M-Interference Queue Analysis by Decomposition Method

by

Xiao-Xiong Yao

A thesis submitted to the
School of Graduate Studies and Research
in partial fulfillment of the requirement for the degree of
Master of Applied Science

Ottawa-Carleton Institute for Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
University of Ottawa

ISBN 0-315-85795-1

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

To my Parents

Contents

List of Figures

Glossary of Notations

$B(t)$: the packet service time distribution function.

$V(t)$: the vacation time distribution function.

$B^*(s)$: the LST of the pdf of the packet service time.

$T^*(s)$: the LST of the pdf of the packet response time.

$W^*(s)$: the LST of the pdf of the packet waiting time.

$V^*(s)$: the LST of the pdf of the vacation time.

$B_g^*(s)$: the LST of the pdf of the message (batch) service time.

$T_g^*(s)$: the LST of the pdf of the message (batch) response time.

$W_g^*(s)$: the LST of the pdf of the message (batch) waiting time.

$G(z)$: the PGF of the message (batch) size distribution.

$Q(z)$: the PGF of the queue length distribution.

$\bar{b}$: the mean packet service time.

$\overline{b^i}$: the ith moment of the packet service time.

$\bar{g}$: the mean of the message (batch) size.

$\overline{g^i}$: the ith moment of the message (batch) size.

λ : the Possoin arrival rate.

$E[\,]$: the mean value of a distribution.

$Var[\,]$: the variance of a distribution.

ACKNOWLEDGEMENTS

This research was supported by Natural Sciences and Engineering Research Council of Canada under Grant No. OGP0042878.

I would like to express my appreciation to my M.A.Sc Committee Consisting of Dr. Nicolas D. Georganas and Dr. S. A. Mahmoud. I am particularly grateful to Dr. Oliver Yang, my advisor, for teaching me queueing theory and for his stimulating and knowledgeable guidance during the course of this work. Thanks must go to Dr. Louis Orozco-Barbosa, who was always eager to help me programming some simulations. I would like to thank all the colleagues for their support.

Finally, this thesis is dedicated to my parents. Their encouragement has motivated me throughout the course of my study.

Chapter 1

Introduction

1.1 Background

Shared-channel communication systems have been developed and extensively studied during the past decade. In such systems a broadcast communication channel is to be shared among a group of geographically distributed stations each demanding access to the channel in a random fashion. The most common shared-channel communication networks are: 1) *Satellite networks,* in which ground stations communicate with each other by a radio channel with the satellite acting as a repeater, 2) *Packet radio networks,* in which many stations communicate with each other by a ground radio channel, 3) *Local area networks,* in which many stations, geographically close to each other, communicate with each other by using a cable medium (i.e., a coaxial cable or a fibre optic cable).

The main issue in designing shared-channel networks is the design of multiple access schemes. A *multiple access scheme* is a set of rules that determines the circumstances under which a station is allowed to transmit. The main goal of a multiple access scheme is to control the stations' transmissions so as to achieve high channel utilizations. Multiple access protocols proposed so far can be classified into the following three categories [48]:

- fixed assignment,
- random access,
- demand assignment.

In a scheme of the first type, a fixed portion of the available channel capacity is dedicated to each station. The most common forms of this technique are FDMA (frequency division multiple access) and TDMA (time division multiple access) [5], [47, sct 2.5] and [48]. Though the allocation can be tailored to the relative need of each user, fixed assignment can be wasteful if the users' demand is highly bursty.

In a scheme of the second type, any station tries to acquire the channel without completely coordinating its action with others. Such schemes include ALOHA [2], Slotted-ALOHA [37] and CSMA [25]. While the performances of these schemes [25] [47, chpt 2] are very good for low traffic demand, they tend to become poor at times of high traffic demand. This is because collisions are more likely to occur when many stations try to access the channel simultaneously (a collision occurs if two or more stations transmit at the same time).

In a scheme of the third type, explicit control information regarding the users' need for the channel capacity is exchanged. The control scheme can be centralized or distributed [40, chpt 8] [47, chpt 2]. In a centralized control scheme, a central controller determines who should have the channel-access right next. In a distributed control scheme, each station monitors the requests of all stations and executes an identical distributed algorithm based on the requests to determine who should have the channel-access right next. In both control schemes, control information must be exchanged through the channel; this means overhead. Therefore, the performances of these schemes are poor for light traffic demand, but are generally good for heavy traffic demand. Examples of these schemes include MSAP (token ring) [27] and others [40, chpt 8] [47, chpt 2].

A common property in the performance analysis of shared-channel networks is the inter-dependence of their queueing behaviors among all the stations in the system that forms a class of interesting problem called the *M-Interference Queueing Problem.* The M-interference queueing problem has been studied in various contexts for a long time and various analytical approaches have been made. For example, a method called topological approximation is presented in [28]; a sub-class of this problem which focuses on several slotted multiple access protocols was studied in [14] and [43].

1.2 Research Motivation and Approaches

In a shared-channel communication network, each station has a buffer where packets wait until their actual transmission. In order to analyze the performance of the network, the construction of an M-dimensional Markov chain with a state vector S = $[S_1, S_2, \cdots, S_M]$ is required, where M is the number of stations and S_i is the number of packets in the ith station's buffer. In general, stations interact with each other through the multiple access schemes and S_i, $i = 1, \cdots, M$ are dependent on each other. Obviously, in the case of having an infinite buffer, the number of states is also infinite. Even in the case of having a buffer with a finite capacity of L units, the number of states ($\sim L^M$ states) may not permit an exact solution of the Markov chain even for small values of L and M. Therefore, for stations with infinite buffer capacities, approximate methods may be required.

In this thesis we study an approach called the *Decomposition Method* that would decompose the interaction problem into a set of simple queueing problems, so that we can concentrate on the queueing of one station only. This method has been used for the analysis of some particular protocols [8] and [43] only. Therefore, we are motivated to provide a more general approach to examine several classes of protocols that are decomposable.

The common model for this research is a system of M symmetric queues sharing a single server through a particular multiple access scheme. For each model, the following steps are taken for the study.

1. Apply the assumption that the queueing behavior of one user in the system can be analyzed by taking into account the influence of the other users through the M/G/1 with vacation model and the M/G/1 queue in which the first customer of each busy period receives an exceptional service. Then the M interfering queues in the system can be decomposed into M individual and independent queues and the performance of the whole system can be evaluated through the analysis of the queueing behavior of one of the M queues only.

2. The M/G/1 with vacation model and the M/G/1 queue in which the first customer of each busy period receives an exceptional service are used to analyze the queueing behavior of a station in the system. The advantage of these models lies as follows. To a station in a shared channel communication network, the access to the communication channel is controlled by the multiple access protocol. In some protocols (i.e.,TDMA and token ring), when the station gets the access right, it can transmit the packets in its buffer, otherwise it has to wait. This can be imagined as the queue being visited by a server when it gets service and the queue has to wait for service when the server is on vacation. So the operation of the M/G/1 queue with vacations exactly reflects the mechanism of these systems (i.e., TDMA and token ring). The dependency among all the stations in the system can be taken into account through the vacation time distribution. In other protocols (i.e., slotted ALOHA), when the station has packets in the buffer it transmits without having to get the access right. However, collisions will happen in this kind of networks. Due to the collision, the service time of each packet will depend on the bahaviors of other stations in the system. In this case, the M/G/1 queue in which the first customer of each busy period receives an exceptional service can be used to model the operation of the multiple access protocols (i.e., slotted ALOHA) and the dependency among all the stations can be taken into account through the service time distribution.

3. Apply more assumptions. For example, the service time of a packet is independent of its waiting time and the vacation time, and the arrival process to each

station is an independent identical Poisson process.

4. Apply simulation techniques to verify our analysis and the assumptions we have made. A set of simulation programs using the QNAP2 [1] software package are developed. For each system in our case studies, simulations are run based on the parameters we use, and simulation results are compared against the analytical results for some typical examples.

1.3 Thesis Organization and Contributions

We will study the shared-channel communication networks by means of case studies that are chosen from three basic categories. They are 1) TDMA which is chosen from the class of fixed assignment multiple access protocols, 2) Token ring which is a typical example of the demand assignment multiple access protocols, 3) Slotted-ALOHA which is a typical example of random multiple access systems. Since each case is an important topic by itself, we shall study each one in full detail and reveal its performances by the decomposition method. Due to the difficulties in the analysis, it is necessary to present some basic analysis in queueing theory. For this reason, this thesis consists of two logical parts. The first part, containing chapters 2 and 3, is a theoretical part that reviews and solves several basic queueing systems. The second part, containing chapters 4 to 6, studies the performances of several shared-channel communication networks by the decomposition approach. The results derived in the first part are used in the second part where we analyze the performances of shared-channel communication networks. Due to the nature of this work, the previous work related to each topic is separately reviewed in each chapter.

Chapter 2, An M/G/1 System with Independent Vacations, introduces the basics of queueing analysis that are modified and extended in later analysis. It summarizes the key results in an M/G/1 system, an M/G/1 system where the first packet of each busy period receives an exceptional service, and finally an M/G/1 system in which the server takes a vacation whenever the queue becomes empty.

Chapter 3, An M/G/1 System with Non-independent Vacations, extends the M/G/1 queue with vacation model by assuming the dependency of vacation periods on the queue states. It derives the expression of the probability generating function (PGF) of the queue length, the expressions of the Laplace-Stieitjes transforms (LST) of the probability density function (pdf) of the response time and the waiting time. Batch Poisson arrival analysis, important for the message delay analysis, is also presented.

