-persistent mode.

In this scheme, all messages are transmitted in One-Persistent Mode. A station seeking access to the channel and finding it busy waits one slot and checks again. Upon finding the channel idle, the station transmits an intialization signal of length $g-i+1$, where i is the priority of its outstanding message. At the end of the initialization, it determines by sensing whether or not there is a collision. If so, the station restarts the algorithm, following a random backoff delay of average d_{ij}, where j is a counter incremented by one each time that the station incurs a backoff delay in its current access attempt; if not, it transmits its message.

Scheme 2: Nonpreemptive Priority. Higher priorities transmit in One-persistent mode, lower priorities in nonpersistent mode.

Fix m. A station with outstanding message of priority $i \leq m$ is one-persistent. It operates exactly as in scheme 1 except that when finding the channel busy, the station waits the backoff delay of d_{ij} slots rather than one slot, before trying again; as in scheme 1, j is incremented by one each time the station incurs a backoff delay during its current access attempt.

Scheme 3: Preemptive Repeat Priority

As in the previous two schemes, a message transmission is permitted only upon successful completion of an initialization period of length $g-i-1$. This scheme differs from the others in the following respects: 1) the channel is sensed busy by a station with outstanding message of priority i only if an initialization signal, or a message of higher priority than i, is in progress; 2) a backoff delay d_{ij} is incurred each time in the current access attempt that the channel is sensed busy or a collision detected; 3) the channel is monitored during message transmission, and the transmission aborted at the onset of a collision. Low-priority transmissions are thus interrupted by high-priority initialization signals and a preempted message is entirely retransmitted.

Results

The delay-throughput characteristics for typical prioritized CSMA/CD and token passing (our model) are shown in Fig.4.5-8. Again, in all cases, a deterministic message transmission time with the same mean provides less delay than an exponential one.

From Figure 4.5, we can notice that in token ring, class 2 message in the deterministic case performs better than class 1 with exponential message transmission time for a very wide range of throughput. CSMA/CD exhibits an identical behavior, but only at low loads.

Figure 4.6 considers the effect of message length variation among priorites. The shorter messages are given higher priority. In the prioritized CSMA/CD, we observe that the higher priority messages experience low delay at low throughput, but the delay increases significantly at higher throughput. This is due to the fact that shorter message transmission times increase the ratio of propagation delay and message transmission time, thereby reducing the channel utilization. We should note also that once a lower priority message gets access to the channel, the short higher piority messages experience a long wait, due to the nonpreemptive nature of the priority mechanism. As in the cases without priority structure, the token ring is less sensitive to the message length variation than CSMA/CD.

Fig.4.7 shows the delay-throughput characteristics of a three-prioritized CSMA/CD and token ring. As expected, delay for the highest priority remains relatively small even for high throughputs. However, there is rapidly increasing discrimination among the priorities as throughput increases in CSMA/CD. For all these throughputs, mean delay for all classes is lower for token ring than even the highest priority for CSMA/CD.

Fig.4.8 is a comparison of the delay-throughput charateristics for schemes 1, 2, and 3 using a typical two-priority CSMA/CD and for a typical two-priority token ring. In CSMA/CD, the difference in delays among priorities is minimum for scheme 1, intermediate for scheme 2, and maximum for scheme 3. However, the prioritized token ring under limited service discipline shows a discrimination between classes which is lower than in the three schemes. In scheme 1, both the priorities are one-persistent and the only difference among the priorities is in the length of the initialization signal during access. In scheme 2, on the other hand, the lower priorities are further penalized by using nonpersistent access. In scheme 3, also, the lower priority is additionally penalized because of preemption.

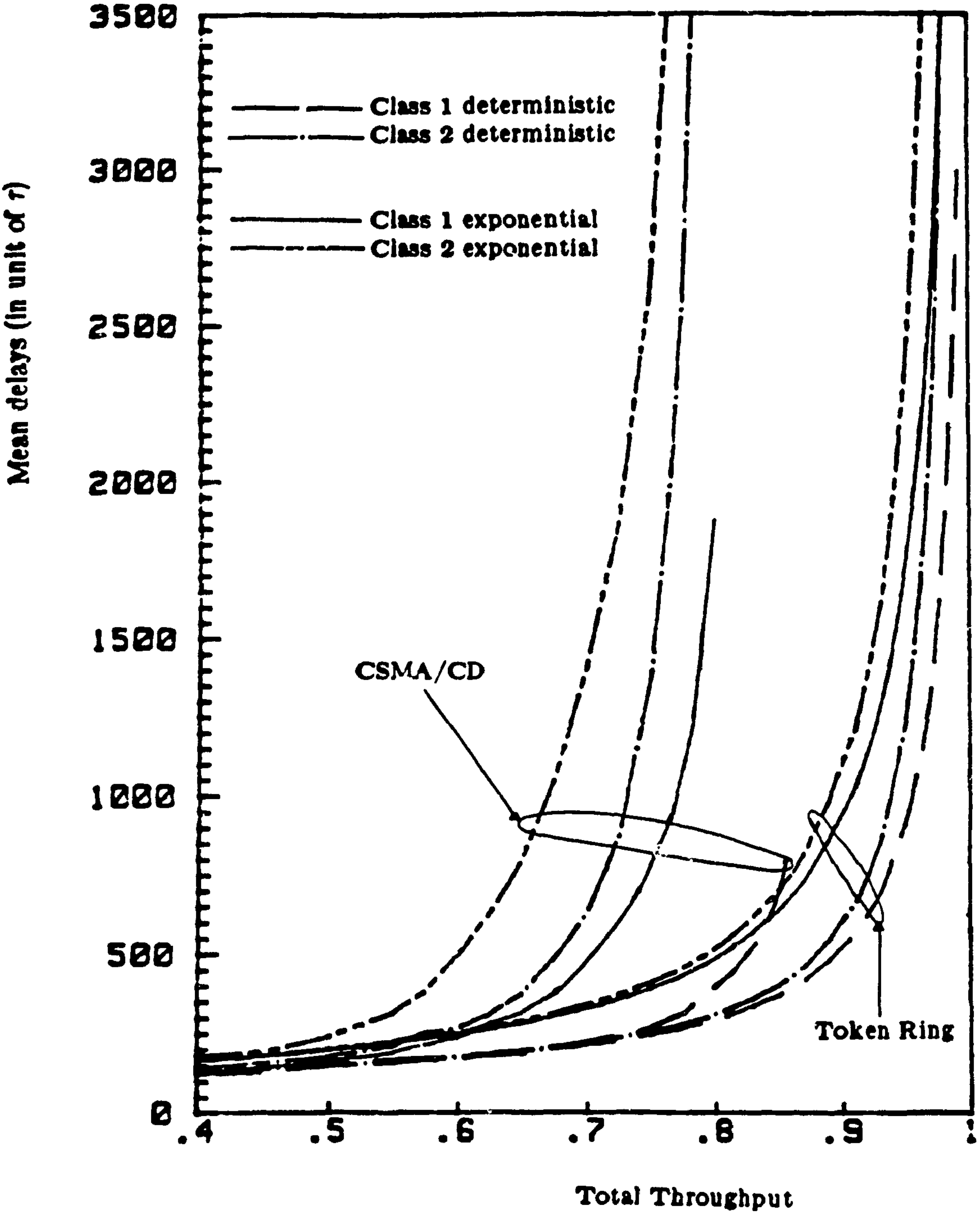


Fig.4.5: Delay comparisons for Two-priority Token Ring and CSMA/CD
Mean message transmission time: 100(in unit of τ)
Scheme 1 CSMA/CD. $d_{ij}=2^j\,50,\ 1 \leq j \leq 7,\ d_{ij}=6400, j>7$.

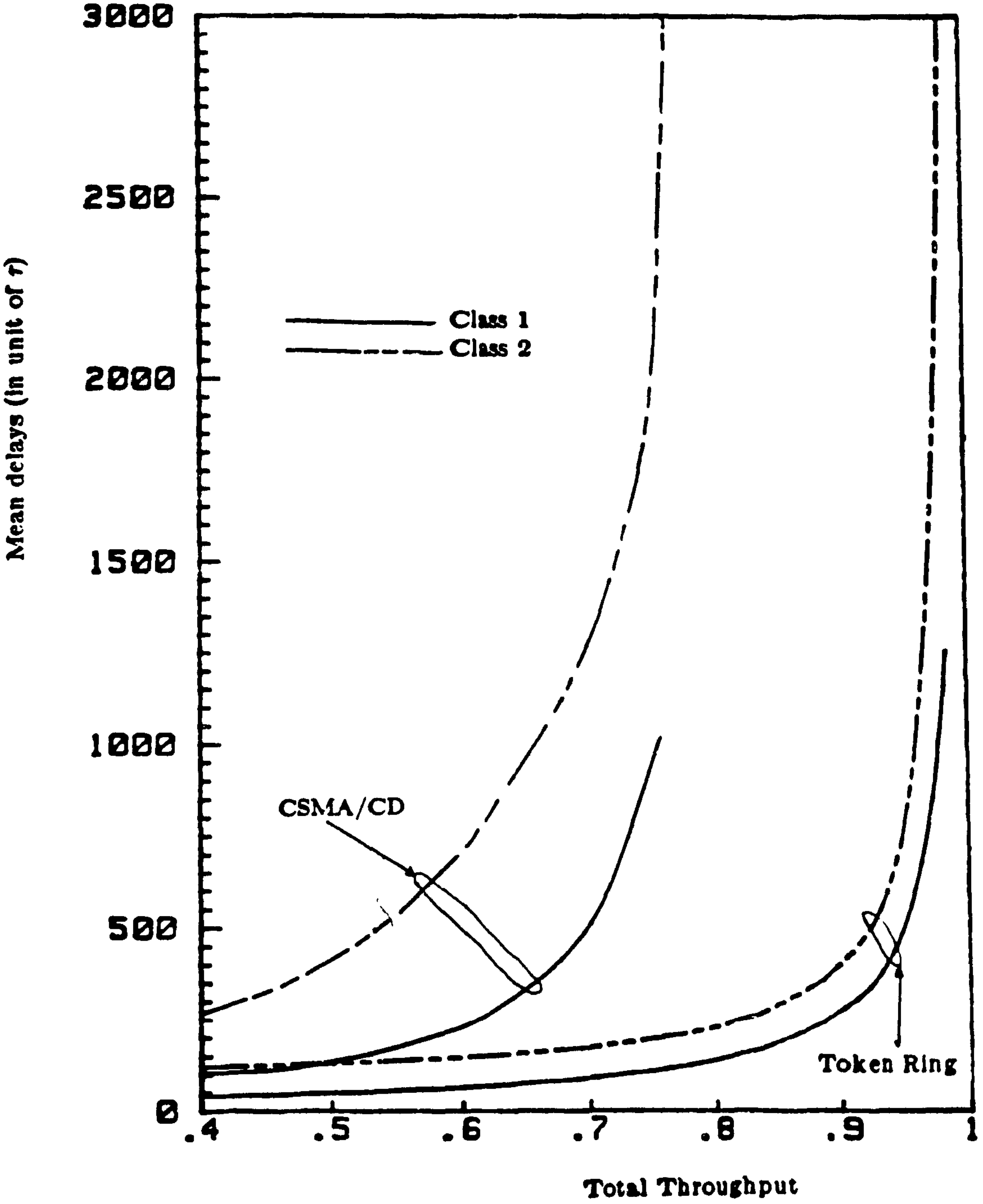


Fig.4.6: Effect of message length variation among priorities for Two-priority Token Ring and CSMA/CD.
Deterministic message transmission times.
Class 1 message mean transmission time: 20
Class 2 message mean transmission time: 100
Scheme 1 CSMA/CD. $d_{1j}=200$, $d_{2j}=500$

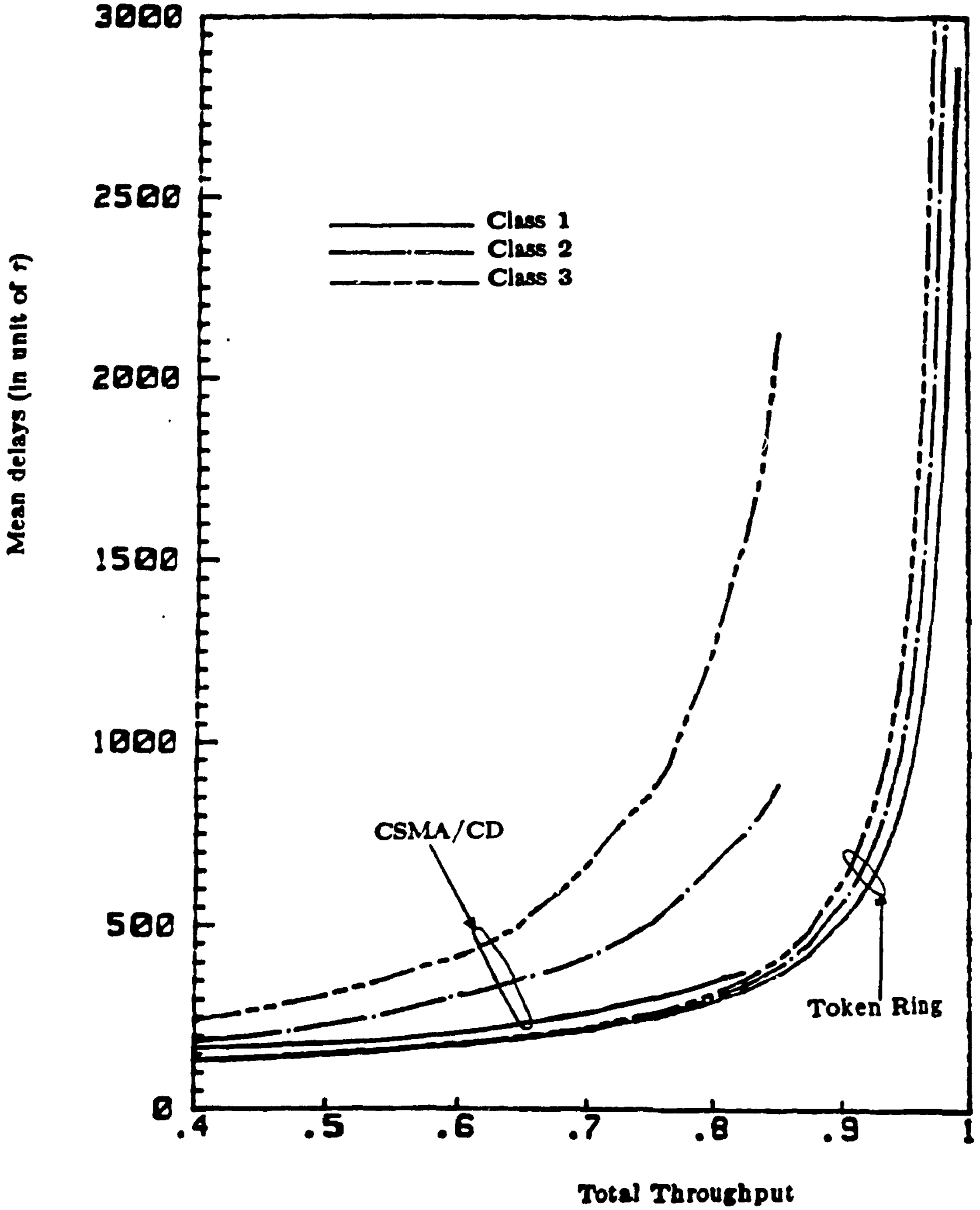


Fig.4.7: Delay comparisons for Three-priority Token Ring and CSMA/CD Deterministic message transmission time with mean 100 Scheme 1 CSMA/CD. d_{ij} =500

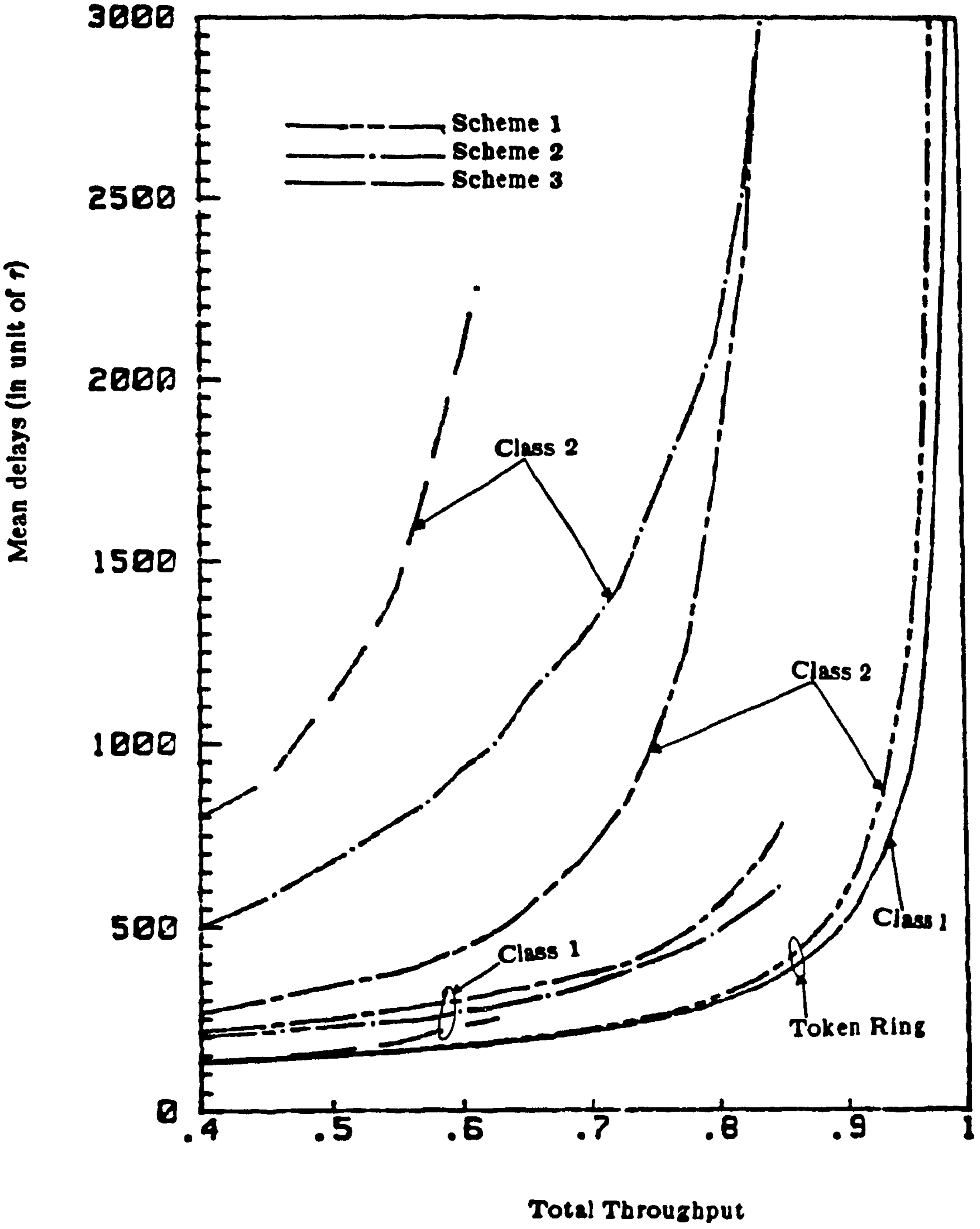


Fig.4.8: Delay comparisons for Two-priority schemes 1, 2, 3 CSMA/CD and Token Ring Deterministic message transmission time with mean 100
d_{ij} =500

4.3 REMARKS

It is important to emphasize again that the operation of the priority mechan'sm is not free. There is a well-defined cost (measured in terms of increased average delay for low classes in order to gain for high classes) associated with its implementation. In general there is a minimum discrimination made between the various priority levels and more we gain for higher classes, the more we lose for lower classes. The intriguing point is the degree of satisfaction we are looking for; how much we wish to gain and conversely until which point we would accept losing.