Chapter 4, An Exact Analysis of TDMA Systems, studies the queueing behavior of TDMA systems with M stations, each with an infinite buffer. LSTs of the pdf of the response time and the waiting time are obtained, and closed form expressions for the first 2 moments are derived for both the packet response time and the message response time.

Chapter 5, Token Ring Analysis, analyzes the performance of token ring systems with M symmetric stations, each with an infinite buffer. Two service policies are considered in this chapter: 1) exhaustive service policy, 2) single service policy. For both policies, the packet delay as well as the message delay expressions are obtained by using the M/G/1 vacation model. The analytical results are validated by the simulation results. Based on that, more performance studies are carried out.

Chapter 6, An Analysis of Slotted ALOHA Systems, studies a slotted ALOHA system with M symmetric stations, each with an infinite buffer. Through the construction and solution of an embedded Markov chain, the number of busy stations, the service time distributions are derived from which closed form expressions for the LST and the mean of the packet response time are obtained.

The main contributions of this thesis lie in the following areas:

1. To promote the decomposition method as an approach to analyze the general M-interference queueing problem.
2. To use the M/G/1 queue with vacation model or the M/G/1 queue in which the first packet receives an exceptional service as a tool to analyze several classes of M-interference queue problems, namely the TDMA systems, the token ring

systems and the slotted ALOHA systems.

3. To derive the expressions of the PGF of the queue length distribution, the LSTs of the pdf of the packet waiting time and the response time, and the message waiting time and the response time for the M/G/1 queue with non-independent vacations. From these, closed form expressions for the first two moments of various distributions are obtained.

4. To provide a new approach to the analysis of TDMA systems. Variances are computed and verified by simulation results.

5. To analyze the message delay performances for both the exhaustive and the single service policies in the token ring systems. The performances of the system under various parameters are discussed and compared.

6. To obtain by a Markov chain approach the state transition probabilities for the number of busy stations in the slotted ALOHA system with a finite user population. Based on these, delay performances of the slotted ALOHA systems are evaluated.

7. Lastly, various QNAP2 simulation programs have been written to verify analytical results, especially the results obtained by approximations and assumptions. The coding of some of these systems, i.e., the slotted ALOHA system, is quite complicated and interesting, and represents some major undertaking.

Chapter 2

An M/G/1 System with Independent Vacations

As a review, we shall summarize in this chapter the key results of the M/G/1 queue. Then we shall examine variations of the M/G/1 system in which the first customer of each busy period receives an exceptional service. Finally we study the M/G/1 system with independent vacations and reveal its property. These reviews provide equations and ideas that will be used in the later study.

2.1 An M/G/1 System

The M/G/1 system [6] [23] is the very basis for the M/G/1 vacation model. It is a single-server, single-queue system with a Poisson packet arrival process at a rate λ, and a random service time with a distribution function $B(t)$. The mean service time $\overline{b}$, the ith moment of the service time distribution $\overline{b^i}$ (i=2, 3 $\cdots$), and the Laplace-Stieltjes transform (LST) of the probability density function (pdf) denoted by $B^*(s)$ are given by

$$B^*(s) = \int_0^\infty e^{-st} dB(t), \tag{2.1}$$

$$\overline{b} = \int_0^\infty t dB(t) = -B^{*(1)}(0), \tag{2.2}$$

$$\overline{b^i} = \int_0^\infty t^i dB(t) = (-1)^i B^{*(i)}(0) \qquad i = 2, 3, \cdots, \tag{2.3}$$

where

$$B^{*(i)}(0) = (\frac{d^i B^*(s)}{ds^i})_{s=0} \qquad i = 1, 2, \cdots. \tag{2.4}$$

We assume that a queue has an infinite buffer to store the waiting packets, and the server serves the queue continuously according to first-come first-serve (FCFS) policy until the queue becomes empty.

2.1.1 Number of Packets in the Queue

By selecting Markov points at the service completion points (departure instants) of each packet, the steady state probability generating function (PGF) [6] [23] for the number of packets in the queue at these points can be analyzed and obtained as,

$$Q_{M/G/1}(z) = \sum_{k=0}^{\infty} q_k z^k = \frac{q_0(1-z)B^*(\lambda - \lambda z)}{B^*(\lambda - \lambda z) - z}, \tag{2.5}$$

where q_0 can be determined by the condition $Q(1) = 1$, and is given by

$$q_0 = 1 - \rho = 1 - \lambda \bar{b}. \tag{2.6}$$

Equation (2.5) is one form of the *Pollaczek-Khinchin transform equation.*

Let $\tilde{L}$ be the queue length random variable and $\tilde{T}$ be the response time random variable. By taking derivative with respective to z and setting z equal to one, the mean queue length can be obtained as

$$E[L] = \rho + \frac{\lambda^2 \overline{b^2}}{2(1-\rho)}. \tag{2.7}$$

From Little's formula [35], we can get the mean response time $E[T]$ as

$$E[T] = \frac{E[L]}{\lambda} = \bar{b} + \frac{\lambda \overline{b^2}}{2(1-\rho)}. \tag{2.8}$$

It is clear from an inspection of equation (2.7) that $\rho < 1$ is the necessary and sufficient condition for the system to achieve steady state.

2.1.2 Packet Waiting Time

Since the waiting time of a packet in the queue is independent of the arrival process and of the packet service time, the relationship between the PGF of the queue length distribution at the departure instants and the LST of the pdf of the waiting time $W^*_{M/G/1}(s)$ is readily available via [6] and [23],

$$Q_{M/G/1}(z) = W^*_{M/G/1}(\lambda - \lambda z)B^*(\lambda - \lambda z). \tag{2.9}$$

Changing the variable to $s = \lambda - \lambda z$, we obtain the LST of the pdf of the waiting time

$$W^*_{M/G/1}(s) = \frac{s(1-\rho)}{s - \lambda + \lambda B^*(s)}. \tag{2.10}$$

Take derivative with respective to s and set s equal to zero, the mean waiting time is given by

$$E[W_{M/G/1}] = \frac{\lambda \overline{b^2}}{2(1-\rho)}, \tag{2.11}$$

and the second moment is

$$E[W^2_{M/G/1}] = \frac{\lambda^2 (\overline{b^2})^2}{2(1-\rho)^2} + \frac{\lambda \overline{b^3}}{3(1-\rho)}. \tag{2.12}$$

Define the response time of a packet $\tilde{T}_{M/G/1}$ to be the sum of its waiting time and its service time. Due to the fact that the service time is independent of the waiting time, thus the LST of the pdf of the response time is

$$T^*_{M/G/1}(s) = W^*_{M/G/1}(s)B^*(s). \tag{2.13}$$

The first and second moments are given by

$$E[T_{M/G/1}] = E[W_{M/G/1}] + E[B] = \bar{b} + \frac{\lambda \overline{b^2}}{2(1-\rho)}, \tag{2.14}$$

$$E[T^2_{M/G/1}] = E[W^2_{M/G/1}] + 2E[B]E[W_{M/G/1}] + E[B^2]. \tag{2.15}$$

From (2.7), (2.11) and (2.14), we have the relation

$$E[L] = \lambda E[T_{M/G/1}] = \lambda(E[W_{M/G/1}] + \bar{b}), \tag{2.16}$$

which again confirms Little's Rule.

2.2 Exceptional Service for the First Packet in a Busy Period

Consider a variation of the M/G/1 system where the first packet of each busy period receives an exceptional service [49] (i.e., this service time has a different distribution function from others that follow). Although it is not a system with vacations, the analytical results are summarized here for their similarity and the needs of later analysis.

Let $B_1^*(s)$ be the LST of the service time pdf of the packet that initializes a busy period, and $\overline{b}_1$ be its mean value. The steady state PGF $Q(z)$ [49] for the number of packets in the queue at departure points is given by

$$Q(z) = q_0 \frac{zB_1^*(\lambda - \lambda z) - B^*(\lambda - \lambda z)}{z - B^*(\lambda - \lambda z)}, \tag{2.17}$$

where

$$q_0 = \frac{1 - \lambda\overline{b}}{1 + \lambda(\overline{b_1} - \overline{b})}. \tag{2.18}$$

The LST of the pdf of the packet response time is then given by

$$T^*(s) = Q(1 - s/\lambda) = \frac{1 - \lambda\overline{b}}{1 + \lambda(\overline{b_1} - \overline{b})} \frac{\lambda B^*(s) - (\lambda - s)B_1^*(s)}{s - \lambda + \lambda B^*(s)}. \tag{2.19}$$

From that we have

$$E[T] = \frac{\overline{b_1}}{1 + \lambda(\overline{b_1} - \overline{b})} + \frac{\lambda\overline{b^2}}{2(1 - \lambda\overline{b})} + \frac{\lambda(\overline{b_1^2} - \overline{b^2})}{2(1 + \lambda\overline{b_1} - \lambda\overline{b})}. \tag{2.20}$$

The LST of the pdf of the waiting time also can be obtained as

$$W^*(s) = \frac{1 - \lambda\overline{b}}{1 + \lambda(\overline{b_1} - \overline{b})} \frac{s - \lambda B_1^*(s) + \lambda B^*(s)}{s - \lambda + \lambda B^*(s)}, \tag{2.21}$$

and its mean

$$\begin{aligned} E[W] &= \frac{\lambda\overline{b^2}}{2(1 - \lambda\overline{b})} + \frac{\lambda(\overline{b_1^2} - \overline{b^2})}{2(1 + \lambda\overline{b_1} - \lambda\overline{b})} \\ &= E[T] - \frac{\overline{b_1}}{1 + \lambda(\overline{b_1} - \overline{b})}. \end{aligned} \tag{2.22}$$

2.3 An M/G/1 System with Independent Vacations

In the following we consider an M/G/1 queueing system with independent vacations. In such a system, the server begins a vacation of a random length each time when the queue becomes empty. If the server returns from a vacation to find one or more packets waiting, it works until the queue becomes empty again (exhaustive service), and then it begins another vacation. If the server returns from a vacation to find no packets waiting, it begins another vacation immediately, and continues in this manner until it finds at least one packet waiting upon returning from a vacation. Thus unlike the M/G/1 system that the server alternates between busy and idle states, the server here is either busy or on vacation at all times. The M/G/1 system with independent vacations is similar to the M/G/1 system in which the first packet in a busy period receives exceptional service in such a way that the packet arrives at an empty queue will be treated differently from other packets. In the system studied in the previous section, the first packet has an exceptional service; while in the M/G/1 system with independent vacations, the first packet has to suffer an additional delay due to the presence of vacations.

The model for a queueing system where the server goes on vacation of a random length whenever it becomes idle was first analyzed in [36]. The server with rest periods was analyzed by using another approach in [39]. As a special case for the server with a starter, the M/G/1 queue with independent vacations was analyzed in [34]. The emphasis in [34] is on the study of queues with a starter, while a queue with independent vacations was considered as a special case. The M/G/1 vacation system with finite buffer capacities was also reported in [32]. A more comprehensive survey of the M/G/1 queue with vacations can be found in [9] and [13]

2.3.1 Number of Packets in the Queue

Let $V^*(s)$, $\overline{v}$, and $\overline{v^2}$ be the LST of the pdf, the mean, and the second moment of the vacation time respectively. As in the M/G/1 queueing analysis, by selecting Markov points at departure points, we have the steady state PGF for the number of packets in the queue at departure points [23],

$$Q(z) = \frac{q_0(1 - V^*(\lambda - \lambda z))B^*(\lambda - \lambda z)}{B^*(\lambda - \lambda z) - z}, \tag{2.23}$$

where q_0 is now given by

$$q_0 = \frac{1 - \lambda \overline{b}}{\overline{v}}. \tag{2.24}$$

Rewriting equation (2.23), we have

$$Q(z) = \frac{1 - V^*(\lambda - \lambda z)}{\overline{v}(1 - z)} Q_{M/G/1}(z), \tag{2.25}$$

where $Q_{M/G/1}(z)$ is given in equation (2.5) for the M/G/1 queueing system.

2.3.2 Packet Waiting Time

Since the waiting time of a packet is independent of the subsequent arrival process, then similar to the M/G/1 queue analysis, the queue length distribution and the waiting time distribution can also be related as

$$Q(z) = W^*(\lambda - \lambda z)B^*(\lambda - \lambda z), \tag{2.26}$$

and the LST of the pdf of the packet waiting time and the packet response time are now given by

$$W^*(s) = \frac{1 - V^*(s)}{s\overline{v}} W^*_{M/G/1}(s), \tag{2.27}$$

$$T^*(s) = \frac{1 - V^*(s)}{s\overline{v}} T^*_{M/G/1}(s). \tag{2.28}$$

The first two moments of the waiting time are

$$E[W] = E[W_{M/G/1}] + \frac{E[V^2]}{2E[V]}, \tag{2.29}$$

$$Var[W] = E[W^2] - (E[W])^2, \tag{2.30}$$

where

$$E[W^2] = E[W^2_{M/G/1}] + \frac{E[V^2]E[W_{M/G/1}]}{E[V]} + \frac{E[V^3]}{3E[V]}, \tag{2.31}$$

in which $E[W_{M/G/1}]$ and $E[W^2_{M/G/1}]$ are the first and the second moments of the waiting time for the M/G/1 system and are given in (2.11) and (2.12), respectively. Accordingly the moments of the response time can be obtained by

$$E[T] = E[W] + E[B], \tag{2.32}$$

$$Var[T] = E[T^2] - (E[T])^2, \tag{2.33}$$

and

$$E[T^2] = E[W^2] + 2E[B]E[W] + E[B^2], \tag{2.34}$$

where $E[B] = \bar{b}$ and $E[B^2] = \overline{b^2}$. Equations (2.25) and (2.27) demonstrate the stochastic decomposition property of the M/G/1 queue with independent vacations.

2.3.3 Batch Arrival

An extension of the M/G/1 vacation model is the batch arrival system with vacations. Each arrival now corresponds to the arrival of a batch of packets, where the batch sizes are independent and identically distributed random variables. The study of this model here can be used for later analysis of message delay performance.

Let g_k be the probability that the batch size (the number of packets in a batch) is k ($k = 1, 2, \cdots$), and let $G(z)$, $\bar{g}$, and $\overline{g^i}$ be its PGF, mean, and ith moment,

$$G(z) = \sum_{k=1}^{\infty} g_k z^k. \tag{2.35}$$

Since the server goes on vacation only when the queue is empty, and begins serving when it returns to find at least one packet in the queue, the service time of a batch is independent of the vacation time and furthermore the service of a batch can be viewed as a superpacket, we have the service time random variable of a batch to be

$$\tilde{b}_g = g_1\tilde{y}_1 + g_2\tilde{y}_2 + \cdots + g_k\tilde{y}_k + \cdots. \tag{2.36}$$

where $\tilde{y}_k$ stands for the service time for k packets. By assuming independent packet service time, the LST of the pdf of the batch service time can be derived as

$$B_g^*(s) = g_1 B^*(s) + g_2 (B^*(s))^2 + \cdots . \tag{2.37}$$

and from equation (2.35), we can conclude that

$$B_g^*(s) = \sum_{k=1}^{\infty} (B^*(s))^k g_k = G(B^*(s)). \tag{2.38}$$

By combining the M/G/1 decomposition property and the batch arrival property [6], we can have the PGF of the queue length distribution in terms of the number of batches in the queue

$$Q_g(z) = \frac{q_{g0}(1 - V^*(\lambda - \lambda z))G(B^*(\lambda - \lambda z))}{G(B^*(\lambda - \lambda z)) - z}, \tag{2.39}$$

where q_{g0} is given by

$$q_{g0} = \frac{1 - \lambda \overline{g} \overline{b}}{\overline{v}}. \tag{2.40}$$

Define the waiting time of a batch to be the time lapse measured from the arrival of the batch till the start of the service of the first packet in the batch. Then the LST of the pdf for the waiting time of a batch can be obtained as

$$W_g^*(s) = \frac{1 - V^*(s)}{sE[V]} \frac{s(1 - \lambda \overline{g} \overline{b})}{s - \lambda + \lambda G(B^*(s))} \tag{2.41}$$

For ease of notations, we shall denote $W^*_{M/G/1}(s) \mid_{B^*(s)=f(s)}$ to mean the same $W^*_{M/G/1}(s)$ in equation (2.10) except $B^*(s)$ is now replaced by $f(s)$. This shall be used for the remainder of the thesis. For example, in equation (2.41), $f(s) = G(B^*(s))$ and equation (2.41) can be written as

$$W_g^*(s) = \frac{1 - V^*(s)}{sE[V]} W^*_{M/G/1}(s) \mid_{B^*(s)=G(B^*(s))} . \tag{2.42}$$

The moments of the waiting time of a batch can be computed as

$$E[W_g] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1} \mid_{B^*(s)=G(B^*(s))}], \tag{2.43}$$

$$Var[W_g] = E[W_g^2] - (E[W_g])^2, \tag{2.44}$$

where $E[W_g^2]$ is given by

$$\begin{aligned} E[W_g^2] &= \frac{E[V^3]}{3E[V]} + \frac{E[V^2]E[W_{M/G/1} \mid_{B^*(s)=G(B^*(s))}]}{E[V]} \\ &+ E[W^2_{M/G/1} \mid_{B^*(s)=G(B^*(s))}]. \end{aligned} \tag{2.45}$$

The response time of a batch is defined to be the time interval starting from the arrival of a batch until the service completion of the last packet of the batch. It is the sum of the waiting time of a batch and the service time of the batch. Due to the independence assumption of the waiting time and the service time of the batch, the response time of the batch can be obtained by

$$T_g^*(s) = W_g^*(s)B_g^*(s). \tag{2.46}$$

Its mean and variance are

$$E[T_g] = E[W_g] + E[B_g], \tag{2.47}$$

$$Var[T_g] = E[T_g^2] - (E[T_g])^2, \tag{2.48}$$

where $E[T_g^2]$ is given by

$$E[T_g^2] = E[W_g^2] + 2E[B_g]E[W_g] + E[B_g^2], \tag{2.49}$$

$E[B_g]$ and $E[B_g^2]$ can be obtained from equation (2.38)

$$E[B_g] = \overline{g}\overline{b}, \tag{2.50}$$

$$E[B_g^2] = (\overline{g^2} - \overline{g})\overline{b}^2 + \overline{g}\overline{b^2}. \tag{2.51}$$

From Little's rule, we can easily have the mean queue length in terms of batch,

$$E[Q_g] = \lambda\left(\frac{E[V^2]}{2E[V]} + \frac{\lambda E[B_g^2]}{2(1-\lambda E[B_g])} + E[B_g]\right). \tag{2.52}$$

Finally, the variance of the queue length can be obtained from

$$Var[Q_g] = \lambda^2 Var[T_g]. \tag{2.53}$$

2.4 Explanation for the Delay of the M/G/1 Queue with Vacations

In the previous sections we showed that the delay in a queue with vacations is the sum of two independent random variables:

- the delay in a queue without vacations;
- additional delay distributed as the residual life of the vacation period.