The simplicity of our analysis, despite its generality, may show that the cost of the priority mechanism is an easily computable function of various network parameters. Gianini and Manfield's analysis confirms the contrary. They treat only a two-priority system; moreover, they are not able to find all expressions for the asymmetric case despite the complexity of the expressions they obtained. The extension of their analysis to systems with three or more priority levels will involve solving large systems of linear equations. Under the same parameters, the following interesting point can be made. The lower bound for the delay of the lowest priority message in G&M model is given by the delay for the lowest priority message in our model which is easily computable.

Although the analysis of the prioritized CSMA/CD is general, its complexity is not negligible. It includes a determination of the nonutilization period for the ith priority and the activity factor defined as the average number of attempts required for a message to be successfully transmitted. Although the nonutilization period can be obtained through the solution of $2g+1$ simultaneous linear equations, on the other hand obtaining the activity factor requires the solution of g simultaneous nonlinear equations.

Limited service discipline can be easily justified either under G&M model or under ours. In fact, this service discipline possesses a characteristic desirable in both models, however for contradictory reasons. In all cases, we wish to restrict service to one in order to reduce the variance of the token cycle. Applying limited service to G&M model will have the effect of lowering the delay for the highest priority, and giving better guaranteed service at a large cost for the lowest priority. We can intuitively affirm that in G&M model, one of the circumstances where the priority mechanism results in a large and severe discrimination between priorities is when service is limited to a single message per token. Applying exhaustive or gated service disciplines to our model will have the effect of lowering the delay for highest priority (however not to the same level as in G&M model) and still at relatively acceptable cost of increased delay for the lower priorities. Under our model, we can intuitively say that the largest discrimination between priorities is achieved when the service is exhaustive, while the smallest is achieved when the service is limited-to-one. Between priority classes, the paradoxal fairness (in our model) and unfairness (in G&M model) is essential and in both models, limited-to-one service discipline is necessary (but not sufficient) to gain "maximum desired benefit" from priority operation. An important consequence of the variability in discrimination among the priorities is that one has the choice of moderate, intermediate, and severe delay difference among priorities, depending on the system requirements under normal operation (usually at low to medium load) or at extreme conditions. Therefore, an ideal prioritized token ring would be a mixture of the models.

The discrimination between priorities is not great unless the network is operating under overload. A combination of other parameters such as the load, the number of stations, and the message transmission times may also contribute for the discrimination among priorities.

General

The increasing use of distributed processing applications and personal computers has led to a need for a flexible strategy for local networking. Support of premises-wide data communications requires a networking and communications service that is capable of spanning the distances involved and that interconnects equipment in a single (perhaps large) building or a cluster of buildings. Although it is possible to develop a single local network to interconnect all the data processing equipment of a premises, this is probably not a practical alternative in most cases. There are several drawbacks:

Reliability: With a single local network, a single interruption, even of short duration, could result in major disruption for users.

Capacity: A single local network could be saturated as the number of devices attached to the network grows over time.

Cost: A single network technology is not optimized for the diverse requirements for interconnection and communication. The presence of large numbers of low-cost minicomputers dictates that network support for these devices be provided at low cost. Local networks that support very low cost attachment will not be suitable for meeting the overall requirement.

Performance: With a single local network, the performance is degraded as the number of devices attached to it grows over time. Furthermore, when the local network is augmented by a priority mechanism, lower class delays may become so large that it would be unacceptable in any realistic application even when the minimum discrimination among classes is achieved.

A more attractive alternative is to combine lower-cost, lower-capacity local networks within buildings or departments and to link these networks with a higher-capacity (and therefore, high cost) local network. This latter network is referred to as backbone local network. Although many studies have shown that intradepartmental and intrabuilding communications greatly exceed interbuilding communications. It is to be expected that the backbone network in a large distributed processing environment will have to sustain high peak load and, as the demand for communications grows over time, high sustained loads. Thus, a major requirement for the backbone is high capacity. In the next chapter, a topology together with accessing technique for interconnection of LANs by a backbone network, both of which use token passing, is studied. Bux and Grillo[52] investigated flow-control issues in this kind of system.

CHAPTER V

The Performance of Interconnected Ring Networks

Local Area Networks connect data sources in a limited geographical area to one another and to a central facility. Such networks have been widely deployed and constitute an essential component of many information processing and information gathering systems. Because of their wide utility and limited area of service, it is becoming increasingly necessary to interconnect them to one another in order to offer better service. The basic motivation for network interconnection is the same as that which led to networking in the first place: a desire to interconnect devices, mainframes, and so forth, to promote or allow sharing of resources and communication among users. In a sense, then, internetting creates the illusion of a single large network. A logical interconnection configuration is a high speed backbone network which offers a common path for all traffic between LANs. Our interest is in a general topology which would allow for growth and flexibility. In this chapter, a topology together with accessing techniques which is a logical solution to the interconnection problem is studied. The specific problem under study is the interconnection of LANs with a ring topology. In order to simplify the development and operation of such systems, it is reasonable to use the same ring topology for the backbone network.

It is of interest, if only from a historical point of view, that the first networks which were dedicated to data traffic were in the form of a hierarchy of ring networks[19,20]. The resulting topology is shown in Figure 5.1. We have a set of outer-rings, which are the LANs connected together by an inner-ring, the backbone network. The point of contact between the LAN and the backbone network where the internetwork traffic flows is the gateway.

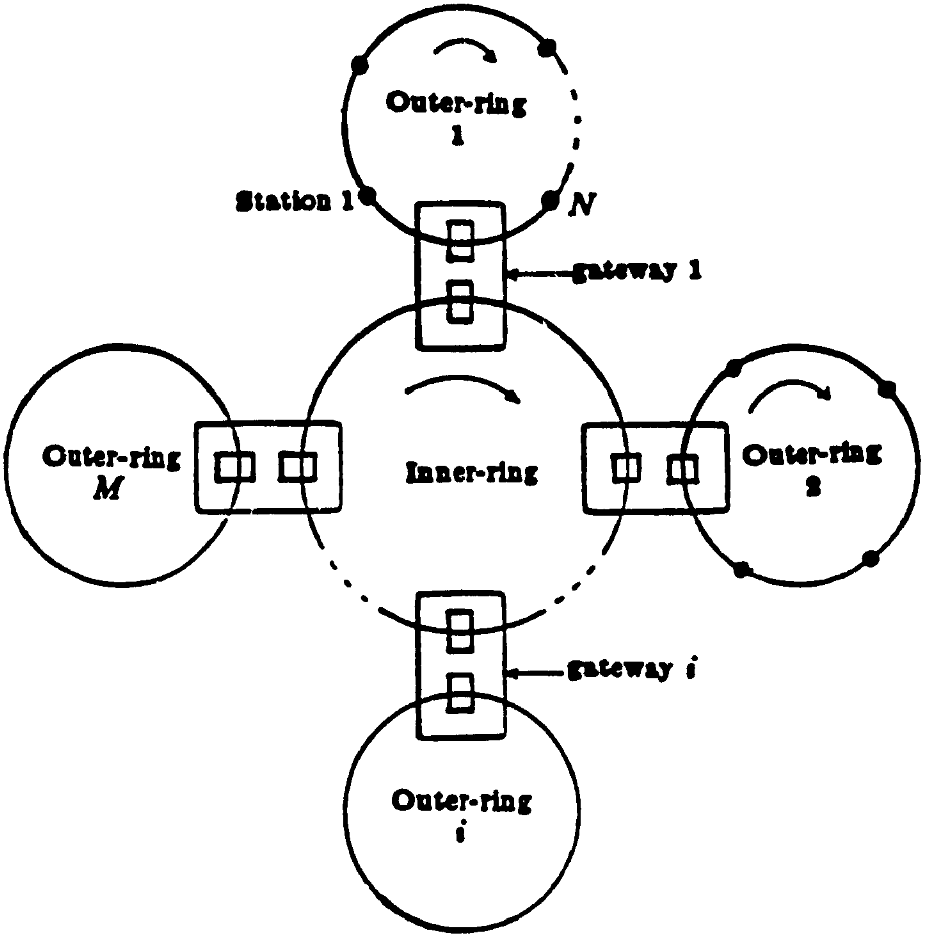


Fig. 5.1: Schematic view of a System of Interconnected Rings.

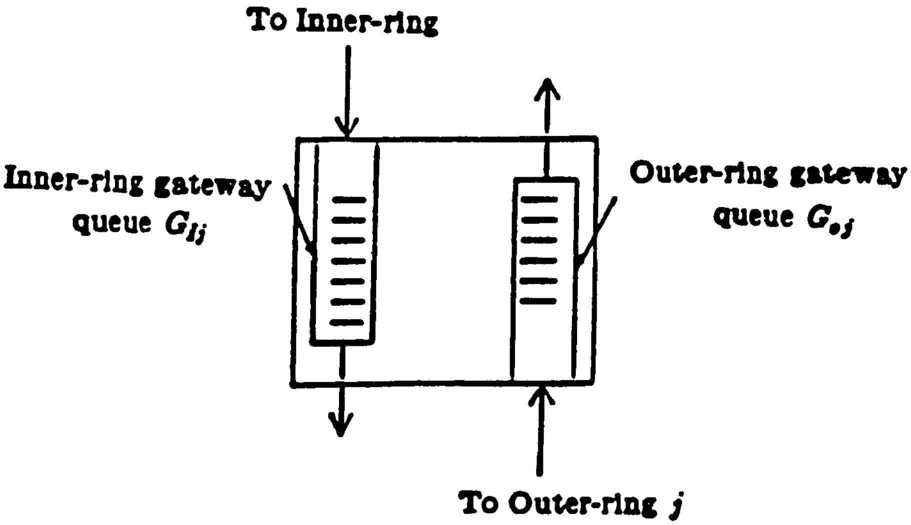

To Inner-ring
Inner-ring gateway queue G_{Ij}
Outer-ring gateway queue G_{oj}
To Outer-ring j

Fig. 5.2: A Gateway Structure.

The gateway depicted in Figure 5.2 has two buffers together with the attendant circuitry. A message destined for a distant local ring passes through a gateway to the inner-ring. The inner-ring conveys the message to the gateway of the distant ring.

From operational point of view, each outer-ring consists of a number of station queues and a half of a gateway referred to as outer-ring gateway queue. Inner-ring queues are the other halves of the gateways carrying traffic from the inner-ring to the outer. They are referred to as inner-ring gateway queues. A fundamental assumption of the study is that token passing is used. (Once the ring topology has been chosen, token passing as the access technique follows logically since it is an almost universal standard).

In the Token Passing technique, a logical token circulates in an outer-ring. The capture of the token by a station allows it to place a message on the ring. The message is accompanied by the address of the destination station. Each station reads the addresses of messages passing through and removes those with its address. In the same fashion messages from local stations are transmitted to the gateways. After a single message is transmitted (limited-to-one service), control of the ring is relinquished by passing the token on to the next station in a logical sequence. We also assume that the gateway queues are treated in the same fashion as all the other queues in the outer-ring networks. The same protocol is followed in the inner (backbone) ring. A token circulates among the inner-ring gateway queues; its capture by an inner-ring gateway queue allows it to transmit a message to an outer-ring gateway queue.

In this chapter, we determine the mean delay for both local and internetwork traffic for two modes of operation at the station queues. The basic mathematical model which is used in this study is as before that of polling. The Poisson arrival of messages to the outer-ring stations has been assumed, with two classes of messages. Class 1 is an inter ring message, while class 2 message stays on the same ring. We should note that the message interarrival times to the gateway queues are not exponential; consequently, the arrival processes to the gateway(outer and inner-ring) queues are non-Poisson. An outer-ring consists of N station queues receiving messages according to a Poisson process and an outer-ring gateway queue with non-Poisson arrival process. The inner-ring connects M inner-ring gateway queues with non-Poisson arrival to each of them. Thus, the aggregate message arrivals to both outer and inner-ring gateway queues are non-Poisson and the results developed in chapter III do not apply. By means of a certain independence assumption and under a limited service discipline, outer-ring stations are modeled as an M/G/1 queue in this chapter. The gateways are modeled as G/G/1 queues. In both types of queues, a message initiating a busy period receives special service. The numerical results derived from the analysis are validated by simulation (see appendix D for the simulation program).

5.1. THE MODEL

The mathematical model assumes M identical outer-rings interconnected by means of a backbone inner-ring. Each outer-ring consists of N station queues and a half of a gateway referred as outer-ring gateway queue. The inner-ring connects M halves of the M identical gateways. The outer and inner-ring gateway queues transfer messages to/from an outer-ring from/to an inner-ring respectively. The station queue $j(j=1,2,..,N)$ on ring $i(i=1,2,..,M)$ will be referred to as q_{ij}. The outer-ring gateway queue on ring i and the inner-ring gateway queue receiving messages from the ring i are referred to as G_{oi} and G_{Ii} respectively. Messages flowing in the network are classified as class 2 or 1, respectively, according to whether they are destinated for the same or another ring than the one on which they are generated.

Each station, with infinite storage capacity, receives an independent Poisson message arrival stream at a rate λ messages per second. With probability α a message is in class 1, and its destination is equally likely to be any other outer-ring. When the outer-ring i server arrives at queue q_{ij}, service is given to only one message, if any is present and then it goes to inspect q_{ij+1} (note that q_{iN+1} is called G_{oi} for the outer-ring gateway queue on the outer-ring i), following a walktime required for the transfer of the token. The inner-ring functions in the same way. If the token is at station j on the inner-ring and there is at least one message in G_{Ij}, a message is transmitted. The token is then passed on to queue G_{Ij+1}.

Two service disciplines are considered. In the first case, all messages at a station queue are treated equally. In the second case, inter-ring messages are given non-preemptive priority over local messages since their path in the network is much longer. In both cases we assume that the probability that a queue is busy is independent of all other queues. This and other approximations are justified through simulation.

5.2. THE ANALYSIS

The dependency between stations, gateways, outer and inner-rings, coupled with the non Poisson property of the arrivals to the gateway queues make an exact analysis untractable; accordingly we present an approximate analysis.

The focus of this analysis is to determine the delay for class 1 and class 2 messages. Since a class 2 message stays on the same outer-ring, its delay is equal to the delay in an outer-ring station queue. In contrast, a class 1 message goes through an initial outer-ring, inner-ring and a final outer-ring. The delay components are outer-ring station queue, inner-ring gateway queue, and outer-ring gateway queue delays, respectively. The objective of the following analysis will be to determine each of the delay components. Thus we shall focus on three distinct types of queues in the system: the outer-ring station queues, the outer-ring gateway queues and the inner-ring gateway queues. All these queues are assumed to be independent except for a weak dependency among their service times to be explained later on. The outer-ring station queues are modeled as M/G/1 with two classes of messages.

Our problem differs from the classical one in that a message arriving at an empty station receives special service since it arrives at a random point during the server cycle. An extension of previous work on non-preemptive priorities is carried out for exceptional first service in order to model the system [55]. The arrival processes to both the outer and inner-ring gateway queues are non Poisson; consequently, they are modeled as G/G/1 queues. As in the previous case, a message arriving at an empty gateway queue receives special service; thus existing G/G/1 results are not applicable and an extension of previous analyses is necessary [55]. Before starting the analysis, it is important to note that the key step of the analysis involves a feedback mechanism which is crucial for the solution. The utilization factors for each of the three types of queue we will consider are assumed known quantities. All the desired expressions and specially mean service times are calculated as functions of these utilization factors. Only at the end, these utilization factors are obtained from their basic definition, as product of mean service time and mean arrival rate. In order to obtain these quantities, it is required to solve a system of three nonlinear equations.

5.2.1 The Outer-Ring Stations

5.2.1.1 Case Without Priority

In this section, each station is modeled as M/G/1 queue. Considering a particular outer-ring station queue, the outer-ring server is fictively always there. However, the service time of a message is no longer its transmission time as in a normal M/G/1 queue, but it includes the walktimes and the residence times of the token in the other queues on that outer-ring as well as its transmission time. The objective is to determine the service time of a message arriving at a station to be served by the outer-ring server. In order to calculate this service time, it is necessary to distinguish between messages that arrive at a busy queue and those that arrive at an idle queue. A message which has waited in a queue will see a departing token when it comes to the head of the queue. The token returns after all the other stations on the ring have had access; whereupon, the message is transmitted. The time elapsing between the accession of a message to the head of a queue and its departure is counted as its service time. This service time is made up of several components, residence of the token at station queues and the gateway queue, the time required to pass the token between queues and the time required to transmit the message itself.

Under the assumption of symmetric traffic on the outer-ring, the LST of the probability distribution for the time of residence of the server (token) in a station queue may be written as

$$F(s) = \alpha\rho_s M_1(s) + (1-\alpha)\rho_s M_2(s) + 1 - \rho_s \tag{5.2.1a}$$

where $M_1(s)$ and $M_2(s)$ are the LSTs for the distributions of the class 1 and class 2 message transmission times, respectively and where ρ_s is the probability of a station queue being busy. α is the portion of class 1 messages. This quantity ρ_s is the traffic load on a station and is a variable to be determined as part of the analysis. The time that the token resides in an outer-ring gateway queue has the corresponding LST

$$G_o(s) = \rho_o M_1(s) + 1 - \rho_o \tag{5.2.1b}$$

where ρ_o is the utilization factor of an outer-ring gateway queue which gives the probability of it being occupied. Again ρ_o will be determined as part of the analysis. It is important to note that when a message which arrived at a busy station queue reaches the head of the queue, the token has just left that station, hence the message sees a complete cycle. Under the independence assumption and identically distributed walk times the LST for the complete service time of a class j message arriving at a busy queue is given by:

$$Y_{s,j}(s) = [G_o(s)][F(s)]^{N-1}[R(s)]^{N+1}M_j(s) \quad j=1,2 \qquad (5.2.2a)$$

where $M_j(s)$ and $R(s)$ are the LST of class j message transmission time and walktime respectvely.

The arrival of a message to an idle queue is assumed to occur at a random point during a cycle of the server in which the station where the message arrives is found empty. The message which arrives to an empty queue sees only part of a token's cycle through the ring, the residual life of this token's cycle. Thus the Laplace transform of the pdf of the service time of a class j message which arrives at an idle station is assumed to be given by:

$$\hat{Y}_{s,j}(s) = \frac{1 - X(s)}{s\bar{X}}.M_j(s) \quad j=1,2 \qquad (5.2.2b)$$

where $X(s)$ and $\bar{X}$ are the LST of the distribution and the mean for the special cycle during which the station where the message arrives is found to be empty. The term $\{[1 - X(s)]/s\bar{X}\}$ represents the LST of the residual life of this special cycle.