Yet we did not give a direct queueing explanation for the fact that the additional delay is distributed as the residual life of the vacation period. We shall do that in this section.

Consider the busy and the idle periods in a regular M/G/1 system (denoted as system A) described in figure 2.1a [23]. We denote busy periods by Y_1, Y_2, $\cdots$ and idle periods by X_1, X_2, $\cdots$. Now, let us impose vacations on this system (the new system is denoted as system B). Assume that the vacation is just another job the server has to do. Thus, if we look from the server's point of view we notice three properties:

1. The server always consumes work at a rate of "one unit of work per unit of time".
2. At time points where a vacation V_i starts, additional work, equaling (in amount) the vacation length, arrives at the system (from the server's point of view).
3. A new vacation starts if and only if the amount of work in the system is exactly zero. This means that the server takes a new vacation either when it finishes working in the M/G/1 system or when it returns from a vacation and finds the M/G/1 system still empty.

This situation is illustrated in figure 2.1b. The solid line represents the total amount of work as seen by the server (denoted by $U_{server}(t)$), while the shadowed area represents the unfinished work (M/G/1 packets) in the M/G/1 system with no vacations (denoted $U_{M/G/1}(t)$). We notice that the server is continuously busy at a rate of "one work unit per time unit" and that the vacation work (vacation packets) always arrives to the server system when $U_{server}(t)$ drops to zero. Next, we notice that the server system serves in first-come first-served (FCFS) order and has the following properties:

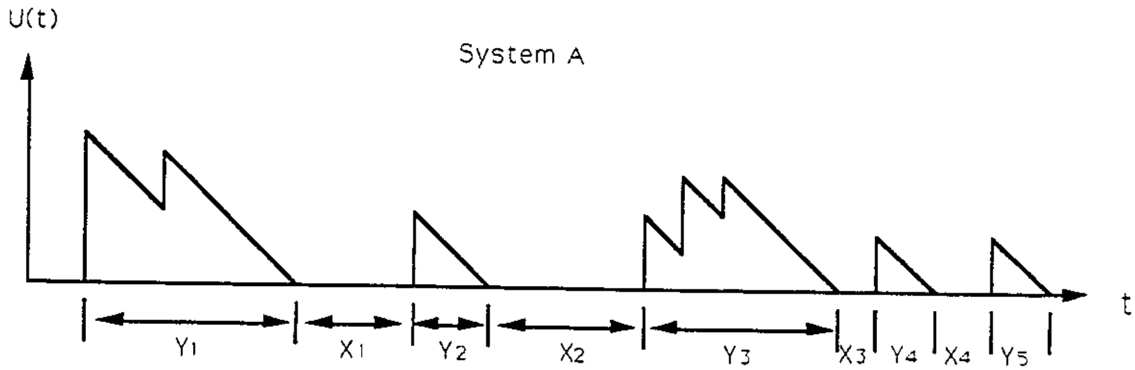
- M/G/1 packets are served according to FCFS policy;
- vacation packets arrive only when there are no M/G/1 packets;
- the server completes service of any packet of any type before serving the next packet (non-preemptive system).

With these properties, it is clear that the total time in the system for an M/G/1 packet arriving at time t to the system with vacations is exactly $U_{server}(t)$. Since the time in system for the same packet in a system without vacations is $U_{M/G/1}(t)$; thus, the additional delay suffered by this packet is given by $U_{server}(t) - U_{M/G/1}(t)$.

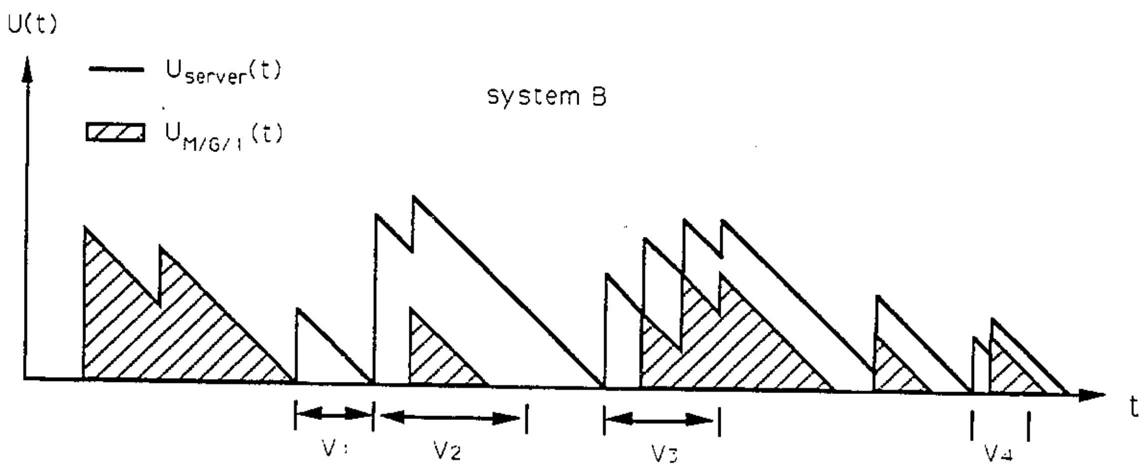
In figure 2.2a. we plot the function difference $U_{server}(t) - U_{M/G/1}(t)$ (denoted by $D(t)$) versus t. In this figure the following properties can be noticed:

1. In time segments corresponding to idle periods in system A (figure 2.1a), $D(t)$ is consumed at the rate of "one work unit per time unit",
2. In time segments corresponding to busy periods in system A, $D(t)$ remains constant.
3. The time epoches where $D(t)$ increases are those corresponding to the beginning of vacations. At such a moment, $D(t) = 0$ and discontinuously increase to the height of the vacation starting at that time.

In this figure we note that $D(t)$ is independent of any property of a system A busy period (excluding its timing) since it stays constant during the duration of such periods. $D(t)$ is determined only by the length of the vacations and the length of the



(a) The unfinished work in a regular M/G/1



(b) Vacation periods "added" to a regular M/G/1

Figure 2.1: The unfinished work in a system with vacation periods

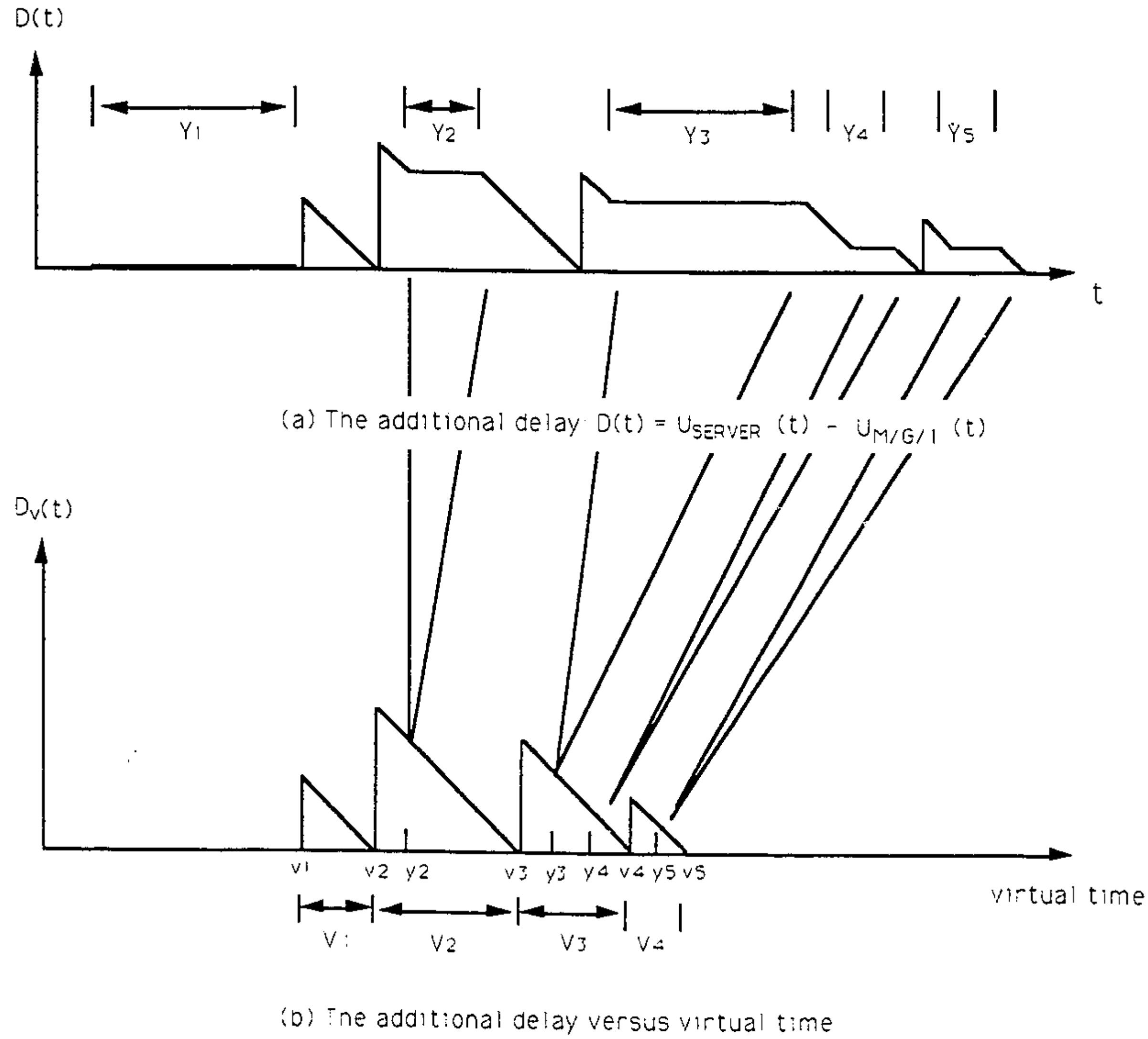


Figure 2.2: The additional delay in a system with vacation periods

system A idle periods. Consequently, the additional delay in the queue with vacations is independent of the delay in the regular M/G/1 system.

Since the additional delay is independent of any property of system A busy periods, we can represent any system A busy period by its starting point only. We do so by contracting the flat segments of $D(t)$ to a point. This is done in figure 2.2b, where the time axis becomes a virtual time axis and a segment Y_i from figure 2.2a is contracted to a point y_i. The point corresponding to the beginning of a vacation, V_i, is denoted by v_i. For this figure we define $D_v(t)$ as the (virtual) additional delay of virtual time t as seen in figure 2.2b. In the transformation from figure 2.2a to 2.2b, we notice the following properties:

1. $D_v(y_i)$ equals the additional delay suffered by all packets of busy period Y_i in figure 2.1a.

2. D_v continuously decreases at the rate of "one work unit per time unit". Whenever D_v becomes zero, it increases by a discontinuous increment.

3. The increment of D_v occur in epoches corresponding to vacation starts. The increment size is the vacation length.

4. Let t be an arbitrary time epoch on the virtual time axis and v_1 be the epoch corresponding to the first vacation starts after t. From properties 2 and 3 and from the structure observed in figure 2.2b, it is clear that $D_v(t) = v_1 - t$.

From these arguments it becomes clear that, in order to find the additional delay suffered in system A, one may compute D_v for the points $\{y_i\}$ in figure 2.2b. We note that the length of a segment $(v_i,\ v_{i+1})$ is distributed according to the distribution of vacation length. Also, the intervals between the adjacent y points represent lengths of idle periods; so, they are exponentially distributed. From the property of the Poisson arrival processes, we have

$$D^*(s) = \frac{1 - V^*(s)}{E[V]s}. \tag{2.54}$$

Yes! this is the residual life of the vacation period! Thus, we arrive at the expected conclusion: D_v, the additional delay of the packets in a system with vacation periods, is distributed as the residual life of a vacation period.

2.5 Summary

The $M/G/1$ queueing system and the M/G/1 queueing system with independent vacations were briefly reviewed in this chapter. As a variation of the M/G/1 queue, the M/G/1 queue in which the first packet of each busy period receives an exceptional service was also studied. It was shown that the delay in a queue with independent vacations consists of the direct sum of two independent variables: 1) the delay in the equivalent queue without vacations. 2) the additional delay suffered due to the presence of vacations. Finally, an explanation was provided for the decomposition property of the delay through a queue with independent vacations. The results presented in this chapter will be the basis for the analysis in following chapters.

Chapter 3

An M/G/1 System with Non-independent Vacations

In this chapter, we introduce and study the M/G/1 System with non-independent vacations. We consider a queueing system in which a vacation is incurred on each service completion. As opposed to the analysis in chapter 2 for an M/G/1 system with independent vacations, we allow the vacation period in this chapter to depend on the queue state, namely the number of packets in the queue. The LST of the pdf of the response time and the waiting time are obtained for systems with Poisson arrivals as well as batch Poisson arrivals.

3.1 Introduction

In some single server multi-queue systems, the server will leave whenever he finishes serving a packet (i.e., TDMA systems and token ring systems with single service policy). The length of a vacation cycle therefore would depend on the queue lengths of other queues in the system and in turn on the length of the previous cycle.

In contrast to the study in chapter 2, we allow the vacation to depend on the queue state. To understand the importance of these variations for the modelling of queueing systems, let us further describe what systems can be modelled by this model, compared to the systems that can be modelled by the M/G/1 queue with independent vacations.

1. The queue with independent vacations can be described as a system with a dedicated server. The server can take vacations only when the queue gets idle. A queue with non-independent vacations can model a system with a shared server. In this system the server is shared by several queues. So the server is sensitive to the number of packets in each queue. If the queue is empty, the server will be away i.e., for a longer time. Clearly, this is not the only application that can be modelled by a queue with non-independent vacations. As a matter of fact, as already stated, the analysis of this system was motivated by the need to analyze TDMA systems as well as token ring systems with a single service policy.

2. A queue with independent vacations is a special case of a queue with non-independent vacations.

A special case of this analysis has been previously reported in [16] for Poisson arrivals and in [29] for batch arrivals. In their analysis, the server will take a vacation after each service and the vacation period is assumed to be independent of the queue state. A Bernoulli schedule vacation model in which the server takes a vacation with probability $1-p$, or continues to serve the queue with probability p after each service is studied in [41]. Note that the case $p=0$ corresponds to the model studied in [16] and the case $p=1$ corresponds to the M/G/1 queue with independent vacations.

The rest of this chapter is organized as follows. In section 3.2, we will give a description of the model and the assumptions used in later analysis. In section 3.3, we derive the queue length PGF, then the LST of the pdf of the response time, the waiting time and their moments in section 3.4. The results are extended to systems with batch arrivals in section 3.5.

3.2 Model and Assumptions

The queueing system considered in this chapter consists of a single server and a single queue. The arrival process to the queue is a Poisson process (later extended to a batch Poisson process) with an arrival rate λ. Packets which arrive at the queue will be served according to a FCFS order with a $B^*(s)$, the LST of the pdf of the service time, a mean $\overline{b}$ and a second moment $\overline{b^2}$. After each service, the server takes a vacation with a random length $\tilde{v}$. If there are no packets waiting in the system when the server returns from a vacation, it takes a vacation with a random length $\tilde{v}_0$ ($\tilde{v}_0$ can be different from $\tilde{v}$). Vacations are repeated until there is at least one packet found at the end of the vacation. The LSTs of the pdf of the vacation periods are denoted as $V_0^*(s)$ and $V^*(s)$. Furthermore we assume that the service time is independent of the waiting time and the vacation time. Also the waiting time of a packet is assumed to be independent of the subsequent arrival process.

3.3 Number of Packets in the Queue

Due to the state-dependency of the vacation periods, we define p_n to be the steady state probability that the server finds n packets waiting in the queue when it arrives. The state transition probability p_{mn} is defined to be the probability of a queue making transition from one state to the other between two successive instants when the server attends the queue.

Denoting the convolution operator by $*$, we have

$$p_{mn} = \int_0^\infty e^{-\lambda t}\frac{(\lambda t)^{n-m+1}}{(n-m+1)!}d(V(t) * B(t)) \qquad m > 0, \tag{3.1}$$

$$p_{0n} = \int_0^\infty e^{-\lambda t}\frac{(\lambda t)^n}{n!}dV_0(t). \tag{3.2}$$

Then $\{p_n\}$ satisfies the following equation of states:

$$p_n = p_0\int_0^\infty e^{-\lambda t}\frac{(\lambda t)^n}{n!}dV_0(t) + \sum_{m=1}^{n+1} p_m\int_0^\infty e^{-\lambda t}\frac{(\lambda t)^{n-m+1}}{(n-m+1)!}d(V(t) * B(t)). \tag{3.3}$$

Introducing the PGF $P(z)$ of p_n, we have

$$P(z) = \sum_{n=0}^{\infty} p_n z^n. \tag{3.4}$$

Substitute the expression of p_n into the equation and simplify, we get

$$P(z) = p_0 \frac{zV_0^*(\lambda - \lambda z) - V^*(\lambda - \lambda z)B^*(\lambda - \lambda z)}{z - V^*(\lambda - \lambda z)B^*(\lambda - \lambda z)}. \tag{3.5}$$

Let q_n be the probability that, at a service completion point, a departing packet leaves behind n packets in the queue. Then q_n is given by

$$q_n = \sum_{m=1}^{n+1} \frac{p_m}{1 - p_0} \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^{n-m+1}}{(n - m + 1)!} dB(t). \tag{3.6}$$

Introducing the PGF $Q(z)$ of q_n, we have the PGF of the queue length distribution at departure points

$$\begin{aligned} Q(z) &= \sum_{n=0}^{\infty} q_n z^n \\ &= \sum_{n=0}^{\infty} \sum_{m=1}^{n+1} \frac{p_m}{1 - p_0} \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^{n-m+1}}{(n - m + 1)!} dB(t) z^n \\ &= \frac{1 - \lambda(\bar{b} + E[V])}{E[V_0]} \frac{B^*(\lambda - \lambda z)(V_0^*(\lambda - \lambda z) - 1)}{z - V^*(\lambda - \lambda z)B^*(\lambda - \lambda z)}, \end{aligned} \tag{3.7}$$

and its mean

$$E[Q] = \lambda \left(\frac{E[V_0^2]}{2E[V_0]} + \frac{\lambda(\overline{b^2} + 2\bar{b}E[V] + E[V^2])}{2(1 - \lambda(\bar{b} + E[V]))} + \bar{b} \right). \tag{3.8}$$