$X(s)$ and $\bar{X}$ are given by :

$$X(s) = G_o(s).[F(s)]^{N-1}.[R(s)]^{N+1}$$

and

$$\bar{X} = -X^{(1)}(0) \qquad (5.2.3)$$

From the above equations, the mean and second moment of the service time of a class j message arriving at a busy or empty station can be determined.

The assumption that enables the analysis to be carried through is, of course, that of independence among the queues. Previous analyses based on this assumption have given reasonably accurate results [56,59]. In view of the other features of the analysis, particularly, the application of the exceptional service to non-Poisson arrival, the assumption seems justified. Finally, as we shall see, the analytical results show excellent agreement with simulation.

We assume that the service times are i.i.d with transforms determined by equations (5.2.2a) and (5.2.2b) for messages arriving at a busy and idle station respectively. Thus a station may be modeled by an M/G/1 queue with two types of messages in which a message initiating a busy period receives different service than the others. When both types of message are treated in the same way, the mean delay of a class j message and the other quantities may be determined by straightforward computation [26,58]. The Laplace transform of the overall message delay, including queueing and service times can be shown to be[26,58]:

$$D_s(s) = \frac{(1-\rho_s)[\lambda Y_s(s) - (\lambda - s)\hat{Y}_s(s)]}{\lambda Y_s(s) + s - \lambda} \tag{5.2.4}$$

where

$$Y_s(s) = \alpha Y_{s1}(s) + (1-\alpha) Y_{s2}(s) \tag{5.2.5a}$$

and

$$\hat{Y}_s(s) = \alpha \hat{Y}_{s1}(s) + (1-\alpha) \hat{Y}_{s2}(s) \tag{5.2.5b}$$

represent the LST of the pdfs for the overall service times when arrived at a busy and an idle outer-ring station queue respectively. $Y_{sj}(s)$ and $\hat{Y}_{sj}(s)$ ($j=1,2$) are given by equations (5.2.2a) and (5.2.2b) respectively.

It is important to note that the queueing delay (waiting time) is the same for both types of messages. The transform of its pdf is given by:

$$W(s) = \frac{D_s(s)}{\rho_s Y_s(s) + (1 - \rho_s)\hat{Y}_s(s)} \qquad (5.2.6a)$$

and the transform of the delay distribution for a class j message is given by:

$$D_{sj}(s) = W(s)\,[\rho_s Y_{sj}(s) + (1 - \rho_s)\hat{Y}_{sj}(s)\,] \quad j=1,2 \qquad (5.2.6b)$$

By successive differentiations, the mean delay $\overline{d_{sj}}$ can be determined. Equation (5.2.6b) and $\overline{d_{sj}}$ represent respectively the transform and the mean delay experienced by a class j message at a station queue. These two terms are functions of ρ_s and ρ_o (utilization factors of a station and an outer-ring gateway queues respectively) which will be determined in the sequel.

5.2.1.2 Case With Nonpreemptive Priority

Now, we consider the case where class 1 messages are given non-preemptive priority over class 2 messages. Also in this case, it is necessary to distinguish between messages that arrive at a busy station queue and those that arrive at an idle queue. As the service times are defined, equations (5.2.2a) and (5.2.2b) representing the LST for the complete service times of a class j message arriving at a busy and an idle outer-ring station queue, respectively, still hold. We shall rely on the Markov chain imbedded at a class j service completion epoch, indicated by the term " j class epoch", where $j=1$ or 2, to obtain the mean delay of each message class at an outer-ring station queue. The results we derive in this section are general results applicable to any prioritized M/G/1 queue in which a customer initiating a busy period receives special service. For our main interest, we consider only two priority levels. However, the results can be extended easily to more than two priorities. This will constitute an extension of the results obtained by Cox and Smith[72].

It is assumed that each station, with infinite storage capacity, receives an independent Poisson message arrival stream at a rate of λ messages per second. With probability α, a message is in class 1 and its destination is equally likely to be any other outer ring. If λ_j $(j=1,2)$ is the arrival rate of class j message, then:

$\lambda_1 = \alpha\lambda$ and $\lambda_2 = (1 - \alpha)\lambda$

Let n_{ij} denote the number of class j messages in an outer ring station queue at the ith departing point.

n_{ij} can be expressed in terms of the number in the queue at the previous epoch and the number of new arrivals.

We may write for $n_{i1} > 0$.

$$n_{i+1,1} = n_{i,1} - 1 + a_{11} \qquad (5.2.7a)$$

$$n_{i+1,2} = n_{i,2} + a_{21} \qquad (5.2.7b)$$

where a_{kj} $(j=1,2)$ is the number of class k messages to arrive during a class j message service time when the queue is nonempty.

If the $(i+1)$st departure is a class 2 epoch, there could not have been any class 1 message in the queue at the ith departure and we have for $n_{i,2} > 0$.

$$n_{i+1,1} = a_{12} \qquad (5.2.8a)$$

$$n_{i+1,2} = n_{i,2} - 1 + a_{22} \qquad (5.2.8b)$$

The final equation can be obtained by considering the situation when the ith departure leaves the queue empty. Since probability of the next arriving message being class j is λ_j/λ, it is clear that, at the $(i+1)$st departure, all messages in the queue arrived during the service time of this message.

Thus with probability λ_j/λ, we have

$$n_{i+1,l} = \hat{a}_{lj}, \quad l,j=1,2 \qquad (5.2.9)$$

Where $\hat{a}_{lj}$ is the number of class l messages to arrive during the service time of the class j message which initiated the busy period.

We define

$$P_1 \triangleq Prob.\left[n_{i,1}>0\right]$$

$$P_2 \triangleq Prob.\left[n_{i,1}=0\ ,\ n_{i,2} > 0\right] \qquad (5.2.10)$$

and

$$P_o \triangleq Prob\left[n_{i,1} = n_{i,2} = 0\right]$$

We calculate the two-dimensional probability-generating functions of $n_{i+1,1}$ and $n_{i+1,2}$ by conditioning on these three events. From equations (5.2.7a,b) to (5.2.9), we have

$$E\left[z_1^{n_{i+1,1}}\ z_2^{n_{i+1,2}}\right] = P_1\, E\left[z_1^{n_{i,1}-1+a_{11}}\ z_2^{n_{i,2}+a_{21}}\ \ / \ n_{i,1}>0\right]$$

$$+ P_2 E\left[z_1^{a_{12}}\ z_2^{n_{i,2}-1+a_{22}}\ \ / \ n_{i,1}=0,\ n_{i,2} >0\right]$$

$$+ P_o \sum_{j=1}^{2} \frac{\lambda_j}{\lambda}\, E\left[z_1^{\hat{a}_{1j}}\ z_2^{\hat{a}_{2j}}\ \ / \ n_{i,1}=n_{i,2}=0\right] \qquad (5.2.11)$$

We follow the same steps as in the one-dimensional case to solve for the probability generating function. The number of message arrivals is independent of the number of messages in the station queue. Since the message arrivals to the outer-ring station queues are Poisson, we have the following familiar relations for the number of arrivals during message service times:

$$E\left[z_1^{a_{1j}} \;\; z_2^{a_{2j}} \right] = Y_{sj}\left(\lambda_1 (1 - z_1) + \lambda_2 (1 - z_2)\right) \qquad (5.2.12a)$$

and

$$E\left[z_1^{\hat{a}_{1j}} \;\; z_2^{\hat{a}_{2j}} \right] = \hat{Y}_{sj}\left(\lambda_1 (1 - z_1) + \lambda_2 (1 - z_2)\right) \qquad (5.2.12b)$$

where $Y_{sj}(s)$ and $\hat{Y}_{sj}(s)$ are the LST of the pdf for the service time of a message when arrived at a busy and an outer-ring station queue respectively.

Again by conditioning on the same events, we may write

$$\begin{aligned} E\left[z_1^{n_{i+1,1}} \; z_2^{n_{i+1,2}}\right] &= P_1\, E\left[z_1^{n_{i+1,1}} \;\; z_2^{n_{i+1,2}} \;\; / \; n_{i+1,1} > 0\right] \\ &+ P_2\, E\left[z_1^{n_{i+1,2}} \;\; / \; n_{i+1,1}=0,\; n_{i+1,2} > 0\right] \\ &+ 1 - P_1 - P_2 \end{aligned} \qquad (5.2.13)$$

By assuming equilibrium has been attained, we may write

$$\begin{aligned} G_1(z_1,z_2) &\triangleq E\left[z_1^{n_{i,1}} \;\; z_2^{n_{i,2}} \;\; / \; n_{i,1}>0\right] \\ &= E\left[z_1^{n_{i+1,1}} \;\; z_2^{n_{i+1,2}} \;\; / \; n_{i+1,1}>0\right] \end{aligned} \qquad (5.2.14a)$$

and

$$\begin{aligned} G_2(z_2) &\triangleq E\left[\, z_2^{n_{i,2}} \;\; / \; n_{i,1}=0 \,,\; n_{i,2}>0\right] \\ &= E\left[z_2^{n_{i+1,2}} \;\; / \; n_{i+1,1}=0,\; n_{i+1,2}>0\right] \end{aligned} \qquad (5.2.14b)$$

Substituting equations (5.2.12a,b), (5.2.13) and (5.2.14a,b) into equation (5.2.11), we obtain after some rearangements the following expression which describes completely an outer-ring station queue.

$$P_1 G_1(z_1,z_2)\, z_2 \,[\, z_1 - Y_{e1}(s_o)\,] + P_2 G_2(z_2)\, z_1 \,[\, z_2 - Y_{e2}(s_o)\,]$$
$$+ P_o\ z_1\, z_2 \,[\, 1 - \sum_{k=1}^{2} \frac{\lambda_k}{\lambda} \hat{Y}_{ek}(s_o)\,] = 0 \qquad (5.2.15)$$

where we define

$$s_o = \lambda_1(1 - z_1) + \lambda_2(1 - z_2)$$

By differentiating equation (5.2.15) with respect to z_1 and z_2, and setting $z_1 = z_2 = 1$, we obtain two equations. Solving this set of equations yields

$$P_1 \triangleq Prob.\left[n_{i,1} > 0\right] = \frac{\lambda_1}{\lambda}\rho_e \qquad (5.2.16a)$$

$$P_2 \triangleq Prob.\left[n_{i,1} = 0\ ,\ n_{i,2} > 0\right] = \frac{\lambda_2}{\lambda}\rho_e \qquad (5.2.16b)$$

with

$$\rho_e \triangleq 1 - P_o = 1 - Prob\left[n_{i,1} = n_{i,2} = 0\right]$$
$$= \frac{\sum_{k=1}^{2} \lambda_k \overline{\hat{y}_{ek}}}{1 - \sum_{k=1}^{2} \lambda_k \left[\overline{y_{ek}} - \overline{\hat{y}_{ek}}\right]} \qquad (5.2.16c)$$

By differentiating a second time equation (5.2.15) with respect to z_1 and with respect to z_2, we obtain three partial derivatives corresponding to all pairs of z_1 and z_2. Setting $z_1 = z_2 = 1$ in these equations yields three simultaneous equations which can be reduced to the following expressions:

$$\overline{n_{11}} = 1 + \frac{\lambda_1 \left\{ \rho_e \sum_{k=1}^{2} \lambda_k \overline{y_{ek}^{(2)}} + (1 - \rho_e) \sum_{k=1}^{2} \lambda_k \overline{\hat{y}_{ek}^{(2)}} \right\}}{\left(1 - \lambda_1 \overline{y}_{e1}\right)} \qquad (5.2.17a)$$

$$\overline{n_{21}} = \frac{\rho_e \lambda_2 \overline{y_{e1}}\left(\overline{n_{11}} - 1\right) + \rho_e \lambda_2 \overline{y}_{e2}\left(\overline{n_{22}} - 1\right) + \lambda_2 \sum_{k=1}^{2} \lambda_k \left[\rho_e \overline{y_{ek}^{(2)}} + (1-\rho_e)\overline{\hat{y}_{ek}^{(2)}}\right]}{\rho_e \left(1 - \lambda_1 \overline{y}_{e1}\right)} \qquad (5.2.17b)$$

$$\overline{n_{22}} = 1 + \frac{2\rho_s \lambda_1 \overline{y_{s1} n_{21}} + \sum_{k=1}^{2} \lambda_k \left[\rho_s \overline{y_{sk}^{(2)}} + (1-\rho_s) \overline{\hat{y}_{sk}^{(2)}} \right]}{2\rho_s \left(1 - \lambda_2 \overline{y}_{s2} \right)} \qquad (5.2.17c)$$

where

$$\overline{n_{11}} = \frac{\partial G_1(z_1, z_2)}{\partial z_1} \bigg|_{z_1 = z_2 = 1} = E\left[n_1 \; / \; n_1 > 0 \right]$$

$$\overline{n_{21}} = \frac{\partial G_1(z_1, z_2)}{\partial z_2} \bigg|_{z_1 = z_2 = 1} = E\left[n_2 \; / \; n_1 > 0 \right]$$

and

$$\overline{n_{22}} = \frac{\partial G_2(z_2)}{\partial z_2} \bigg|_{z_1 = z_2 = 1} = E\left[n_2 \; / \; n_1 = 0 \, , \, n_2 > 0 \right]$$

The expected number of class 2 messages at a 2-epoch is then found by substituting eq. (5.2.17b) in eq. (5.2.17c) as:

$$\overline{n_{22}} = 1 + \frac{\lambda_2 \left\{ \rho_s \sum_{k=1}^{2} \lambda_k \overline{y_{sk}^{(2)}} + (1 - \rho_s) \sum_{k=1}^{2} \lambda_k \overline{\hat{y}_{sk}^{(2)}} \right\}}{2\rho_s \left(1 - \lambda_1 \overline{y}_{s1} \right) \left(1 - \lambda_1 \overline{y}_{s1} - \lambda_2 \overline{y}_{s2} \right)} \qquad (5.2.18)$$

and after applying Little's results to customers in each class queue we have for the average waiting time of each message class

$$\overline{w'_{s1}} = \frac{\rho_s}{\lambda_1} \left(\overline{n}_{11} - 1 \right) = \frac{\rho_s \sum_{k=1}^{2} \lambda_k \overline{y_{sk}^{(2)}} + (1 - \rho_s) \sum_{k=1}^{2} \lambda_k \overline{\hat{y}_{sk}^{(2)}}}{2 \left(1 - \lambda_1 \overline{y}_{s1} \right)} \qquad (5.2.19a)$$

$$\overline{w'_{s2}} = \frac{\rho_s}{\lambda_2} \left(\overline{n}_{22} - 1 \right) = \frac{\rho_s \sum_{k=1}^{2} \lambda_k \overline{y_{sk}^{(2)}} + (1 - \rho_s) \sum_{k=1}^{2} \lambda_k \overline{\hat{y}_{sk}^{(2)}}}{2 \left(1 - \lambda_1 \overline{y}_{s1} \right) \left(1 - \lambda_1 \overline{y}_{s1} - \lambda_2 \overline{y}_{s2} \right)} \qquad (5.2.19b)$$

and the average delay of a class j message including its service time in an outer ring station queue is:

$$\overline{d_{sj}} = \overline{w_{sj}} + \rho_s \overline{y}_{sj} + (1 - \rho_s)\overline{\hat{y}_{sj}}, \qquad j = 1,2 \tag{5.2.20}$$

We should note that in both cases $\overline{d}_{s2}$ represents the average delay of a class 2 message not only in the outer-ring station but also in the complete system. $\overline{d}_{s1}$ is the first of three components of the delay experienced by a class 1 message. The other two components of this delay, denoted by $\overline{d}_I$ and $\overline{d}_o$ for mean delays in the inner-ring and destination outer-rings respectively, will be determined next.

5.2.2 The Gateway Queues as Generalized G/G/1

The arrival processes to both the outer and inner-ring gateway queues are non-Poisson. Consequently, they are modeled as G/G/1 queues. In this section, each outer-ring and inner-ring gateway queue is to be modeled as a G/G/1 queue in which a message initiating a busy period receives special service. First, let us consider an outer-ring gateway queue.

We consider the situation where messages arrive at an outer-ring gateway queue according to an independent increment process, i.e. the intervals between arrivals are a set of independent, identically distributed random variables. Messages may queue while awaiting service. We assume that the outer-ring gateway queues may have infinite capacity. We distinguish two different probability distributions for the service times. The service time of messages which arrive at an empty outer-ring gateway queue, thereby initiating a busy period, may be different than those for messages which have waited in the queue. The probability distributions of the service times are otherwise arbitrary except for a constraint on the average service time which insures stability.

In order to proceed with the analysis we define the following quantities:

$C_n \triangleq$ the event of the nth customer arriving to the queue.

$t_n \triangleq$ interarrival time between C_{n-1} and C_n .

$y_{on} \triangleq$ service time for C_n when arrived at a busy queue.

$\hat{y}_{on} \triangleq$ service time for C_n when arrived at an idle queue.

$w_{o_n} \triangleq$ waiting time (in queue) for C_n .

We assume that the random variables $\{t_n\}$, $\{y_{on}\}$ and $\{\hat{y}_{on}\}$ are independent and are given, respectively, by the distribution functions $A(t)$, $\hat{B}(\hat{y}_o)$ and $B(y_o)$ independent of the subcript n .

For convenience we define two new random variables

$$e_n \triangleq y_{on} - t_{n+1}$$

$$\hat{e}_n \triangleq \hat{y}_{on} - t_{n+1} \tag{5.2.21}$$

These random variables are merely the difference between the service time for C_n and the interarrival time between C_{n+1} and C_n . For a stable system, we will require that the expectations of e_n and $\hat{e}_n$ be negative.

The waiting time of the $(n+1)$th message may be expressed in terms of the waiting time of the nth message as follows:

$$w_{on+1} = \max\left\{0,\ w_{on} + e_n\, U(w_{on}) + \hat{e}_n \left[1 - U(w_{on})\right]\right\} \tag{5.2.22a}$$

where

$$U(w_{on}) = \begin{cases} 1 & \text{for } w_{on} > 0 \\ 0 & \text{for } w_{on} \leq 0 \end{cases} \qquad (5.2.22b)$$

In equation (5.2.22a) we note the difference between messages which arrive at an empty queue ($w_{on} = 0$) and a non empty queue ($w_{on} > 0$). The term $w_{on} + e_n U(w_{on}) + \hat{e}_n \; [1 - U(w_{on})]$ represents the sum of unfinished work (w_{on}) found by C_n plus the service (y_{on} or $\hat{y}_{on}$), which it now adds to the unfinished work, less the time duration (t_{n+1}) until the arrival of C_{n+1}. For a stable system, we will require that the expectation of $\{e_n U(w_{on}) + \hat{e}_n \left[1 - U(w_{on})\right]\}$ be negative. This is equivalent to the expectations of e_n and $\hat{e}_n$ being both negative.

We now define, as in [57], another random variable which is the "other half " of the waiting time, and corresponds to the length of the idle period which is terminated with the arrival of C_{n+1}, namely

$$v_n = -min\left\{0, \; w_{on} + e_n U(w_{on}) + \hat{e}_n [1 - U(w_{on})]\right\} \qquad (5.2.23)$$

We note that either w_{on+1} or v_n must be 0, i.e. the product $\{w_{on+1} v_n\}$ is always nul. From these definitions we observe that

$$w_{on+1} - v_n = w_{on} + e_n U(w_{on}) + \hat{e}_n [1 - U(w_{on})] \qquad (5.2.24)$$

Forming the transform on both sides of eq.(5.2.24), we obtain

$$E\left[e^{-s(w_{on+1} - v_n)}\right] = E\left[e^{-s\left\{w_{on} + e_n U(w_{on}) + \hat{e}_n [1 - U(w_{on})]\right\}}\right] \qquad (5.2.25)$$

The left hand side of eq.(5.2.25) may be written as:

$$E\left[e^{-s(w_{on+1} - v_n)}\right] = E\left[e^{sv_n} / v_n > 0\right] . P\left[v_n > 0\right]$$

$$+ E\left[e^{-sw_{on+1}}/v_n = 0\right].P\left[v_n = 0\right] \qquad (5.2.26)$$

Allowing the limit as $n \rightarrow \infty$, we obtain the following transform expression from eq.(5.2.26) :

$$E\left[e^{-s(w_o - v)}\right] = (1 - \rho_o)I_o(-s) + W_o(s) - (1 - \rho_o) \qquad (5.2.27)$$

where $w_{on} \rightarrow w_o$ and $v_n \rightarrow v$

$I_o(s)$ is the Laplace transform of the idle period and

$$1 - \rho_o = P[v > 0] \qquad (5.2.28)$$

= P[arrival finds the outer-ring gateway queue idle]

We may also write the limit form of the right hand side of eq.(5.2.25) as :

$$E[e^{-s(w_o + e)}/w_o > 0].P[w_o > 0] + E[e^{-st}/w_o = 0].P[w_o = 0]$$

$$= [W_o(s) - (1 - \rho_o)]C_o(s) + (1 - \rho_o)\hat{C}_o(s) \qquad (5.2.29)$$

where $C_o(s)$ and $\hat{C}_o(s)$ are the Laplace transforms for the density describing the random variables e and $\hat{e}$ respectively. Thus, from the eq.(5.2.27) and (4.2.29), using (4.2.25), we obtain immediately

$$(1 - \rho_o)I_o(-s) + W_o(s) - (1 - \rho_o) = [W_o(s) - (1 - \rho_o)]C_o(s) + (1 - \rho_o)\hat{C}_o(s) \, (5.2.30)$$

This last equation finally gives us

$$W_o(s) = \frac{(1 - \rho_o)[1 - I_o(-s) + \hat{C}_o(s) - C_o(s)]}{1 - C_o(s)} \qquad (5.2.31)$$

It is important to notice that equation (5.2.31) is the LST of the pdf for a customer waiting time in any G/G/1 queue in which a customer initiating a busy period is given special service. This constitutes the generalization of the result obtained by P.D Welch [58] and which now applies to the system G/G/1.

The transform of the message delay, including queueing and service times, in an outer-ring gateway queue $D_o(s)$ is given by:

$$D_o(s) = \frac{(1-\rho_o)[1-I_o(-s) + \hat{C}_o(s) - C_o(s)]}{1-C_o(s)} \cdot \left(\rho_o Y_o(s) + [\,1-\rho_o\,]\hat{Y}_o(s)\right) \quad (5.2.32)$$

where

$$C_o(s) = A_o(-s)Y_o(s)$$
$$\hat{C}_o(s) = A_o(-s)\hat{Y}_o(s) \quad (5.2.33)$$

Thus, the transform of the message delay in an outer-ring gateway queue may be known once $A_o(s)$ (the transform of the interarrival to the outer-ring gateway queue), $I_o(s)$ (the transform of the idle period), $Y_o(s)$ (the transform of the service time when arrived at a busy queue), $\hat{Y}_o(s)$ (the transform of the service time when arrived at an empty queue) are determined. The approach used in section 5.1 can be followed in order to obtain $Y_o(s)$ and $\hat{Y}_o(s)$. An outer-ring gateway queue is served by the server on an outer-ring, therefore:

$$Y_o(s) = [F(s)]^N [R(s)]^{N+1} M_1(s) \quad (5.2.34)$$

where $M_1(s)$ and $R(s)$ are the LST of class 1 message transmission and walk times respectvely. $F(s)$, the LST of the probability distribution for time of residence of a token in an station queue is given by eq.(5.2.1a).

When a message arrives at an empty outer-ring gateway queue, the transform of its service time is given by:

$$\hat{Y}_o(s) = \frac{1-X(s)}{s\overline{X}} . M_1(s) \quad (5.2.35)$$

with

$$X(s) = [F(s)]^N . [R(s)]^{N+1}$$

and

$$\bar{X} = -X^{(1)}(0) \tag{5.2.36}$$

$X(s)$ and $\bar{X}$ are the LST of the distribution and the mean for the special cycle during which the outer-ring gateway queue where the message arrives is found empty.

It is important to note that equation (5.2.35) is an approximation for the LST of the service time of a message which arrives at an empty outer-ring gateway queue. This approximation is not valid for D/D/1 queues.

Now let us determine $A_o(s)$, the transform of the interarrival time to an outer-ring gateway queue. The arrivals to an outer-ring gateway queue come from the inner-ring gateway queues.

Let:

$j \triangleq$ inner-ring gateway queue number which generated the last arrival to this outer-ring gateway queue.

$k \triangleq$ number of complete cycles the server went through since the last arrival until the next arrival to the same outer-ring gateway queue.

$i \triangleq$ number of inner-ring gateway queues visited by the inner-ring server following the last complete cycle.

$\beta_{jk} \triangleq$ probability that a message from the inner-ring gateway g_{Ij} is destinated to the outer-ring gateway queue $g_{ok}\,(k \neq j)$

If we assume that a message from an inner-ring gateway queue is equally likely to go to any one of the $(M-1)$ outer-ring gateway queues, then $\beta_{jk} = \dfrac{1}{(M-1)} = \beta.$

Under these assumptions, the LST of the interarrival time may be written as:

$$A_o(s) = \beta \sum_{k=0}^{\infty} \sum_{j=1}^{M-1} \sum_{i=1}^{M-j-1} (1 - \beta\rho_I)^{k(M-1)+i-1} \beta\rho_I \,.[G_I(s)]^k \,[H_I(s)]^{kM+i-1}.[R(s)]^{k(M-1)+i} M_1(s)$$

$$+ \qquad (5.2.37)$$

$$\beta \sum_{k=0}^{\infty} \sum_{j=1}^{M-1} \sum_{i=M-j+1}^{M} (1 - \beta.\rho_I)^{k(M-1)+i-2} \beta\rho_I \,[G_I(s)]^{k+1} [H_I(s)]^{kM+i-1} [R(s)]^{k(M-1)+i} M_1(s)$$

where $G_I(s)$ represents the LST of the pdf for the token residence time in an inner-ring gateway queue (see eq. (5.2.44).

$H_I(s)$ is the LST of the probability distribution for residence time in an inner-ring gateway queue, conditioned on the event that there is no message addressed to the outer-ring gateway queue. $H_I(s)$ is given by:

$$H_I(s) = \frac{\rho_I(1 - \beta)}{1 - \rho_I\beta} M_1(s) + \frac{(1 - \rho_I)}{1 - \rho_I\beta} \qquad (5.2.38)$$

where ρ_I is the probability that an inner-ring gateway queue is busy.

The term $\frac{\rho_I(1 - \beta)}{1 - \rho_I\beta}$ represents the probability that an inner-ring gateway queue is busy given that it does not have a message to the desired destination, while the term $\frac{(1 - \rho_I)}{1 - \rho_I\beta}$ represents the probability that an inner-ring gateway queue is idle given that it does not have a message to the desired destination.

The token goes to $(kM + i - 1)$ inner-ring gateway queues before the next message to the same outer-ring gateway queue is generated. In each of the cycles of the token, there are visits to the inner-ring gateway queue serving the same outer-ring; these visits cannot generate the next message. Each one of the $k(M - 1) + i$ other visits is assumed to generate the next message with probability $(\beta\rho_I)$, and not generate it with probability $(1 - \beta\rho_I)$. Thus the number of visits in between two arrivals to the same outer-ring gateway queue is assumed to be geometric. In the final incomplete cycle, the token may or may not go to the inner-ring gateway queue serving the same outer-ring, and this accounts for the two terms in the expression.

To obtain $I_0(s)$, the transform of the idle period in which the queue is empty, we have assumed that the departure of the message which leaves the outer-ring gateway queue idle may occur at any random point during the interarrival time of the messages. Under this assumption, the idle period may be seen as the residual life of the interarrival time and its transform is given by:

$$I_0(s) = \frac{1-A_0(s)}{s\bar{t}_0} \qquad (5.2.39)$$

where $\bar{t}_0$, a function of ρ_I, is the mean interarrival time to an outer-ring gateway queue.

Again, at this point, it is important to note that eq. (5.2.39) is an approximation not valid for D/D/1 queues but completely valid only for random departure.

Now, all the transforms are known. By successive differentiations of equation (5.2.32) and after some algebra, we can determine $\bar{d}_0$, the mean delay suffered by a message in an outer-ring gateway queue (queueing time plus service time).

$$\begin{aligned}\bar{d}_0 = {} & \frac{\sigma_{y_0}^2 + \sigma_{I_0}^2 + (1-\rho_0)^2(\bar{I}_0 + \overline{\hat{y}_0} - \overline{y_0})^2}{2(1-\rho_0)(\overline{I_0} + \overline{\hat{y}_0} - \overline{y_0})} \\ & - \frac{\sigma_{y_0}^2 - \sigma_{\hat{y}_0}^2 + (\overline{y_0} - \overline{\hat{y}_0})(\overline{y_0} + \overline{\hat{y}_0} - 2\bar{t}_0)}{2(\overline{I_0} + \overline{\hat{y}_0} - \overline{y_0})} \\ & - \frac{\overline{I_0^{(2)}}}{2(\overline{I_0} + \overline{\hat{y}_0} - \overline{y_0})} + \rho_0\bar{y}_0 + (1-\rho_0)\overline{\hat{y}_0} \qquad (5.2.40)\end{aligned}$$

Equations (5.2.32) and (5.2.40) represent the Laplace transform and the mean delay in an outer-ring gateway queue respectively. Even if $Y_o(s)$, $\hat{Y}_o(s)$, $A_o(s)$ and $I_o(s)$ are known and their corresponding moments easily evaluated, these two expressions remain functions of ρ_g, ρ_o and ρ_I which will be determined following analysis of the inner-ring gateway queue.

Next we shall analyze the inner-ring gateway queues. The arrival process to these queues is also non-Poisson. Thus, an inner-ring gateway queue may be modeled as a G/G/1 queue. However, the analysis of the inner-ring gateway queue is very similar to that of the outer-ring gateway queue just presented and thus we may write similarly the following equations:

The transform of the pdf of a message delay in an inner-ring gateway queue is given by the following modification to eq. (5.2.32).

$$D_I(s) = \frac{(1-\rho_I)[1 - I_I(-s) + \hat{C}_I(s) - C_I(s)]}{1 - C_I(s)} \cdot \Big(\rho_I Y_I(s) + [1 - \rho_I]\hat{Y}_I(s)\Big) \qquad (5.2.41)$$

where

$$\begin{aligned} C_I(s) &= A_I(-s)Y_I(s) \\ \hat{C}_I(s) &= A_I(-s)\hat{Y}_I(s) \end{aligned} \qquad (5.2.42)$$

ρ_I represents the probability of an inner-ring gateway queue being busy and $A_I(s)$ is the LST of the interarrival time to an inner-ring gateway queue.

We should note that there are only class 1 messages in the inner-ring. We assume that their transmission time is the same as in the outer-rings and the walktime between inner-ring gateway queues is also the same as in an outer-ring. In this case, if M is the number of gateways, the conditional transform of the pdf of the service time of a message which arrives at a busy inner-ring gateway queue is given by:

$$Y_I(s) = [G_I(s)]^{M-1}.[R(s)]^M.M_1(s) \tag{5.2.43}$$

where $G_I(s)$ represents the transform of the pdf for the residence time in an inner-ring gateway queue.

$$G_I(s) = \rho_I M_1(s) + 1 - \rho_I \tag{5.2.44}$$

When a message arrives at an empty inner-ring gateway queue the LST of the pdf of its service time may be approximated by:

$$\tilde{Y}_I(s) = \frac{1 - X_I(s)}{s\overline{X_I}}.M_1(s) \tag{5.2.45}$$

with

$$X_I(s) = [G_I(s)]^{M-1}.[R(s)]^M$$

and

$$\overline{X_I} = -X_I^{(1)}(0) \tag{5.2.46}$$

Now let us determine $A_I(s)$, the transform of the interarrival to an inner-ring gateway queue. The arrivals to an inner-ring gateway queue come from the outer-ring gateway queues.

Let:

$j \triangleq$ outer-ring station queue number which generated the last arrival to this inner-ring gateway queue.

$k \triangleq$ number of complete cycles the outer-ring server went through since the last arrival until the next arrival to that inner-ring gateway queue.

$i \triangleq$ number of outer-ring station queues visited by the outer-ring server following the last complete cycle and which did not have message destinated to the inner-ring gateway queue.

$\beta_l \triangleq$ probability that a message from the outer-ring station queue is destinated to the inner-ring gateway queue I

If we assume that the new arrival is equally likely to come from any of the N outer-ring station queues, then $\beta_l = 1/N$.

Under these conditions, we can write $A_I(s)$, the transform of the pdf of the interarrival time to an inner-ring gateway queue as:

$$A_I(s) = \frac{1}{N} \sum_{k=0}^{\infty} \sum_{j=1}^{N} \sum_{i=1}^{N-j} (1-\alpha\rho_s)^{kN+i-1} \alpha\rho_s [G_o(s)]^k [F_o(s)]^{kN+i-1} [R(s)]^{k(N+1)+i} M_1(s)$$

$$+ \qquad (5.2.47)$$

$$\frac{1}{N} \sum_{k=0}^{\infty} \sum_{j=1}^{N} \sum_{i=N-j+2}^{N+1} (1-\alpha\rho_s)^{kN+i-2} \alpha\rho_s [G_o(s)]^{k+1} [F_o(s)]^{kN+i-2} [R(s)]^{k(N+1)+i} M_1(s)$$

where $G_o(s)$, the transform of the residence time in an outer-ring gateway queue is given by:

$$G_o(s) = \rho_o M_1(s) + 1 - \rho_o \qquad (5.2.48)$$

$F_o(s)$ is the LST of the probability distribution for residence time in an outer-ring station queue, conditioned on the event that there is no message addressed to the inner-ring gateway queue. $F_o(s)$ is given by:

$$F_o(s) = \frac{\rho_s(1-\alpha)}{1-\rho_s\alpha} M_2(s) + \frac{(1-\rho_s)}{1-\rho_s\alpha} \qquad (5.2.49)$$

The transform of the pdf of the idle period can be obtained under the same assumption as for an outer-ring gateway queue.

$$I_I(s) = \frac{1-A_I(s)}{s\overline{t_I}} \qquad (5.2.50)$$

where $\overline{t_I}$, a function of ρ_s and ρ_I, is the mean interarrival time to an inner-ring gateway queue.

The mean delay suffered by a message in an inner-ring gateway queue is given by :

$$\bar{d}_I = \frac{\sigma_{y_I}^2 + \sigma_{i_I}^2 + (1-\rho_I)^2(\overline{I_I} + \overline{\hat{y}_I} - \overline{y_I})^2}{2(1-\rho_I)(\overline{I_I} + \overline{\hat{y}_I} - \overline{y_I})} - \frac{\sigma_{y_I}^2 - \sigma_{\hat{y}_I}^2 + (\overline{y_I} - \overline{\hat{y}_I})(\overline{y_I} + \overline{\hat{y}_I} - 2\bar{i}_I)}{2(\overline{I_I} + \overline{\hat{y}_I} - \overline{y_I})} - \frac{\overline{I_I^{(2)}}}{2(\overline{I_I} + \overline{\hat{y}_I} - \overline{y_I})} + \rho_I \overline{y}_I + (1-\rho_I)\overline{\hat{y}_I} \qquad (5.2.51)$$

Since $Y_I(s)$, $\hat{Y}_I(s)$, $A_I(s)$ and $I_I(s)$ are known, their corresponding moments can be easily determined, but still these terms are functions of ρ_s, ρ_o and ρ_I.

For the case when local and inter-ring messages are treated equally, the LST of the pdf for the delay of a class j message ($D_{sj}(s)$) is given by equation (5.2.6b). When there is a priority structure at the outer-ring station queues, the mean delay experienced by a class j message is given by equation (5.2.20). The transforms of the pdf of a message delay in an outer and inner ring gateway queue are given by equations (5.2.32) and (5.2.41) respectively. Their respective averages are given by equations (5.2.40) and (5.2.51). Since all the intermediate quantities required to obtain the mean delays and in some cases the distribution of the delay are functions of ρ_s, ρ_o and ρ_I, the only unknowns are these utilization factors. We have assumed that ρ_s, ρ_o and ρ_I (probability that an outer-ring station queue, outer and inner-ring gateway queues are busy respectively) were known quantities. All the expressions and particularly the service time distributions were calculated in terms of them. From the definition of the utilization factor as the product of mean arrival rate and mean service time:

$$\rho_e \;= \alpha\lambda[\,\rho_e\,\bar{y}_{e1} + (1-\rho_e)\overline{\hat{y}_{e1}}\,] + (1-\alpha)\lambda[\,\rho_e\,\bar{y}_{e2}+(1-\rho_e)\bar{\hat{y}}_{e2}\,] \qquad (5.2.52a)$$

$$\rho_o \;= \lambda_o[\,\rho_o\,\bar{y}_o + (1-\rho_o)\overline{\hat{y}_o}\,] \qquad (5.2.52b)$$

and

$$\rho_I \;= \lambda_I[\,\rho_I\,\bar{y}_I + (1-\rho_I)\overline{\hat{y}_I}\,] \qquad (5.2.52c)$$

where the expressions in brackets are the unconditional mean message service times for the three kinds of queues we have analyzed. λ_o and λ_I represent the arrival rate to an outer-ring and an inner-ring gateway queue respectively. They are given by $1/\bar{t}_o$ and $1/\bar{t}_I$ respectively.

Equations (5.2.52a,b,c) constitute a set of three non-linear equations in ρ_e, ρ_o and ρ_I which can be solved numerically. Solving for ρ_e, ρ_o, ρ_I and substituting into the corresponding expressions determines all the quantities we are looking for. The mean delay experienced by an inter-ring (class 2) message is given by:

$$\overline{d_1} = \overline{d_{e1}} + \overline{d_o} + \overline{d_I} \qquad (5.2.53a)$$

and for local (class 2) messages

$$\overline{d_2} = \overline{d_{e2}} \qquad (5.2.53b)$$

This concludes the analysis of the system of interconnected ring networks.