3.4 Packet Waiting Time

Again assume that the waiting time of a packet is independent of the subsequent arrival process. From the equation

$$Q(z) = W^*(\lambda - \lambda z)B^*(\lambda - \lambda z), \tag{3.9}$$

the LST of the PDF of the waiting time is obtained to be

$$W^*(s) = \frac{1 - V_0^*(s)}{E[V_0]} \frac{1 - \lambda(\bar{b} + E[V])}{s - \lambda + \lambda V^*(s)B^*(s)}. \tag{3.10}$$

Note that the waiting time consists of two factors. One of them is the residual life of the vacation period of $\tilde{v}_0$, the other, denoted as $W_{M/G/1}(s) \mid_{B^*(s)=B^*(s)V^*(s)}$, is the LST of the pdf of the packet waiting time for the M/G/1 queue in which $B^*(s)$ is replaced by $B^*(s)V^*(s)$. This was referred to as the general decomposition property of the delay in the M/G/1 queue with vacations [13]. As a result, the mean and the variance of the packet waiting time are given by

$$E[W] = \frac{E[V_0^2]}{2E[V_0]} + E[W_{M/G/1} \mid_{B^*(s)=B^*(s)V^*(s)}], \tag{3.11}$$

$$Var[W] = E[W^2] - (E[W])^2, \tag{3.12}$$

where $E[W^2]$ is given by

$$\begin{aligned} E[W^2] &= \frac{E[V_0^3]}{3E[V_0]} + \frac{E[V_0^2]}{E[V_0]} E[W_{M/G/1} \mid_{B^*(s)=B^*(s)V^*(s)}] \\ &\quad + E[W^2_{M/G/1} \mid_{B^*(s)=B^*(s)V^*(s)}]. \end{aligned} \tag{3.13}$$

The LST of the pdf of the response time and its moments can be obtained by

$$T^*(s) = W^*(s)B^*(s), \tag{3.14}$$

$$E[T] = E[W] + \bar{b}, \tag{3.15}$$

$$Var[T] = E[T^2] - (E[T])^2, \tag{3.16}$$

$$E[T^2] = E[W^2] + 2\bar{b}E[W] + \overline{b^2}. \tag{3.17}$$

By Little's rule, the mean queue length can be obtained as

$$E[Q] = \lambda E[T]. \tag{3.18}$$

The variance can be obtained as before

$$Var[Q] = \lambda^2 Var[T]. \tag{3.19}$$

If we let $V_0^*(s) = V^*(s)$, i.e., equations (3.10) and (3.11) become respectively

$$W^*(s) = \frac{1 - \lambda(\bar{b} + E[V])}{E[V]} \frac{1 - V^*(s)}{s - \lambda + \lambda V^*(s)B^*(s)}, \tag{3.20}$$

and

$$E[W] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1} \mid_{B^*(s)=B^*(s)V^*(s)}]. \tag{3.21}$$

They are exactly the same as the results in [16]. On the other hand, if we let $V^*(s) = 1$ ($\tilde{v} = 0$), equations (3.7) and (3.10) reduce to

$$Q(z) = \frac{(V_0^*(\lambda - \lambda z) - 1)}{E[V_0]} \frac{(1 - \lambda\bar{b})B^*(\lambda - \lambda z)}{z - B^*(\lambda - \lambda z)}, \tag{3.22}$$

$$W^*(s) = \frac{1 - \lambda\bar{b}}{E[V_0]} \frac{1 - V_0^*(s)}{s - \lambda + \lambda B^*(s)}. \tag{3.23}$$

They are the same as the results in chapter 2 for the M/G/1 system with independent vacations.

3.5 Batch Arrival

The system can be extended to the queue with batch arrivals. Recall that the Poisson batch arrival has a generating function $G(z)$, the first and the second moments of $\overline{g}$ and $\overline{g^2}$ respectively. Since the server takes a vacation after each service, the service time of a batch cannot be simply treated as the compound service time of the packets in the batch. However, if we assume that the service time of a packet is independent of the vacation time, we have the service time of a batch with a size of k packets to be

$$\tilde{b}_k = \tilde{b}_{(1)} + \tilde{v} + \tilde{b}_{(2)} + \cdots + \tilde{v} + \tilde{b}_{(k)}, \tag{3.24}$$

where $\tilde{b}_{(k)}$ is the service time of the kth packet in the batch. Apply the LST to the equation, we have the LST of the pdf of the service time of a batch with k packets $B_k^*(s)$ to be

$$B_k^*(s) = (B^*(s))^k (V^*(s))^{k-1}. \tag{3.25}$$

For a batch with its PGF

$$G(z) = \sum_{k=1}^{\infty} g_k z^k, \tag{3.26}$$

the service time of the batch can be obtained by

$$\tilde{b}_g = g_1\tilde{b}_1 + g_2\tilde{b}_2 + \cdots + g_k\tilde{b}_k + \cdots, \tag{3.27}$$

where $\tilde{b}_k$ is given in equation (3.24). The LST of the pdf of the service time of a batch can be derived as

$$B_g^*(s) = g_1B_1^*(s) + g_2B_2^*(s) + \cdots + g_kB_k^*(s) + \cdots. \tag{3.28}$$

From equations (3.25) and (3.26), we have the LST of the pdf of the service time distribution of a batch,

$$B_g^*(s) = \sum_{k=1}^{\infty} g_k(B^*(s))^k(V^*(s))^{k-1}. \tag{3.29}$$

After simplifying, we have

$$B_g^*(s) = \frac{G(B^*(s)V^*(s))}{V^*(s)}. \tag{3.30}$$

Define the waiting time of a batch to be the time lapse measured from the arrival of the batch till the start of service of the first packet in the batch. Considering the independence among the packet service time, the vacation duration $\tilde{v}_0$ and the vacation duration $\tilde{v}$, we can view the sum of the packet service time and the vacation time $\tilde{v}$ as the "virtual service time". As a result, we can apply the decomposition property and the batch arrival property [6] to obtain the LST of the pdf for the waiting time of a batch

$$\begin{aligned} W_g^*(s) &= \frac{1-V_0^*(s)}{sE[V_0]}\frac{s(1-\rho')}{s-\lambda+\lambda G(V^*(s)B^*(s))}, \\ &= \frac{1-V_0^*(s)}{sE[V_0]}W_{M/G/1}^*(s) \mid_{B^*(s)=G(B^*(s)V^*(s))} \end{aligned} \tag{3.31}$$

where $\rho' = \lambda\ \overline{g}(\overline{v} + \overline{b})$ and $W_{M/G/1}^*(s) \mid_{B^*(s)=G(B^*(s)V^*(s))}$ is the LST of the pdf of the packet waiting time for the M/G/1 queue in which $B^*(s)$ is replaced by $G(B^*(s)V^*(s))$. The mean and the variance are

$$E[W_g] = \frac{E[V_0^2]}{2E[V_0]} + E[W_{M/G/1} \mid_{B^*(s)=G(B^*(s)V^*(s))}], \tag{3.32}$$

$$Var[W_g] = E[W_g^2] - (E[W_g])^2, \tag{3.33}$$

where

$$\begin{aligned} E[W_g^2] &= \frac{E[V_0^3]}{3E[V_0]} + \frac{E[V_0^2]}{E[V_0]} E[W_{M/G/1} \mid_{B^*(s)=G(B^*(s)V^*(s))}] \\ &\quad + E[W^2_{M/G/1} \mid_{B^*(s)=G(B^*(s)V^*(s))}]. \end{aligned} \tag{3.34}$$

Again define the response time of a batch to be the time interval starting from the arrival of a batch until the service completion of the last packet in the batch. The response time of a batch is composed of two independent components. One is the waiting time of the batch, and the other is the service time of all the packets in the batch. Therefore we have the LST of the pdf of the response time for a batch

$$T_g^*(s) = W_g^*(s) B_g^*(s), \tag{3.35}$$

and its moments

$$E[T_g] = E[W_g] + E[B_g], \tag{3.36}$$

$$Var[T_g] = E[T_g^2] - (E[T_g])^2, \tag{3.37}$$

where $E[T_g^2]$ is given by

$$E[T_g^2] = E[W_g^2] + 2E[B_g]E[W_g] + E[B_g^2], \tag{3.38}$$

and from equation (3.30)

$$E[B_g] = \overline{g}(\overline{b} + \overline{v}) - \overline{v}, \tag{3.39}$$

$$\begin{aligned} E[B_g^2] &= (\overline{g^{(2)}} - \overline{g})(\overline{b} + \overline{v})^2 + \overline{g}(\overline{b^2} + 2\overline{b}\overline{v} + \overline{v^2}) \\ &\quad - 2\overline{v}(\overline{g}(\overline{b} + \overline{v}) - \overline{v}) - \overline{v^2}. \end{aligned} \tag{3.40}$$

Again, if we let $V_0^*(s) = V^*(s)$, equations (3.31) and (3.32) become

$$W_g^*(s) = \frac{1 - V^*(s)}{sE[V]} \cdot \frac{s(1-\rho')}{s - \lambda + \lambda G(V^*(s)B^*(s))}, \tag{3.41}$$

where $\rho' = \lambda\, \overline{g}(\overline{v} + \overline{b})$, and

$$E[W_g] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1} \mid_{B^*(s)=G(B^*(s)V^*(s))}]. \tag{3.42}$$

They are exactly the same as the results in [29]. On the other hand, if we let $V^*(s) = 1$ ($\tilde{v} = 0$), equations (3.31) reduces to

$$W_g^*(s) = \frac{1 - V^*(s)}{sE[V]} W_{M/G/1}^*(s) \mid_{B^*(s)=G(B^*(s))}, \tag{3.43}$$

which is exactly the same as equation (2.41) in chapter 2 for the M/G/1 system with independent vacations and batch arrivals.