5.3. NUMERICAL RESULTS

In this section, some numerical calculations and simulation results are presented to verify the assumptions introduced in the analysis. In each example, the mean delays, as calculated using our analysis, are compared with the results of a simulation, given as 95% confidence intervals. In all cases, the maximum error remains below 15%.

The simulation program ANAIS in Fortran/77 runs in a VAX-UNIX environment, but is easily transportable to other machines. The outer and inner rings work basically as described by the flow diagram in chapter III. The major difference is that we can not predict a schedule-arrival event clock for the gateway queues since the interarrival times to these queues are not exponentially distributed. Since each ring has its own server, there is one schedule-departure event clock for each of them. However, the only one schedule-departure to consider is the minimum of all of them.

The minimum and maximum number of messages to be simulated are given as inputs and in all runs, more than 100,000 messages were simulated. Six usable replications of the simulation experiment were used in order to obtain each point. The total number of outer-ring station queues (nq) is given as an integer multiple of the number of outer-rings (nrgs). The value of α is given but also computed interactively. To ensure that the given and the computed values are equal, we may have to run the program more than once in order to obtain a usable replication. Five starting values for random number generators are also inputs. They are for the aggregate message arrival process, the message transmission time, message source station, determination of the outer-ring gateway destination queue for internetwork messages, and finally the determination of the messages class. Message arrival process, message transmission times, message class and source station queue are determined exactly as explained in chapter III. The determination of the outer-ring gateway destination queue for internetwork messages is obtained through discrete mappings of standard uniform variates. All that is necessary is to divide up the range of the uniform variate into $M-1$ portions, one for each possible destination outer-ring gateway queue. This is equivalent to attributing a zero width portion to the outer-ring gateway queue which is on the outer-ring from where the internetwork message is originated.

For the examples we consider, messages are assumed to be exponentially distributed with mean length 100 in units of the total propagation delay of the outer-ring. The walktime per station in the same units is $1/(N+1)$. The theoretical mean delay results are plotted against the total arrival rate ($\lambda_T = N.M. \lambda$ per unit of total propagation delay) for different values of N (number of stations on an outer-ring) and $M=4$, number of outer-rings in Figures 5.3-8. We should note that the three types of queues constitute three kinds of bottlenecks in the system. Our objective is to reduce the delay of inter-ring messages but at a small cost for local messages. To reach these objectives, the proportion of the internetwork traffic (α) is first chosen as .1, which is a reasonable value. Cases of 20 and 40 stations per outer-ring are considered. For $\alpha=.1$, the simulation results for the case of four outer-rings with 20 stations each are given in tables 5.1 and 5.2. Table 5.1 summarizes the simulated mean delays in an outer-ring and table 5.2 summarizes the total mean delays for inter-ring messages. For this value of α, outer-ring gateway queues are bottlenecks since they tend to saturate faster than outer-ring station queues. Inner-ring gateway queues are not saturated and the delay in the inner-ring appears negligible when outer-ring gateway queues are under heavy loading. At outer-ring station queues, the triangles and the circles on the corresponding figures show the simulation results for class 1 and class 2 messages, respectively. For the total mean delay of class 1 messages, the triangles on the figures show the simulation results for the case without priority while the crosses are for the priority case. For class 2 messages at outer-ring station queues, only simulation results for the priority case are shown. Because of the small value of α, the difference with the nonpriority case is not significant.

In Figures 5.3 and 5.5, the single solid line shows the mean message delay in an outer-ring, with class 1 messages given priority while the broken lines show the case without priority. As may be seen, the class 1 message delay has been reduced compared to the non-priority case while the increase in class 2 message delay is very little and not distinguishable from the broken lines, which is to be expected because of the small values of α. Since outer-ring station queues are not heavily loaded (compared to outer-ring gateway queues), the discrimination between priority classes is not subtantial. For 20 and 40 stations on an outer-ring, the outer-ring gateway queues carry, respectively, double and quadruple of the traffic on an outer-ring station. Consequently, with 40 stations, outer-ring gateway queues reach saturation faster than in the case of 20 stations; and the priority benefit with 40 stations per outer-ring is smaller than for the case of 20 stations.

Figures 5.4 and 5.6 show the end-to-end mean message delay for class 1 with and without priority. Again priority achieves more reduction in delay for the case of 20 stations. Even though outer-ring gateway queues saturate faster than outer-ring station queues, outer-ring stations dominate the ring because of their number. The approximation seems to overestimate the arrival rate to outer-ring gateway queues and therefore the delay in these queues. We notice that the computed values given by the analysis overbound the simulation. Similar results have been observed for an analysis of a star network using a similar analytical approach [59].

The solution in this case where outer-ring gateway queues are bottlenecks is to consider another service discipline at these queues. However, this may have a negative effect on outer-ring station queues in which messages (in both classes) may experience longer delay than before. For outer-ring station queues to become bottlenecks, the value of α should be small (less than $1/N$). This is illustrated by Figures 5.7 and 5.8 for a system of four outer-rings, each connecting 20 stations and for α=.02. Again because of the small portion of internetwork messages, local messages are almost not penalized by the priority mechanism and class 2 mean delays for the two cases are not distinguishable from each other. We can see that at heavy loading, mean delay for internetwork messages is substantially reduced to the same level as for local messages. The reason is that for this value of α outer-ring station queues saturate faster than the gateway queues, which enable us to emphasize the advantage of giving priority to internetwork traffic at an outer-ring station queue.

λ_T	Class 1 mean delay	Class 2 mean delay
.016	195 ± 16	198 ± 18
.02	203 ± 22	253 ± 27
.024	286 ± 31	349 ± 35
.03	497 ± 50	592 ± 63
.032	680 ± 74	815 ± 83
.034	898 ± 92	1306 ± 160
.035	1356 ± 168	2370 ± 253

Table 5.1 : Simulation mean delays in an outer-ring station queue.
4 outer-rings, 20 stations per outer-ring, α=.1.

λ_T	With priority	Without priority
.016	510 ± 42	513 ± 48
.02	590 ± 49	612 ± 57
.024	704 ± 54	793 ± 69
.030	1109 ± 93	1215 ± 130
.032	1512 ± 137	1640 ± 196
.034	3095 ± 322	3864 ± 428

Table 5.2 : Simulation total mean delays for inter-ring messages.
4 outer-rings, 20 stations per outer-ring, α=.1.

Remarks

It is clear that the outer-rings are not symmetric systems because of the presence of outer-ring gateway queues. Consequently, mean delays in outer-ring station queues should be affected by their position relative to the outer-ring gateway queue. For the simulation (see appendix C), we had to take the average delay over all outer-ring station queues. We did not attempt to give mean delay at each station. This was not useful because it was not possible to obtain from the mathematical analysis.

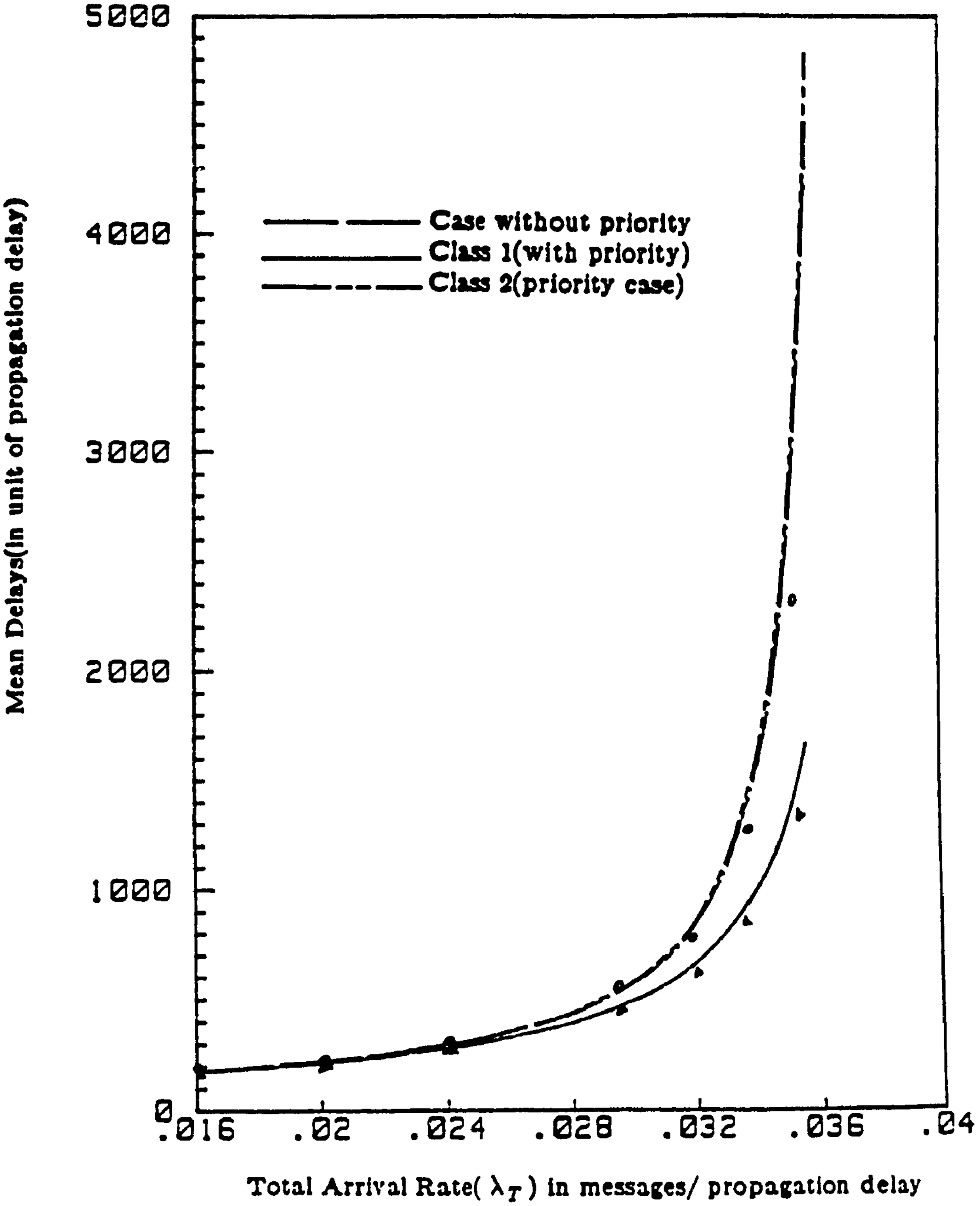


Fig.5.3: Class 1 and class 2 message mean delay in an outer-ring station queue versus total arrival rate
4 outer-rings
20 stations per outer-ring
α=.1

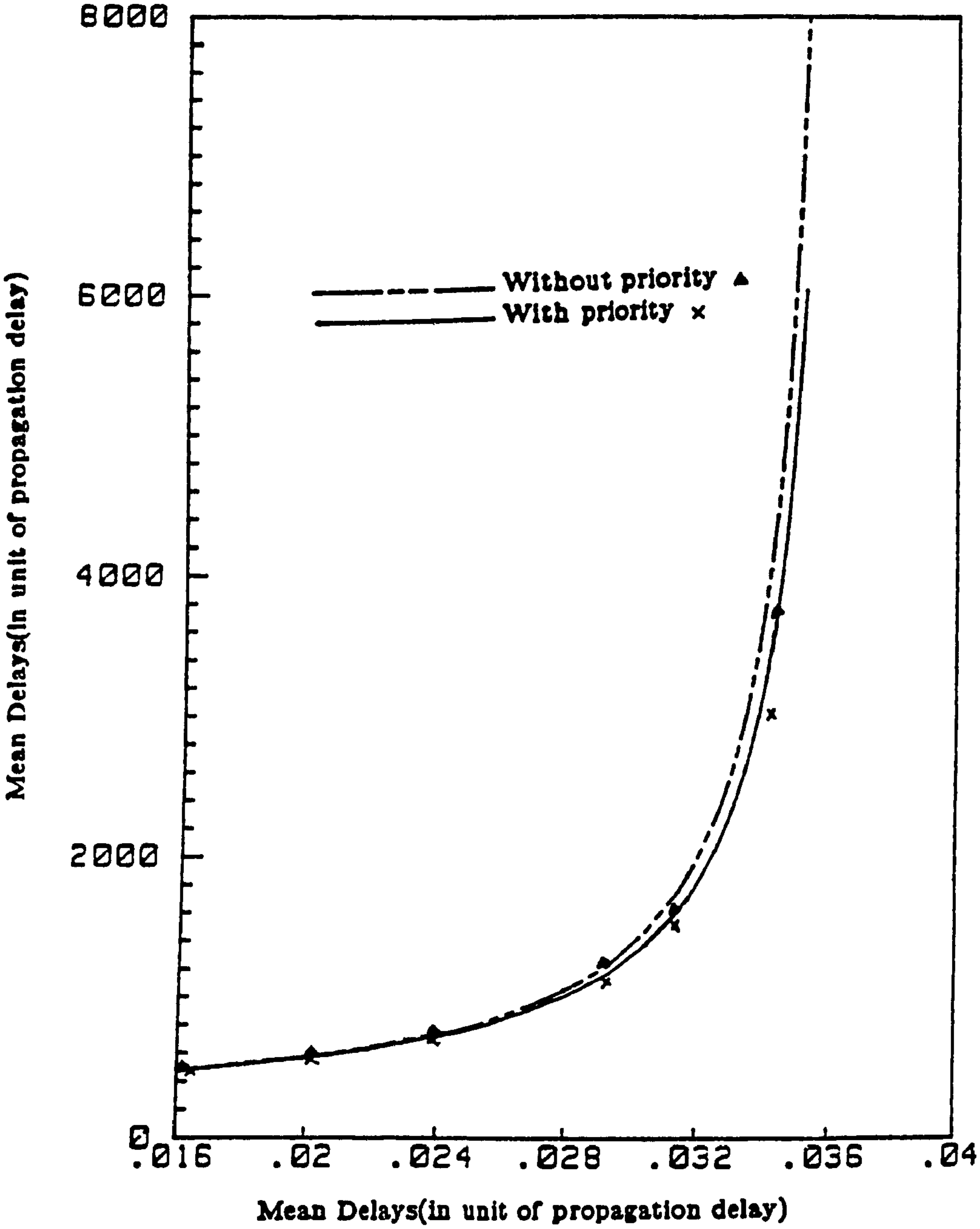


Fig.5.4: The total mean delay of class 1 message versus total arrival rate
(Class 1 is inter-ring message)
20 stations per outer-ring
σ=.1

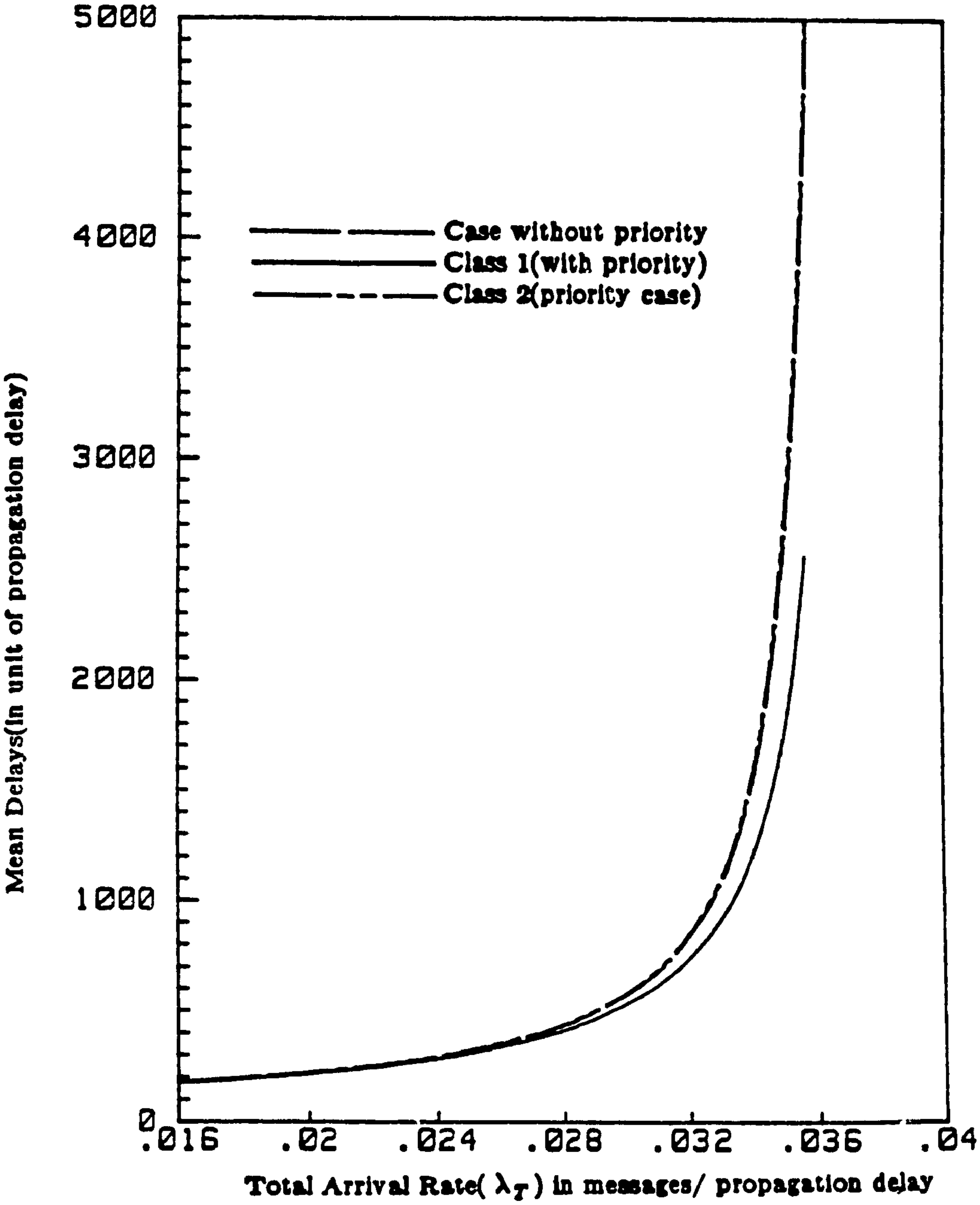


Fig.5.5: Class 1 and class 2 message mean delay in an outer-ring station queue versus total arrival rate
4 outer-rings
40 stations per outer-ring
α==.1

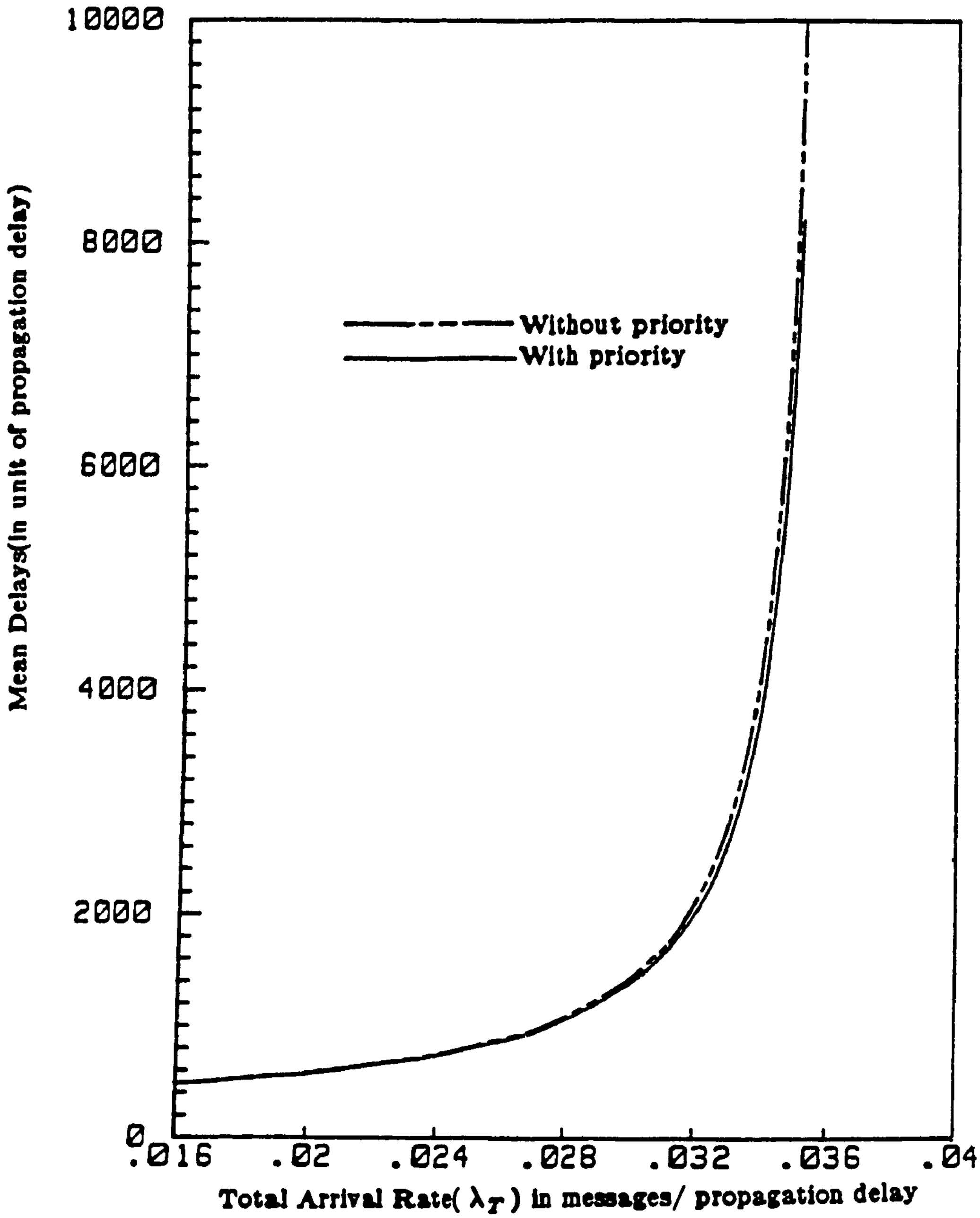


Fig.5.6: The total mean delay of class 1 message versus total arrival rate
(Class 1 is inter-ring message)
40 stations per outer-ring
α=.1

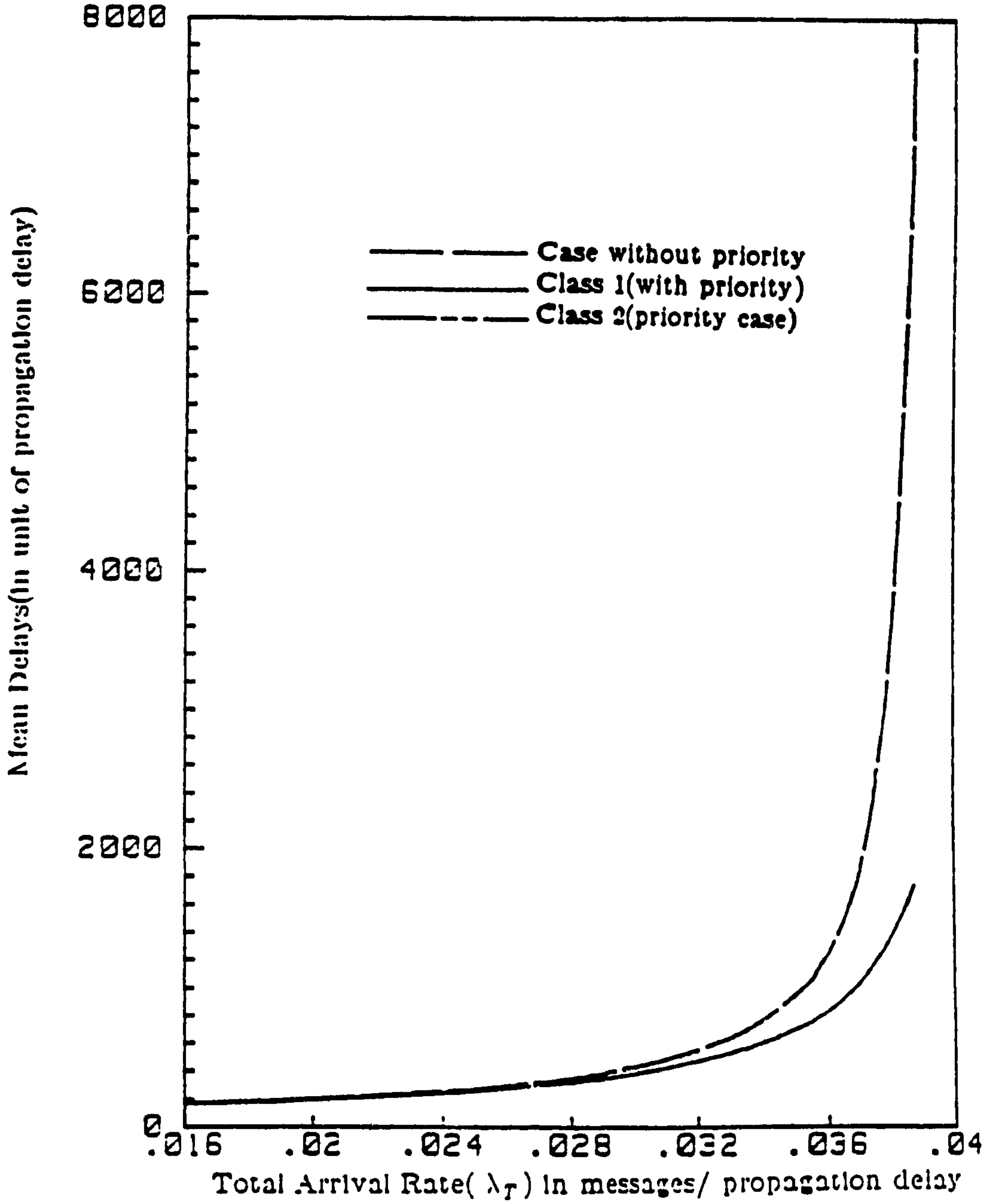


Fig.5.7: Class 1 and class 2 message mean delay in an outer-ring station queue versus total arrival rate
4 outer-rings
20 stations per outer-ring
α=.02

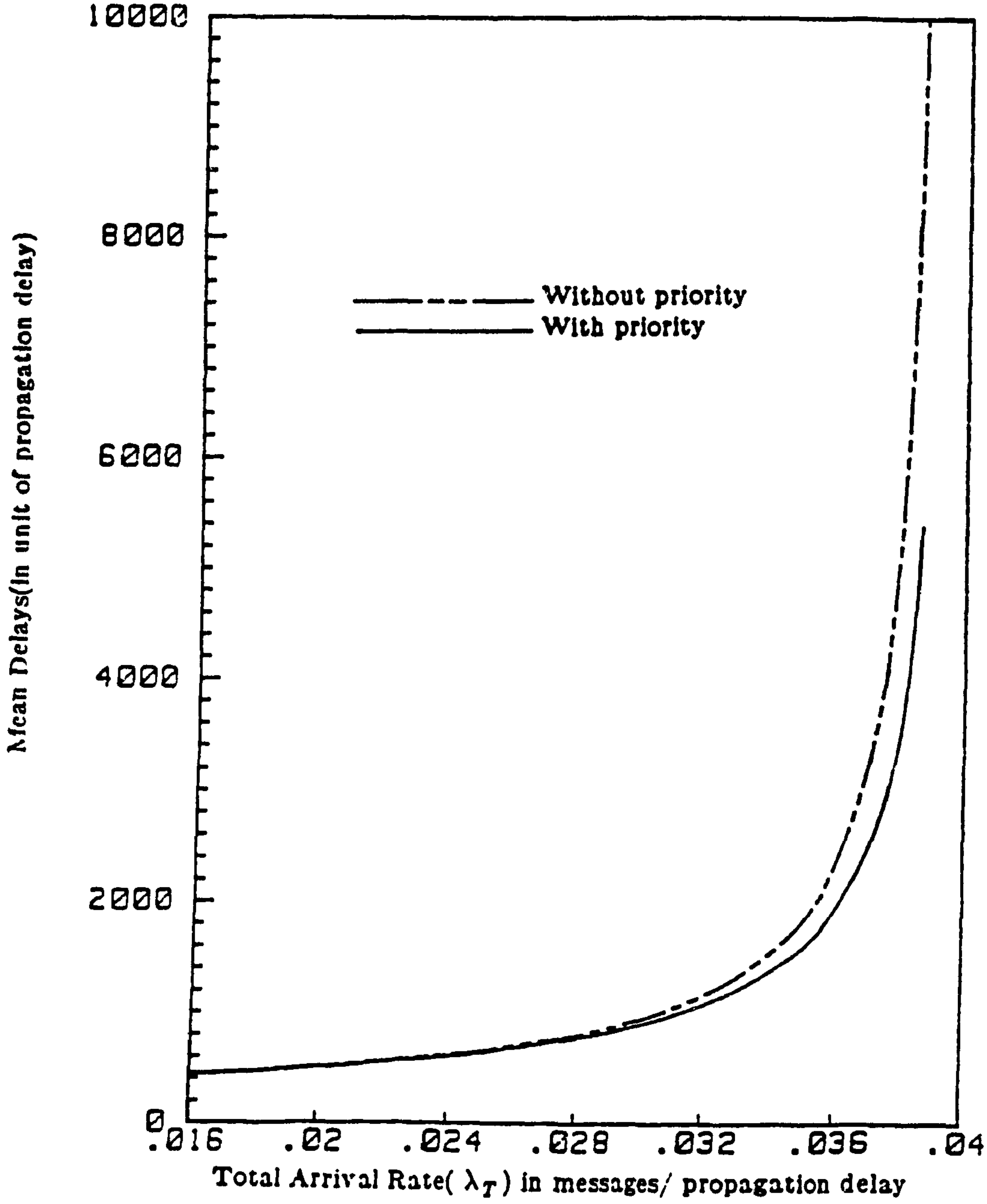


Fig.5.8: The total mean delay of class 1 message versus total arrival rate
(Class 1 is inter-ring message)
40 stations per outer-ring
α=.02

CHATER VI

CONCLUSIONS AND FUTURE WORK

This thesis has been concerned with the priority mechanism in token ring and the interconnection of token ring networks through gateways. The main difficulty with these issues arises from the mathematical complexity of polling systems in which the stations interconnected are strongly dependent.

The main contributions of the research described in this thesis have been to provide analytical solutions, even though approximate, to these important issues in LANs. There are still some open problems related the priority mechanism in LANs and their interconnection. The results presented in this thesis and related issues that may benefit from further investigations are summarized next.

In chapter III, an asymmetric prioritized token ring operation with nonpreemptive priority queueing and a limited-to-one service disciplines at each station was considered. In nonsymmetric cases, we found that message delay at station is affected by the order in which that station is polled and lower classes are more sensitive to the polling order effect. The problem remains since an exact solution even for the case of only two stations is not available. Other interesting variations that need further investigations are the gated and the exhaustive service disciplines. We think that the limited service discipline allows a moderate discrimination among priorities while the gated service offers a medium discrimination and the exhaustive service a severe delay difference between priorities. However, rigorous mathematical treatments are needed to quantify these levels of discrimination.

The results obtained from our model developed in chapter III were compared with some results reported in the literature in chapter IV. We observed that at low loading even lower classes in Gianini and Manfield's model (under exhaustive or gated service) perform better than all priorities in our model. Also at low traffic, all the schemes in the prioritized CSMA/CD give better delay. The most important result is that the mean delay for the lowest priority class, which can be easily calculated in our model, constitutes a lower bound for the average delay of the lowest priority class in G&M model. We affirm that the circumstance where the priority mechanism would result in a large and severe discrimination between priorities in G&M model is when service is limited to a single message per token. A systematic and rigorous study should prove interesting. In both models, the paradoxal fairness (in our model) and unfairness (in G&M model) of the limited-to-one service is necessary, but not sufficient, to gain "maximum desired benefit" from priority operation.

Finally in chapter V, the problem of the interconnection of LANs by a backbone network, both using token passing, was considered. Two general results were developed. The extensions of the prioritized M/G/1 and the G/G/1 to cases where a message initiating a busy period receives special service have general applicabilities. Recently, a network with star topology has been accurately modeled as an M/G/1 queue in which messages arriving at an empty queue receive service different than those which arrive at a busy queue. We think that the prioritized M/G/1 results may constitute a good approximation for the star augmented by a priority mechanism at the source queues. Since the service times as defined in [59] still hold, it is only a matter of substitution. It appears evident that allowing mixed service strategies on outer-rings, that is for the outer-ring gateway queues to transmit more than a single message per token will reduce one of the delay components of the inter-ring messages. Unfortunately, the pseudo-conservation law is not helpful because of the non-Poisson characteristic of the arrival process to outer-ring gateway queues. A generalization of the pseudo-conservation law to the case of general arrival process should prove very interesting. Rigorous mathematical treatments are needed to quantify and compare the reduction of delay in an outer-ring gateway queue and the increase of the delay in a station queue.

The advantage of the ring in allowing a large number of stations to be connected to the system is important in practical applications for the outer LANs. Since the number of the stations on the backbone network is usually not large, a backbone star will also be a viable alternative. Benefitting from the characteristics of a star topology, it will be possible for the backbone stations to transmit and receive simultaneously at the full bandwidth of the medium. We think that this would improve the performance of the system significantly when inner-ring gateway queues are bottlenecks.

REFERENCES

[1] A.R. Kaye, "Analysis of a Distributed loop for Data Trasmission", *Proceedings of the Symposium on Computer-Communications Networks and Teletraffic*, Polytechnic Institute of Brooklyn, April 1972.

[2] C. Mack, T. Murphy and N.L. Webb, "The Efficiency of N machines unidirectionally Patrolled by One Operative when Walking Time and Repair Times are constant," *Journal of the Royal Statistical Society*, Vol.19, No.1, 1957.

[3] C. Mack, "The Efficiency of N Machines Uni-directionally Patrolled by One Operative when Walking Time is constant and Repair Times are Variable," *Journal of the Royal Statistical Society*, Vol.19, No.1, 1957.

[4] T. Takine, Y. Takahashi and T. Hasegawa, "Exact Analysis of Asymmetric Polling Systems with Single Buffers," *to appear in IEEE Transaction on Communication.*

[5] H. Takagi, "Analysis of Polling Systems", MIT Press 1986.

[6] R.B. Cooper and G. Murray, "Queues Served in Cyclic Order," *Bell System Technical Journal*, Vol.48, No.3, 1969.

[7] R.B. Cooper, "Queues Served in Cyclic Order: Waiting Times," *Bell System Technical Journal*, Vol.49, No.3, 1970.

[8] M.A. Leibowitz, "An Approximate Method for Treating a Class of Multiqueue Problems," *IBM Journal of Research and Development*, Vol.5, No.3, 1961.

[9] O. Hashida, "Analysis of Multiqueues," *Review of the Electrical Communication Laboratories*, Vol.20, Nos.3-4, 1972.

[10] M. Eisenberg, "Queues with Periodic Service and Changeover Times," *Operations Research*, Vol.20, No.2, 1972.

[11] M.J. Ferguson and Y.J. Aminetzah, "Exact Results for Nonsymmetric Token Ring Systems," *IEEE Transactions on Communications*, Vol.Com-33, No.3, 1985.

[12] ANSI/IEEE, *802.5 Token Ring Access Method*, 1985

[13] J. Gianini and D.R. Manfield, "An Analysis of Symmetric Polling Systems with Two Priority Classes," *Performance Evaluation*, 1988.

[14] M. Murata and H. Takagi "Mean waiting times in nonpreemptive priority M/G/1 queues with switchover times, in: O.J.Boxma, J.W. Cohen and H.C.Tijms, eds., *Teletraffic Analysis and Computer Performance Evaluation* (North-Holland, Amsterdam, 1986)

[15] T. Nishida, M. Murata, H. Miyahara and K. Takashima, "Prioritized Token Passing Method in Ring-Shaped Local Area Networks", *IEEE International Conference on Communication 1983*, pp.D2.2.1-D2.2.6, Boston, Massachusetts, June 19-22, 1983.

[16] O.C.Ibe and X.Cheng, "Approximate Analysis of Asymmetric Single-Service Token-Passing Systems", *IEEE International Conference on Communication 1987*, pp.17.6.1-17.6.6, Seattle, Washington, June 7-10, 1987.

[17] Standars *ECMA -80, -81, -82, -89, -89, and 90,* European Computer Manafacturers Association.

[18] Standars *ISO 8802/2, 8802/3, 8802/4, 8802/5,* International Organization for Standardization.

[19] N.D. Farmer and E.E. Newhall, "An Experimental Distributed Switching System to Handle Bursty Traffic", *Proc. of the ACM Symposium on Problems of Optimization of Data Communication Systems,* Pine Mountain, GA., pp. 1-34, October 1969.

[20] J.R. Pierce, "Network for the Block Switching of Data", *Bell System Technical Journal,* Vol. 51, pp. 527-530, June 1972

[21] B.K. Penney and A.A. Baghdadi, "Survey of computer communication loop networks, Parts 1 and 2," *Comput. Commun.*, vol. 2, 1979.

[22] A. Hopper, "Data ring at Computer Laboratory, University of Cambridge, Computer Science Technology: Local area networking," Washington, DC, National Bureau of Standards, 1977.

[23] "Local communication controller description," IBM Publication, 1981.

[24] C.C. Kearnes and M.T. Liu, "A loop network for simultaneous transmission of variable length messages," in *Proc. 2nd Annual Symp. on Comput. Architecture,* Houston, TX, 1975.

[25] E.R. Hafner, Z. Nenandal, and M. Tschanz, "A digital loop communications system," *IEEE Trans. Commun., vol. COM-22, 1974.*

[26] J.F. Hayes, Modeling and Analysis of Computer Communications Networks, Plenum Press, 1984.