3.6 Summary

The M/G/1 queueing system with non-independent vacations was studied in this chapter. The PGF of the queue length distribution, and the LST of the pdf of the waiting time and the response time are obtained. It was shown that the delay distribution in the queue with vacations is composed of the sum of two independent variables: 1) the delay suffered due to the residual life of the vacation period $\tilde{v}_0$. 2) the packet waiting time for the M/G/1 queue in which the $B^*(s)$ is replaced by $B^*(s)V^*(s)$. The analysis was done for both the pure Poisson arrival and the batch Poisson arrival.

Chapter 4

An Exact Analysis of TDMA Systems

In this chapter we study the queueing behavior of the time division multiple access (TDMA) scheme, a method that carries out the sharing of the transmission channel by making a deterministic sequential allocation of time intervals (slots) to each station. The TDMA scheme can be considered as a method for synchronizing the transmission of different stations in the system. For this reason, the transmission of one station does not depend on the behavior of other stations, and this makes the analysis simpler. Two types of data arrivals, single packets and messages are considered in this chapter. The M/G/1 queue with non-independent vacations will be used to model the system, so the results derived in chapter 3 are used throughout this chapter.

4.1 Introduction

In a TDMA system [5] [19, chpt 5] [47, sct 2.5] [48], each station is assigned a fixed time duration (a time slot) on the communication channel for the transmission of its packets. After one station's time duration has elapsed, the channel is switched to another station with a synchronous operation.

TDMA systems were previously studied in [5]. The model permits a general distribution for the number of packets arrived within a time slot. The PGF of the queue size (in number of packets) at time instants just prior to the beginning of each time slot was solved. The mean delay experienced by a virtual message arrival was also obtained. By assuming Poisson message arrivals and employing a model of the M/G/1 queue in which the first packet of each busy period receives an exceptional service, results were obtained in [30] for the steady state PGF of the queue size (in number of messages) as seen by a random observer as well as the mean delay actually experienced by messages. In parallel, the same results were obtained in [18] by using another analytical approach. The delay performance of TDMA systems were also analyzed in [7] and [38] for stations with infinite buffer capacities and in [50] for stations with finite buffer capacities.

4.2 Model and Assumptions

The system considered in this chapter consists of M stations. The time is slotted and time slots, one for each station, are organized into frames of, say, M slots indexed from 1 to M as shown in figure 4.1. In every frame, each station is allocated a time slot. Time slots with the same index in consecutive frames form a TDMA channel. Let the duration of a frame be T seconds. A station uses a slot of T/M seconds (provided the queue is non-empty) to transmit a packet; it then has to wait for $(M-1)T/M$ seconds before it can transmit another packet. Packet transmissions can only happen at the beginning of the particular slot and a packet corresponds to the amount of data that can be transmitted into a slot.

It is assumed that data packets (or messages) arrive at a station according to a Poisson process. Arrival packets are stored in an infinite buffer until the portion of a frame (slot) dedicated to the station is available. Messages are stored in their entirety and broken into packets before they are transmitted. We imagine this as the station being visited by a server. The server provides service during its visit to the station with a constant service time of M/T seconds if the queue is not empty;

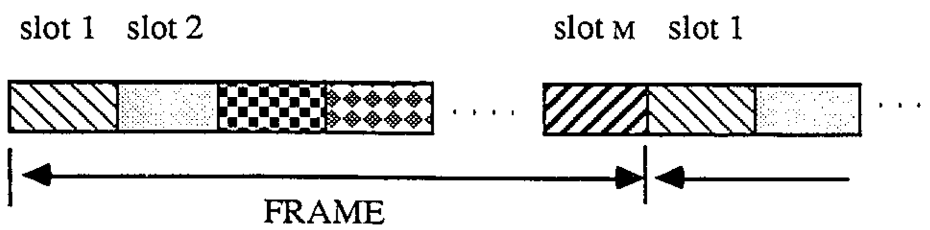


Figure 4.1: The TDMA channel

otherwise the server goes on vacation. Each time after serving a packet, the server will take a vacation $V = (M-1)T/M$ seconds. If there are no packets waiting when the server returns from a vacation, it takes another vacation $V_0 = T$ seconds (i.e, the time slot which belongs to the station is not used and considered as part of the vacation). Vacations are repeated until there is at least one packet found at the end of the vacation. The functional relationship is illustrated in figure 4.2. Note that the

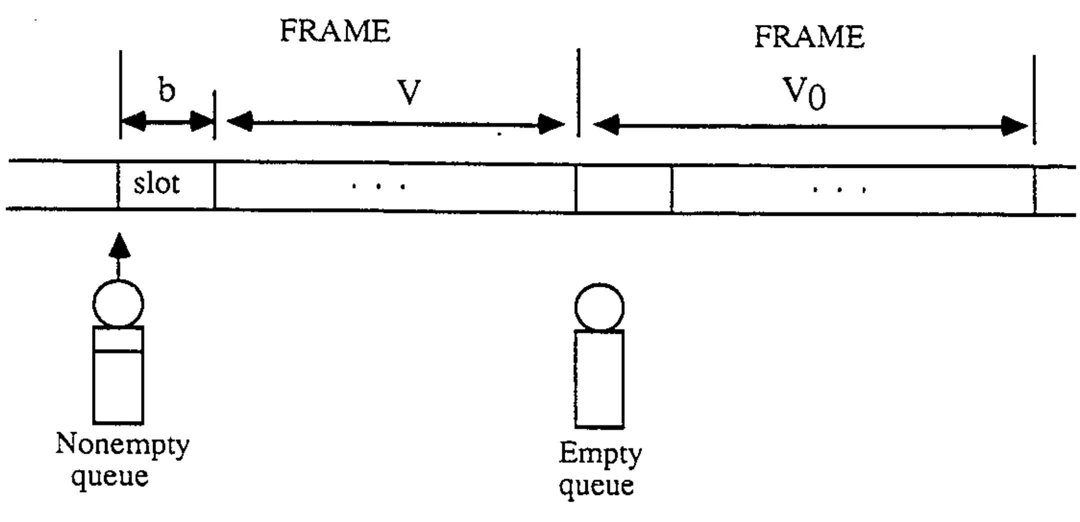


Figure 4.2: The vacation model of a station using a TDMA channel

action that the server takes for one station does not depend on the states of other stations, but depends on the state of that station itself. More important, observe that the vacation taken, i.e., V or V_0, would depend on the queue state. Also the time required for the server to cycle through all stations is a constant. Therefore the stations may be analyzed independently. For this reason, we shall decompose a TDMA system into M independent stations and study the performance of the TDMA system through the analysis of one of the M stations. The station utilizing a TDMA channel can be modelled by the queue with non-independent vacations.

It should be pointed out that in the models used in [18] and [30], the service time is not defined as the one used in our model. The service time of a packet in [18] and [30] was defined to be the sum of the time during which the packet is at the head of the queue with no transmission in progress and its transmission time of T/M seconds. Thus, in those models, the service time of a packet is exactly equal to T seconds which is the duration of a TDMA frame except for the packet which arrives to find an empty system; the service time of such packets is a random variable distributed between T/M and $T + T/M$ seconds. As a result, the waiting time of a packet obtained in the analysis here will be different from those in [18] [30]. Also high moments of the response time and the waiting time are not provided in those analysis due to the difficulty, but they can easily be derived by using our model.

4.3 Packet Delay Analysis

Consider a station in the TDMA system as a single-server queue with Poisson arrivals at a rate of λ packet per second. From the model, the service time of a packet is a constant of T/M seconds. Introduce the LST of the pdf of the service time, we have

$$B^*(s) = \int_0^\infty e^{-st} dB(t) = e^{-sT/M}, \tag{4.1}$$

and its first and ith moments

$$\overline{b} = \frac{T}{M}, \tag{4.2}$$

$$\overline{b^i} = \frac{T^i}{M^i}. \tag{4.3}$$

The vacation period initiated by an empty queue, $\tilde{v}_0$, is a constant of T seconds, while the vacation period following a service of a packet, $\tilde{v}$, is $(M-1)/MT$ seconds. Their LST and moments are given by

$$V_0^*(s) = \int_0^\infty e^{-st} dV_0(t) = e^{-sT}, \tag{4.4}$$

$$V^*(s) = \int_0^\infty e^{-st} dV(t) = e^{-(M-1)sT/M}, \tag{4.5}$$

$$E[V_0] = T, \tag{4.6}$$

$$E[V] = \frac{(M-1)T}{M}, \tag{4.7}$$

$$E[V_0^i] = T^i, \tag{4.8}$$

$$E[V^i] = \frac{((M-1)T)^i}{M^i}. \tag{4.9}$$

Let $\tilde{L}(t)$ be the number of packets in the station (both in the queue and in service) at time t. Denote the steady state probability of having n packets in the queue as