[27] W. Bux and M. Schlatter, "An Approximate Method for the Performance Analysis of Buffer Insertion Rings," *IEEE Trans. Commun., vol. COM-31, 1983.*

[28] F.E. Ross, "FDDI-fiber, farther, faster," in *Proc. IEEE Infocom '86,* Miami, FL, 1986.

[29] M.J. Johnson, " Reliability Mechanisms of the FDDI High Bandwidth Token Ring Protocol," *Comput. Networks and ISDN Systems,* Vol. 11, No. 2, pp. 121-131, 1986.

[30] N. Abramson, "The ALOHA system-Another alternative for computer communications," in *1970 Fall Joint Comp. Conf. AFIPS Conf. Proc.*, vol. 37.

[31] L.G. Roberts, "ALOHA Packets System with and without Capture", *Computer Communications Review, 1975.*

[32] W. Bux, "Local area subnetworks: A performance comparison", *IEEE Trans. Commun., vol. COM-29, 1981.*

[33] J.F. Hayes, "Local Distribution in Computer Communications", *IEEE Communications Magazine, 1981.*

[34] B. Avi-Itzhak, W.W. Maxwell and L.W. Miller, "Queueing with Alternating Priorities," *Operations Research,* Vol.13, No.2, March 1965.

[35] M.F. Neuts and M. Yadin, "The Transient Behavior of the Queue with Alternating Priorities, with Special Reference to the Waiting Times," Mimeo Series No.136, Department of Statistics, Purdue University, 1968.

[36] L. Takacks, "Two Queues Attended by a Single Server," *Operations Research,* Vol.16, No.3, 1968.

[37] J.S. Syes, "Simplified Analysis of an Alternating-Priority Queueing Model with Setup Times," *Operations Research,* Vol.18, No.6, 1970.

[38] M. Eisenberg, "Two Queues with Changeover Times," *Operations Research,* Vol.19, No.2, 1971.

[39] J.F. Hayes and D.N. Sherman, "A Study of Data Multipexing Techniques and Delay Performance," *Bell System Technical Journal,* 51, pp. 1985-2011,

November 1972.

[40] M.A. Liebowitz, "A note on Some Fundamental Parameters of Multiqueue Systems," *IBM Journal of Research and Development,* Vol.6, No.4, 1962.

[41] M. Eisenberg, "Two Queues with Alternating Service," *SIAM Journal on Applied Mathematics,* Vol.36, No.2, 1979.

[42] O.J. Boxma, "Two Symmetric Queues with Alternating Service and Switching Times," Performance 84.

[43] O. Hashida and K. Ohara, "Line Accommodation Capacity of a Communication Control Unit," *Review of The Electrical Communication Laboratories,* Vol.20, Nos.3-4, 1972.

[44] O.J. Boxma and B. Meister, "Waiting Times in Multi-Queue Systems with Cyclic Service," *IBM Research Report* No.RZ1348, 1984.

[45] O.J. Boxma and B. Meister, "Waiting Times Approximations for Cyclic Service Systems with Switch-Over Times", Report OS-R8510, Centre for Mathematics and Computer Science, 1985.

[46] A.G. Konhein, "Chaining in a loop System," *IEEE Transactions on Communications,* Vol. COM-24, No. 2, pp.203-210, Feb. 1976.

[47] K.S. Watson, (1984) Performance evaluation of cyclic service strategies- a survey. In *Performance '84,* ed. E. Gelenbe, North-Holland, Amsterdam, pp.521-533.

[48] O.J. Boxma and W.P. Groenendijk, "Pseudo-conservation laws in Cyclic-Service Systems," *J. Appl. Prob.,* 24, pp. 949-964, 1987.

[49] L. Kleinrock, Queueing Systems, Vol. 2. Wiley, New York.

[50] W. Bux, "Performance issues in local-area networks," *IBM Syst. J.,* vol.23, No. 4, pp. 351-374, 1984.

[51] W. Bux, "Local-area subnetworks: A performance comparison," *IEEE Trans. Commun.,* Vol. COM-29, pp. 1465-1473, 1981.

[52] W. Bux and D. Grillo, "Flow Control in Local Area Networks of Interconnected Token Rings," *IEEE Trans. Commun.,* Vol. COM-33, October 1985.

[53] O.C. Ibe and X. Cheng, "Stability conditions for polling systems" *Internal Report at Georgia Institute of Technology.*

[54] G.B. Swartz, "Polling in a loop system," *J. ACM,* vol.27, pp.42-59, 1980.

[55] B. Grela-M'poko, M. Mehmet-Ali and J.F. Hayes, "Extensions to the Exceptional Service Model", under review.

[56] J.F. Hayes and D.N. Sherman, "A study of Data Multiplexing Techniques and Delay Performance", *Bell System Technical Journal,* Vol.51, November 1972.

[57] L. Kleinrock, Queueing Systems ,volume 1 : Theory, John Wiley & Sons, 1975.

[58] P.D. Welch, " On a Generalized M/G/1 Queueing Process in Which the First Customer in a Busy Period Receives Exceptional Service ", *Operations Research,* 12, 736-752, 1964.

[59] M. Mehmet-Ali, J.F. Hayes and A. Elhakeem, "Traffic Analysis of a Local Area Network with a Star Topology", *IEEE Transaction on Communications,* June 1983.

[60] R.M.Metcalfe and D.R.Boggs, "Ethernet: Distributed packet switching for local area networks," *Commun. Ass. Comput. Mach.,* vol. 19, July 1976.

[61] L. Kleinrock anu F.A. Tobagi, "Carrier Sense Multiple Access for Packet Switched Radio Channels," *Proceedings of the IEEE Inter. Conf. on Comm.*, Minneapolis, Minnesota, June 1979.

[62] F.A. Tobagi and V.B. Hunt, "Performance Analysis of Carrier Sense Multiple Access with Collision Detection," *Computer Networks*, vol. 4, Oct./Nov. 1980.

[63] S.S. Lam, "A carrier sense multiple access protocol for local networks," *Computer Networks*, Vol. 4, Feb. 1980.

[64] P.T.R. Wang and M.F. McGurrin, "CSMA/CD Lans With Overload Detection and Protection," *Computer Networks and ISDN Sistems*, Vol. 11, Feb. 1986.

[65] F.A.Tobagi, F.Borgonovo and L.Fratta "Expessnet: A High-Performance Integrated-Services Local area Network," *IEEE Journal on Selected Areas in Communications*, Vol.SAC-1, November 1983.

[66] J.O.Limb and C.Flores, " Description of Fasnet-A unidirectional local-area com,munications network," *Bell Syst. Tech. J.*, Vol. 61, Sept. 1982.

[67] R.V.Schmidt, E.G.Rawson, R.E.Norton, S.B.Jackson and M.D.Bailey, " Fibernet II: A fiber optic ethernet," *IEEE J. Select. Areas Commun.*, vol. SAC-1, Nov.1983.

[68] E.S. Lee and P.I.P. Boulton, "The principles and performance of Hubnet: A 50 Mbits/s glass fiber local area network," *IEEE J. Select. Areas Commun.*, vol. SAC-1, Nov. 1983.

[69] D.T.W. Sye, "A metiopolitan area network, "Special Issue on Fiber Optics for Local Communications," *IEEE J. Select. Areas Commun.*, vol. SAC-3, Nov. 1985.

[70] G.L. Chaoudhury and S.S. Rappaport, "Priority Access Schemes Using CSMA-CD," *IEEE Transactions on Communications*, vol. COM-33, 1985.

[71] D.R. Manfield, "Analysis of a priority polling system for two-way traffic," *IEEE Transactions on Communications*, vol. COM-33, 1985.

[72] D.R. Cox and W.L. Smith, *Queues*, Methuen, New York, 1961

APPENDIX A

NOTATION

l_i Number of priority classes at station i

λ_{il} Arrival rate of class l messages at station i , $l=1,2,..,l_i$.

$\lambda_i = \sum_{l=1}^{l_i} \lambda_{il}$ Total arrival rate at station i.

$M_{il}(s)$ Laplace-Stieltjes Transform (LST) of the pdf of a class l message transmission time at queue i.

m_{il} Class l mean message transmission time at queue i.

$m_{il}^{(2)}$ Second moment of a class l message at station i.

mo_{il} Mean residual life of a class l message at station i.

$R_i(s)$ LST of the pdf of the reply interval (walktime) at queue i.

r_i Mean walktime at station i.

$r_i^{(2)}$ Second moment of the walktime at station i.

ro_i Mean residual life of the polling time at station i.

Pm_{jl} Probability that the server is serving a class l at station j when the tagged message arrives.

Pr_j Probability that the server is polling station j when the tagged message arrives.

V_{jil} Passage time from station j to station i, given that the tagged message arrived when the server was serving a class l message at station j. This time is measured from the time service to that class l message at station j ends.

V_{ji} Passage time from station j to station i, given that the tagged message arrived when the server was walking to station j. This time is measured from the time the walktime to station j ends.

U_{il} The intervisit time for station i, given that a class l message is served at station i.

N_{ij_c} Number of class j_c messages seen by the class j_c tagged message at station i when it arrives.

Nw_{il} Number of higher priority messages which arrive during the waiting time of the class j_c tagged message ($l=1,2,\ldots,j_c-1$).

N_{il} Number of higher class messages seen by the class j_c tagged message at station i when it arrives ($l=1,2,\ldots,j_c-1$).

ρ_s Probability that an outer-ring station queue is busy (its utilization factor).

ρ_I Probability that an inner-ring gateway queue is busy (its utilization factor).

ρ_o Probability that an outer-ring gateway queue is busy (its utilization factor).

$Y_{sj}(s)$ LST of the pdf of the service time of a class j message arriving at a busy station.

$\bar{y}_{sj}$ Mean class j message service time when arrived at a busy station.

$\overline{y_{sj}^{(2)}}$ Second moment of class j message when arrived at a busy station.

$\hat{Y}_{sj}(s)$ LST of the pdf of the service time of a class j message arriving at an idle station.

$\bar{\hat{y}}_{sj}$ Mean class j message service time when arrived at a busy station.

$\overline{\hat{y}_{sj}^{(2)}}$ Second moment of class j message when arrived at a busy station.

$Y_o(s)$ LST of the pdf of the service time of a message arriving at a busy outer-ring gateway queue.

$\bar{y}_o$ Mean message service time when arrived at a busy outer-ring gateway queue.

$\overline{y_o^{(2)}}$ Second moment of a message service time when arrived at a busy outer-ring gateway queue.

$\sigma^2_{y_o}$ Variance of a message service time when arrived at a busy outer-ring gateway queue.

$\hat{Y}_o(s)$ LST of the pdf of the service time of a message arriving at an idle outer-ring gateway queue.

$\bar{\hat{y}}_o$ Mean message service time when arrived at a busy outer-ring gateway queue.

$\overline{\hat{y}_o^{(2)}}$ Second moment of a message service time when arrived at a busy outer-ring gateway queue.

$\sigma^2_{\hat{y}_o}$ Variance of a message service time when arrived at a busy outer-ring gateway queue.

$Y_I(s)$ LST of the pdf of the service time of a message arriving at a busy inner-ring gateway queue.

$\bar{y}_I$ Mean message service time when arrived at a busy inner-ring gateway queue.

$\overline{y_I^{(2)}}$ Second moment of a message service time when arrived at a busy inner-ring gateway queue.

$\sigma^2_{y_I}$ Variance of a message service time when arrived at a busy inner-ring gateway queue.

$\hat{Y}_I(s)$ LST of the pdf of the service time of a message arriving at an idle inner-ring gateway queue.

$\bar{\hat{y}}_I$ Mean message service time when arrived at a busy inner-ring gateway queue.

$\overline{\hat{y}_I^{(2)}}$ Second moment of a message service time when arrived at a busy inner-ring gateway queue.

$\sigma^2_{\hat{y}_I}$ Variance of a message service time when arrived at a busy inner-ring gateway queue.

$A_o(s)$ LST of the pdf of the interarrival time to an outer-ring gateway queue.

$\bar{t}_o$ Mean messages interarrival to an outer-ring gateway queue.

$\sigma^2_{t_o}$ Variance of the interarrival time at an outer-ring gateway queue.

$A_I(s)$ LST of the pdf of the interarrival time to an inner-ring gateway queue.

$\bar{t}_I$ Mean messages interarrival to an inner-ring gateway queue.

$\sigma^2_{t_I}$ Variance of the interarrival time at an inner-ring gateway queue.

$W_o(s)$ LST of the pdf of the waiting time in an outer-ring gateway queue.

$W_I(s)$ LST of the pdf of the waiting time in an inner-ring gateway queue.

$I_o(s)$ LST of the pdf of an outer-ring gateway queue idle period.

$\bar{I}_o$ Mean outer-ring gateway queue idle period.

$\overline{I_o^{(2)}}$ Second moment of an outer-ring gateway queue idle period.

$I_I(s)$ LST of the pdf of an inner-ring gateway queue idle period.

$\bar{I}_I$ Mean inner-ring gateway queue idle period.

$\overline{I_I^{(2)}}$ Second moment of an inner-ring gateway queue idle period.

APPENDIX B

CONFIDENCE INTERVAL AND MAXIMUM ERROR OF ESTIMATE

For each simulation conduced in this thesis, the method of replicating the simulation experiment is used for analyzing the data. The experiment is run many times using different (and presumely independent) streams of random numbers for different runs.

Suppose that we replicate the experiment m times, and each replication generates n observations,

$d_i^{(j)}$; $1 \le i \le n$, $1 \le j \le m$.

The number n corresponds to the total number of messages generated and $d_i^{(j)}$ is the delay of the ith message in the jth experiment. Thus, we have the total of $n^* = mn$ usable observations. We denote the sample mean delay of the j replication by

$$\overline{d^{(j)}} = \frac{\sum_{i=1}^{n} d_i^{(j)}}{n}, \quad j = 1,2,\ldots,n$$

Then, the set $\{\overline{d^{(j)}};\ 1 \le j \le m\}$ forms m independent and indentically distributed random variables. The best estimates of the mean and variance σ_d^2 of the variables $\overline{d^{(j)}}$'s are given by the overall sample mean delay

$$\overline{d} = \frac{1}{m}\sum_{i=1}^{m} \overline{d^{(j)}} = \frac{1}{nm}\sum_{j=1}^{m}\sum_{i=1}^{n} d_i^{(j)}$$

and the sample variance

$$s_d^2 = \frac{1}{m-1}\sum_{i=1}^{m}(\overline{d^{(j)}} - \overline{d})^2,$$

respectively. Since the number messages generated is sufficiently large, the variables $\overline{d^{(j)}}$'s are approximately normally distributed. Then, the distribution of the variable

$$\bar{d} - mean.(s_d{}^2/m)^{1/2}$$

can be approximated by the t-distribution with $m-1$ degrees of freedom. Thus, the confidence interval for the mean is given by

$$\bar{d} \pm E$$

where $E = \frac{t_{\alpha/2;m-1}}{m^{1/2}}.s_d$ is the maximum error of estimate.

APPENDIX C

PROGRAM ERIC-MAUREEN

```
c***************************************************************
c************ PRIORITIZED TOKEN- RING *****************
c***************************************************************
                                program Eric-Maureen
        program priority
        integer qs(2,50),qst(50),nar(2,50),nars(50),nofd(2,50)
        integer iseed(4),xy1,xy2,zy1,zy2,c,zt,tnq,not,nq,narv
        integer nofdt,qse,narcl(2),nofdcl(2)
        real sum3,sum,su(2)
        real gavdel(2,50),v(2),N(2),u1,u2,r1,r2,wt,sum1,sum2
        real st,at,x1,x2,z
        real cat(2,50,900),avdel(2,50),del(2,50)
        real alfa,alfa1,sdel(2),adel(2),avwt(2)
c************** nq : number of queues in the system**************
        print*,'how many queues(nq)?'
        print*,'nq=?'
        read*,nq
c***** Generation of starting values for random numbers generators**
c***************1. arrival time of messages
c***************2. message class
c***************3. message transmission time
c***************4. message source queue
c
        do 15 i=1,4
        print*,'iseed(',i,')=?'
        read*,iseed(i)
15      continue
c **************Generation of arrival rate v(1) and ****************
c **************v(2)(the transmission rate) ***********************
        do 16 i1=1,2
        print*,'v(',i1,')=?'
        read*,v(i1)
16      continue
c*********** not : minimum number of customers to be generated *******
c*********** not : minimum number of customers to be generated *******
c**************** wt : walktime *********************
        print*,'alfa=?'
        read*,alfa
        print*,'not=?'
        read*,not
        print*,'notsup=?'
        read*,notsup
        wt=(1./float(nq))
        write(10,106) v(1),v(2),alfa,wt,not,notsup
```

```
106          format(' ',5x,'v(1)=',f11.5,'v(2)=',f11.5,'alfa=',f11.5
        &  ,'wt=',f11.5,'not=',i11,'notsup=',i11)
c qse    : system size
c nar(i,j) : number of class i arrivals to queue j
c qs(i,j)  : number of class i messages in queue j
c qst(j)    : number of messages in queue j
c nars(j)   : number of arrivals to queue j
c narcl(i) :number of class i arrivals to the system
c nofdcl(i) : number of class i departures from the system
c cat(i,j,l) : the lth class i arrival time at queue j
c cdt(i,j,l) : the lth class i departure time from queue j
c nofd(i,j): number of class i departures from queue j
c narv      : number of arrivals to the system
c N(1)      : next arrival time
c N(2)      : next departure time
c tnow      : time now
c ******************Initialization**********************
             tnq=nq+1
             kb=0
             narv=0
             nofdt=0
             qse=0
             t=0.
             tnow=0.
             sum1=0.
             sum2=0.
             sum3=0.
             sum=0.
             servt=0.
             trfic=0.
             do 19 j1=1,2
             sdel(j1)=0.
             narcl(j1)=0
             nofdcl(j1)=0
             su(j1)=0.
             do 191 j2=1,nq
             qst(j2)=0
             nars(j2)=0
             qs(j1,j2)=0
             nar(j1,j2)=0
             nofd(j1,j2)=0
             gavdel(j1,j2)=0.

 191         continue
 19          continue

             N(2)=10**37
c************* First there is an arrival ****************
             xy1=irand(iseed(1))
             iseed(1)=xy1
             x1=float(xy1)/2147483647.0
             at=-(1.0/v(1))*alog(1-x1)
             t=t+at
```

```
            N(1)=tnow+t
c
100         if(N(1).le.N(2)) then

            zy4=irand(iseed(4))
            iseed(4)=zy4
            u2=float(zy4)/2147483647.0
            do 21 j20=1,nq
            r2=float(j20)/float(nq)
            if(u2.le.r2) then
            goto 67
            else
            endif

21          continue
67          k=j20

            qnb=k
            zy2=irand(iseed(2))
            iseed(2)=zy2
            u1=float(zy2)/2147483647.0
            if(alfa.ge.u1) then
            c=1
            else
            c=2
            endif
            sum=sum+at
            goto 110
            else
            goto 120
            endif
c************ARRIVALS** ARRIVALS** ARRIVALS*****************
110         tnow=N(1)

            qse=qse+1
            narv=narv+1
            qst(k)=qst(k)+1
            qs(c,k)=qs(c,k)+1
            nar(c,k)=nar(c,k)+1
            narcl(c)=narcl(c)+1
            alfa1=float(narcl(1))/float(narv)
            nars(k)=nars(k)+1
            cat(c,k,qs(c,k))=tnow
            if(narv.eq.1) then
            goto 120
            else
            endif

105         if(alfa1.eq.alfa.and.narv.ge.not.and.narv.le.notsup
          & .or.narv.eq.notsup) then

            N(1)=10**37
            else
          kk=0
          xy1=irand(iseed(1))
          iseed(1)=xy1
          x1=float(xy1)/2147483647.0
          at=-(1.0/v(1))*alog(1-x1)
          N(1)=N(1)+at
          endif
          goto 100
c **********DEPARTURE**DEPARTURE**DEPARTURE*************
120       kb=kb+1
          if(narv.eq.1.and.kb.eq.1) then
          j=qnb
          else
          endif
          if(qst(j).ge.1) then
          if(qs(1,j).ge.1) then
          i=1
          goto 102
          else
          if(qs(2,j).ge.1) then
          i=2
          goto 102
          else
          endif
          endif
          else
          j=j+1
          N(2)=N(2)+wt
          if(j.eq.tnq) then
          j=1
          else
          endif
          goto 100
          endif
102       kk=1
          zt=irand(iseed(3))
          iseed(3)=zt
          z=float(zt)/2147483647.0
          st=-(1.0/v(2))*alog(1-z)
          sum3=sum3+st
          if(i.eq.1) then
          su(1)=su(1)+st
          else
          su(2)=su(2)+st
          endif
          if(narv.eq.1.and.kb.eq.1) then
          N(2)=tnow+st
          else
          N(2)=N(2)+st
          endif
          tnow=N(2)
          nofd(i,j)=nofd(i,j)+1
          nofdcl(i)=nofdcl(i)+1
```

```
      nofdt=nofdt+1
      avdel(i,j)=tnow-cat(i,j,1)
      gavdel(i,j)=gavdel(i,j)+avdel(i,j)
      qs(i,j)=qs(i,j)-1
      qse=qse-1
      qst(j)=qst(j)-1
      if(qs(i,j).ge.1) then
      do 88 j88=1,qs(i,j)
      j89=j88+1
      cat(i,j,j88)=cat(i,j,j89)
88    continue
      else
      endif
      if(alfa1.eq.alfa.and.narv.ge.not.
     & and.narv.eq.nofdt.and.narv.le.notsup.
     & or.narv.eq.notsup.and.nofdt.eq.narv) then
      goto 222
      else
      j=j+1
      if(j.eq.tnq) then
      j=1
      else
      endif
      N(2)=N(2)+wt
      if(narv.eq.1) then
      goto 105
      else
      goto 100
      endif
      endif
222   do 300 i=1,2
      do 301 j=1,nq
      del(i,j)=gavdel(i,j)/nofd(i,j)
      sdel(i)=sdel(i)+del(i,j)
301   continue
      avint=sum/narv
      arate=1./avint
      avservt1=su(1)/nofdcl(1)
      avservt2=su(2)/nofdcl(2)
      avservt=alfa1*avservt1+(1.-alfa1)*avservt2
      adel(i)=sdel(i)/nq
      write(10,310) i,adel(i),nofdcl(i),narcl(i),avservt,arate,alfa1
310   format(' ',1x,'i=',i11,2x,'adel(i)=',f16.5,
     & 'nofdcl(i)=',i11,'narcl(i)=',i11,'avservt=',f11.5,
     & 'arate=',f11.5,'alfa1=',f11.5)
300   continue
      stop
      end
```

APPENDIX D

PROGRAM ANAIS

```
              c*************************************************************
              c************        INTERCONNECTION        OF        RINGS
              ********************
              c*************************************************************
                                          program anais
              integer qs(2,5,50),nar(2,5,50),nofd(2,5,50),qst(5,50),iseed(5),narv(5)
              integer qse(5),qp(50),kb(5),qnb(5),narcl(2)
              integer c,nofdpt,jd,kgo,kgi,ndpi,ndpo,ndps(2)
              integer zy,xv,nq,nrgs,systze,narvis,tnq
              integer not,c0,c3,c5,zt,xy
              integer ir,c01,c8,notsup
              real alfa,wt,x,z,st,t,u,r1,at,alfa1
              real v(2),N(2),tnow,D(5),cat(2,5,50,500)
              real delsc(2),delgo,delgi,dc1,dc2,dgi,dgo
              real Na(1),Nago(5),Nagi(5),avdel(2,5,50)
              real gavdel(2,5,50)
c*****nq is the number of outer-ring station queues in the system*****
c*************************************************************
              print*,'how many station queues(nq)?'
              print*,'nq=?'
              read*,nq
c****************************************************
c iseed(.) For generation of random numbers for
c   (1) arrival time of a message
c   (2) transmission time of a message
c   (3) message source queue
c   (4) for the destination queue(inner-ring messages)
c   (5) for message class
c*****************************************************
              do 15 i=1,5
              print*,'iseed(',i,')=?'
              read*,iseed(i)
15            continue
c**************************************************************
c***Enter the arrival rate v(1) and the transmission rate v(2)***
c**************************************************************
              do 16 i1=1,2
              print*,'v(',i1,')=?'
              read*,v(i1)
 16           continue
c*************************************************************
c*******not is the minimum number of messages to simulate********
c*************************************************************
```

```
          print*,'not=?'
          read*,not
c**************************************************************
c******notsup is the maximum number of messages to simulate******
c**************************************************************
          print*,'notsup=?'
          read*,notsup
          c***************nrgs is the number of outer-rings*******
          print*,'nrgs=?'
          read*,nrgs
c****************Given portion of class 1 messages****************
print*,'alfa=?'
read*,alfa
wt=(1./((float(nq)/float(nrgs))+1))
c************************************************************
c***  N(1)       : next arrival time
c***  N(2)       : next departure time
c***  t          : time
c***  tnow       : time now
c***  narvls     : number of arrivals to the system
c***  systze     : system size
c***  alfa1      : computed value of α
c***  nar(l,j,k) : number of class l arrivals in queue k of outer-ring j
c***  qs(l,j,k)  : number of class l messages in queue k of outer-ring j
c***  nar(k)     : number of arrivals at ring k
c***  qst(j,k)   : number of messages in queue j of ring k
c**************************************************************
c**********************INITIALISATION************************
c**************************************************************
          delgo=0.
          delgl=0.
          ndpl=0
          ndpo=0
          tnow=0.
          narvls=0
          nofdpt=0
          systze=0
          t=0.
          c0=nq/nrgs
          lr=nrgs+1
          c01=c0+1
          do 113 lc=1,2
          ndps(lc)=0
          narcl(lc)=0
          delsc(lc)=0.
          do 11 l1=1,lr
          kb(l1)=0
          narv(l1)=0
          qse(l1)=0
          if(l1.eq.lr) then
          nqs=nrgs
          else
          nqs=c01
          endif
          do 19 j1=1,nqs
          qst(i1,j1)=0
          nar(ic,i1,j1)=0
          nofd(ic,i1,j1)=0
          qs(ic,i1,j1)=0
          D(i1)=10**37
          Nago(i1)=D(i1)
          Nagi(i1)=D(i1)
          gavdel(ic,i1,j1)=0.
19        continue
11        continue
113       continue
c******************************************************************
c******************************************************************
c***************First there is an arrival************************
c******************************************************************
c******************************************************************
          kk=0
          xy=irand(iseed(1))
          iseed(1)=xy
          x=float(xy)/2147483647.0
          at=-(1.0/v(1))*alog(1-x)
          t=t+at
          Na(1)=tnow+t

100       N(1)=Na(1)
          do 91 j=1,nrgs
          if(Nago(j).le.N(1)) then
          N(1)=Nago(j)
          kgo=j
          else
          endif
91        continue

          do 92 ji=1,nrgs
          if(Nagi(ji).le.N(1)) then
          N(1)=Nagi(ji)
          kgi=ji
          else
          endif
92        continue

          N(2)=D(1)
          do 93 jd=1,ir
          if(D(jd).le.N(2)) then
          N(2)=D(jd)
          kjd=jd
          else
          endif
```

```
93        continue
          if(N(1).le.N(2)) then
          if(N(1).eq.Na(1)) then
c*************Determination of the message class*********************
          zy2=irand(iseed(5))
          iseed(5)=zy2
          u1=float(zy2)/2147483647.0
          if(u1.le.alfa) then
          c=1
          else
          c=2
          endif
c**********Determination of the message source queue***************
          zy=irand(iseed(3))
          iseed(3)=zy
          u=float(zy)/2147483647.0
          do 20 i20=1,nq
          r1=float(i20)/float(nq)
          if(u.lt.r1) then
          goto 66
          else
          endif
20        continue
66        kkk=i20
c****************Determination of the outer-ring*********************
          do 1002 j5=1,nrgs
          c3=j5*c0
          if(kkk.le.c3) then
          goto 1003
          else
          endif
1002      continue
1003      nra=j5
c************************nra is the outer-ring************************
          ka=kkk-((j5-1)*c0)
          if(narv(nra).eq.0.and.kb(nra).eq.0) then
          qnb(nra)=ka
          else
          endif
c***********************ka is the station source queue*****************
          goto 110

          else

          if(N(1).eq.Nagl(kgl)) then
          ka=kgl
          nra=ir
          c=1
          goto 111
          else
          if(N(1).eq.Nago(kgo)) then
          ka=c01
          nra=kgo
          c=1
          goto 112
          else
          endif
          endif
          endif

          else
          j3=kjd
          goto 120
          endif
c***********************************************************
c*******************FOR ARRIVALS*************************
c***********************************************************
110       tnow=N(1)

          narcl(c)=narcl(c)+1
          narvis=narvis+1
          sysize=sysize+1
          alfa1=float(narcl(1))/float(narvis)
          nar(c,nra,ka)=nar(c,nra,ka)+1
          qs(c,nra,ka)=qs(c,nra,ka)+1
          narv(nra)=narv(nra)+1
          qse(nra)=qse(nra)+1
          qst(nra,ka)=qst(nra,ka)+1
          c8=qs(c,nra,ka)
          cat(c,nra,ka,c8)=tnow

c***********If it is the first arrival at that outer-ring,***********
c***********we have to schedule its departure******************
          if(narv(nra).eq.1.and.kb(nra).eq.0) then
          j3=nra
          goto 120
          else
          endif

          if(alfa1.eq.alfa.and.narvis.ge.not.and.narvis.le.notsup.
        & or.narvis.eq.notsup) then
          Na(1)=10**37
          else
          xy=irand(iseed(1))
          iseed(1)=xy
          x=float(xy)/2147483647.0
          at=-(1.0/v(1))*alog(1-x)
          Na(1)=Na(1)+at
          endif
          goto 100
c***********************************************************
```

```
111         tnow=N(1)

            nar(c,nra,ka)=nar(c,nra,ka)+1
            qs(c,nra,ka)=qs(c,nra,ka)+1
            narv(nra)=narv(nra)+1
            qse(nra)=qse(nra)+1
            qst(nra,ka)=qst(nra,ka)+1
            c8=qs(c,nra,ka)
            cat(c,nra,ka,c8)=tnow
c************If it is the first arrival to the inner-ring,************
c************we have to schedule its departure.*****************
            if(narv(nra).eq.1.and.kb(nra).eq.0) then
            j3=nra
            qnb(j3)=ka
            goto 120
            else
            endif
            Nagi(ka)=10**37
            goto 100
c**********************************************************
112         tnow=N(1)

            nar(c,nra,ka)=nar(c,nra,ka)+1
            qs(c,nra,ka)=qs(c,nra,ka)+1
            narv(nra)=narv(nra)+1
            qse(nra)=qse(nra)+1
            qst(nra,ka)=qst(nra,ka)+1
            c8=qs(c,nra,ka)
            cat(c,nra,ka,c8)=tnow
            Nago(nra)=10**37
            goto 100
c**********************************************************
c*****************FOR DEPARTURES*********************
c**********************************************************
120         kb(j3)=kb(j3)+1
            if(j3.eq.ir) then
            tnq=nrgs+1
            goto 210
            else
            tnq=c0+2
            goto 209
            endif

209         c5=narv(j3)
            if(c5.eq.1.and.kb(j3).eq.1) then
            qp(j3)=qnb(j3)
            else
            endif
            if(qst(j3,qp(j3)).ge.1) then
            if(qs(1,j3,qp(j3)).ge.1) then
            i=1
            goto 102
```

```
        else
        if(qs(2,j3,qp(j3)).ge.1) then
        l=2
        goto 102
        else
        endif
        endif
        else
        qp(j3)=qp(j3)+1
        k12=qp(j3)
        if(k12.eq.tnq) then
        qp(j3)=1
        else
        endif
        D(j3)=D(j3)+wt
        goto 100
        endif

102     kk=1
        zt=irand(iseed(2))
        iseed(2)=zt
        z=float(zt)/2147483647.0
        st=-(1.0/v(2))*alog(1-z)
        if(narv(j3).eq.1.and.kb(j3).eq.1) then
        D(j3)=cat(l,j3,qp(j3),1)+st
        else
        D(j3)=D(j3)+st
        endif
        N(2)=D(j3)
        tnow=N(2)
        nofd(l,j3,qp(j3))=nofd(l,j3,qp(j3))+1
        qs(l,j3,qp(j3))=qs(l,j3,qp(j3))-1
        c8=qs(l,j3,qp(j3))
        qst(j3,qp(j3))=qst(j3,qp(j3))-1
        qse(j3)=qse(j3)-1
        avdel(l,j3,qp(j3))=tnow-cat(l,j3,qp(j3),1)
        dept=tnow
        gavdel(l,j3,qp(j3))=gavdel(l,j3,qp(j3))+avdel(l,j3,qp(j3))
        if(qs(l,j3,qp(j3)).ge.1) then
        do 88 j88=1,qs(l,j3,qp(j3))
        j89=j88+1
        cat(l,j3,qp(j3),j88)=cat(l,j3,qp(j3),j89)
88      continue
        else
        endif
        if(l.eq.2.or.qp(j3).eq.c01) then
        nofdpt=nofdpt+1
        systze=systze-1
        else
        endif
        if(qp(j3).ne.c01.and.l.eq.1) then
c**Nagi(j3) : arrival time to inner-ring gateway queue j3***
```

```
          Nagi(J3)=tnow
          else
          endif
          if(qp(J3).eq.c01) then
          ndpo=ndpo+1
          else
          ndps(i)=ndps(i)+1
          endif

          if(alfa1.eq.alfa.and.narvis.ge.not.and.narvis.eq.nofdpt.and.
     &    narvis.le.notsup.or.narvis.eq.notsup.and.narvis.eq.nofdpt) then
          goto 222
          else
          qp(J3)=qp(J3)+1
          k12=qp(J3)
          if(k12.eq.tnq) then
          qp(J3)=1
          else
          endif
          D(J3)=D(J3)+wt
          if(narvis.eq.1) then
          xy=irand(iseed(1))
          iseed(1)=xy
          x=float(xy)/2147483647.0
          at=-(1.0/v(1))*alog(1-x)
          Na(1)=Na(1)+at
          else
          endif
          goto 100
          endif

 210      c5=narv(J3)
          if(c5.eq.1.and.kb(J3).eq.1) then
          qp(J3)=qnb(J3)
          else
          endif
          if(qs(1,J3,qp(J3)).ge.1) then
          goto 202
          else
          qp(J3)=qp(J3)+1
          k12=qp(J3)
          if(k12.eq.tnq) then
          qp(J3)=1
          else
          endif
 309      D(J3)=D(J3)+wt
          goto 100
          endif

 202      i=1
          zt=irand(iseed(2))
          iseed(2)=zt
          z=float(zt)/2147483647.0
          st=-(1.0/v(2))*alog(1-z)
```

```
      if(narv(j3).eq.1.and.kb(j3).eq.1) then
      D(ir)=cat(i,j3,qp(j3),1)+st
      else
      D(ir)=D(ir)+st
      endif
      N(2)=D(ir)
      tnow=N(2)
      ndpl=ndpl+1
      nofd(i,j3,qp(j3))=nofd(i,j3,qp(j3))+1
      qs(i,j3,qp(j3))=qs(i,j3,qp(j3))-1
      qst(j3,qp(j3))=qst(j3,qp(j3))-1
      qse(j3)=qse(j3)-1
      avdel(i,j3,qp(j3))=tnow-cat(i,j3,qp(j3),1)
      dept=tnow
      c8=qs(i,j3,qp(j3))
      gavdel(i,j3,qp(j3))=gavdel(i,j3,qp(j3))+avdel(i,j3,qp(j3))
      if(qs(i,j3,qp(j3)).ge.1) then
      do 888 j888=1,qs(i,j3,qp(j3))
      j889=j888+1
      cat(i,j3,qp(j3),j888)=cat(i,j3,qp(j3),j889)
888   continue
      else
      endif
c**Determination of the destination outer-ring gateway queue**
      xv=irand(iseed(4))
      iseed(4)=xv
      u=float(xv)/2147483647.0
      do 49 j20=1,nrgs-1
      r1=float(j20)/float(nrgs-1)
      if(u.lt.r1) then
      k0=j20+1
      if(k0.eq.qp(j3)) then
      k0=k0+1
      if(k0.eq.ir) then
      k0=1
      else
      endif
      else
      endif
      goto 79
      else
      endif
49    continue
c*******Nago(k0) : arrival time to outer-ring gateway queue k0*********
79    Nago(k0)=tnow
      qp(j3)=qp(j3)+1
      k12=qp(j3)
      if(k12.eq.tnq) then
      qp(j3)=1
      else
      endif
      D(j3)=D(j3)+wt
```

```
              goto 100
c***************************************************
c***************************************************
222           do 64 i=1,2
              do 65 j=1,ir
              if(j.eq.ir) then
              kj=nrgs
              else
              kj=c01
              endif
              do 68 k=1,kj
              if(j.ne.ir.and.k.ne.c01) then
              delsc(i)=delsc(i)+gavdel(i,j,k)
              else
              if(i.eq.1.and.j.ne.ir.and.k.eq.c01) then
              delgo=delgo+gavdel(1,j,k)
              else
              endif
              endif
              if(j.eq.ir.and.i.eq.1) then
              delgi=delgi+gavdel(1,j,k)
              else
              endif
68            continue
65            continue
64            continue
              ds1=delsc(1)/ndps(1)
              ds2=delsc(2)/ndps(2)
              dgi=delgi/ndpi
              dgo=delgo/ndpo
              dc1=ds1+dgi+dgo
              dc2=ds2
              ds=alfa1*ds1+(1.-alfa1)*ds2
c**********dc1 : class 1 message mean delay*********************
c**********dc2 : class 2 message mean delay*********************
c**********ds  : overall mean delay at a station queue*************
              write(10,6766) dc1,dc2,ds,alfa1
6766          format(1x,'dc1=',f11.5,'dc2=',f11.5,'ds=',f11.5,'alfa1=',f11.5)
              stop
              end
```