$$q_n = \lim_{t \to \infty} Prob[\tilde{L}(t) = n], \tag{4.10}$$

and its PGF

$$Q(z) = \sum_{n=0}^\infty q_n z^n. \tag{4.11}$$

We now proceed to find the steady state PGF of q_n by using the queue with non-independent vacations model. Repeating equation (3.7) derived in chapter 3, the queue length distribution is,

$$Q(z) = \frac{1 - \lambda(\bar{b} + E[V])}{E[V_0]} \frac{B^*(\lambda - \lambda z)(V_0^*(\lambda - \lambda z) - 1)}{z - V^*(\lambda - \lambda z) B^*(\lambda - \lambda z)}, \tag{4.12}$$

and accordingly the LSTs of the pdf of the waiting time of a packet and the response time of a packet are

$$W^*(s) = \frac{1 - \lambda(\bar{b} + E[V])}{E[V_0]} \frac{1 - V_0^*(s)}{s - \lambda + \lambda V^*(s) B^*(s)}, \tag{4.13}$$

$$T^*(s) = W^*(s) B^*(s). \tag{4.14}$$

Moments of the response time and the waiting time of a packet can be obtained in a similar manner as before. From equations (3.11) and (3.13), we have the mean waiting time of a packet

$$E[W] = \frac{E[V_0^2]}{2E[V_0]} + \frac{\lambda(E[V^2] + 2\bar{b}E[V] + \overline{b^2})}{2(1-\lambda(\bar{b} + E[V]))}, \tag{4.15}$$

and its variance

$$Var[W] = E[W^2] - (E[W])^2, \tag{4.16}$$

where

$$\begin{aligned} E[W^2] \;=\; & \frac{\lambda^2(E[V^2] + 2\bar{b}E[V] + \overline{b^2})^2}{2(1-\lambda(\bar{b}+E[V]))^2} \\ & + \frac{\lambda(E[V^3] + 3\bar{b}E[V^2] + 3E[V]\overline{b^2} + \overline{b^3})}{3(1-\lambda(\bar{b}+E[V]))} \\ & + \frac{\lambda(E[V^2] + 2\bar{b}E[V] + \overline{b^2})}{2(1-\lambda(\bar{b}+E[V]))}\frac{E[V_0^2]}{E[V_0]} + \frac{E[V_0^3]}{3E[V_0]}. \end{aligned} \tag{4.17}$$

The mean response time of a packet in the station can be obtained from equation (3.15)

$$E[T] = \frac{E[V_0^2]}{2E[V_0]} + \frac{\lambda(E[V^2] + 2\bar{b}E[V] + \overline{b^2})}{2(1-\lambda(\bar{b} + E[V]))} + \bar{b}, \tag{4.18}$$

and its variance

$$Var[T] = E[T^2] - (E[T])^2, \tag{4.19}$$

where

$$E[T^2] = E[W^2] + 2\bar{b}E[W] + \overline{b^2}. \tag{4.20}$$

Now substituting equations (4.2) to (4.9) into equations (4.15) to (4.20), we have the following results for packet delay performances:

$$E[W] = \frac{T}{2} + \frac{\lambda T^2}{2(1-\lambda T)} = \frac{T}{2(1-\lambda T)}, \tag{4.21}$$

$$E[T] = \frac{T}{2} + \frac{\lambda T^2}{2(1-\lambda T)} + \frac{T}{M}, \tag{4.22}$$

$$E[W^2] = \frac{\lambda^2 T^4}{2(1-\lambda T)^2} + \frac{\lambda T^3}{3(1-\lambda T)} + \frac{\lambda T^3}{2(1-\lambda T)} + \frac{T^2}{3}, \tag{4.23}$$

$$E[T^2] = (\frac{T}{M})^2 + \frac{T^2}{M(1-\lambda T)} + E[W^2]. \tag{4.24}$$

4.4 Message Delay Analysis

In general messages arrived at a station are random in length. Each message has to be broken into fixed-size packets before it is transmitted. Due to the operation of the TDMA scheme, the message delay performance will be different from the packet delay performance. This is because only one packet per slot is transmitted. If a data message has more than one packets, the message can only be transmitted after several frame cycles.

Consider a station in a TDMA system has data messages arrival according to a Poisson process. The number of packets required to dispatch a message is given by a general PGF

$$G(z) = \sum_{k=1}^{\infty} g_k z^k, \tag{4.25}$$

its mean $\overline{g}$ and its second moment $\overline{g^2}$. Recall all the notations used before, $B^*(s)$ be the LST of the pdf of the service time of a packet, $V_0^*(s)$ be the LST of pdf of the vacation period initiated by an empty queue and $V^*(s)$ be the LST of the pdf of the vacation period following the service of a packet.

Since the service time of a message is the aggregate service time of its constituent packets and the service time of a packet is independent of the vacation duration, the service time of a message can be obtained by the same way as to obtain the service time of a batch in chapter 3. From the derivation there, we have the LST of the pdf of the service time of a message

$$B_g^*(s) = \frac{G(B^*(s)V^*(s))}{V^*(s)}. \tag{4.26}$$

Define the waiting time of a message to be the time lapse measured from the arrival of the message until the start of the service of its first packet. Applying the results of the queue with non-independent vacations with batch arrivals in equation (3.31), we have the LST of the pdf of the message waiting time to be

$$W_g^*(s) = \frac{1 - V_0^*(s)}{sE[V_0]} \cdot \frac{s(1-\rho')}{s - \lambda + \lambda G(V^*(s)B^*(s))}, \tag{4.27}$$

where $\rho' = \lambda\overline{g}(\overline{v}+\overline{b})$. The mean and the variance can be obtained from

$$E[W_g] = \frac{E[V_0^2]}{2E[V_0]} + E[W_{M/G/1}\mid_{B^*(s)=G(B^*(s)V^*(s))}], \tag{4.28}$$

$$Var[W_g] = E[W_g^2] - (E[W_g])^2, \tag{4.29}$$

where

$$\begin{aligned} E[W_g^2] &= \frac{E[V_0^3]}{3E[V_0]} + \frac{E[V_0^2]}{E[V_0]}E[W_{M/G/1}\mid_{B^*(s)=G(B^*(s)V^*(s))}] \\ &\quad + E[W^2_{M/G/1}\mid_{B^*(s)=G(B^*(s)V^*(s))}]. \end{aligned} \tag{4.30}$$

The response time of a message is defined to be the time interval measured from the arrival of the message until the service completion of its last packet. So the response time of a message is the sum of the waiting time of the message and the service time of the message. Its LST and moments can be obtained as

$$T_g^*(s) = W_g^*(s)B_g^*(s) = W_g^*(s)\frac{G(B^*(s)V^*(s))}{V^*(s)}, \tag{4.31}$$

$$E[T_g] = E[W_g] + E[B_g], \tag{4.32}$$

$$Var[T_g] = E[T_g^2] - (E[T_g])^2, \tag{4.33}$$

and

$$E[T_g^2] = E[W_g^2] + 2E[B_g]E[W_g] + E[B_g^2], \tag{4.34}$$

where the first and the second moments of the service time of a message are given respectively by

$$E[B_g] = \overline{g}(\overline{b}+\overline{v}) - \overline{v}, \tag{4.35}$$

$$\begin{aligned} E[B_g^2] &= (\overline{g^2}-\overline{g})(\overline{b}+\overline{v})^2 + \overline{g}(\overline{b^2}+2\overline{b}\overline{v}+\overline{v^2}) \\ &\quad -2\overline{v}(\overline{g}(\overline{b}+\overline{v})-\overline{v}) - \overline{v^2}. \end{aligned} \tag{4.36}$$

Using the moments of the vacation time and the packet service time given before, we have the following results:

$$E[W_g] = \frac{T}{2} + \frac{\lambda\overline{g^2}T^2}{2(1-\lambda\overline{g}T)}, \tag{4.37}$$

$$E[T_g] = \frac{T}{2} + \frac{\lambda \overline{g^2} T^2}{2(1-\lambda \overline{g} T)} + \overline{g} T - \frac{(M-1)T}{M}, \tag{4.38}$$

$$E[W_g^2] = \frac{T^2}{3} + TE[W_g] + \frac{\lambda^2 \, (\overline{g^2} T^2)^2}{2(1-\lambda \overline{g} T)^2} + \frac{\lambda(\overline{g^3} T^3)}{3(1-\lambda \overline{g} T)}, \tag{4.39}$$

$$E[T_g^2] = E[W_g^2] + \frac{2\overline{g} T E[W_g]}{M} + \overline{g^2} T^2 + \frac{(M-1)^2 T^2}{M^2} + \frac{2\overline{g}(M-1)T^2}{M}. \tag{4.40}$$

4.5 Numerical Results

In this section we examine the delay performance trade-offs through some numerical examples. We consider a system with a population of $M = 10$. The slot time which is equal to the transmission time of the fixed length packet is normalized to one without loss of generality. Messages which arrive at the stations under consideration consist of two cases: single-packet messages corresponding to the performance of packet delay; multiple-packet messages in which the number of packets in a message form a geometric distribution with means of 2, 5, 10.

In order to evaluate the performance measures, mean response times are compared against both simulation results and the results presented in [30]; while the mean waiting time and the variance of the response time are evaluated by comparing them against simulation results only since no other results on variance have been reported so far. In the presentation of our results, all measured times have been normalized with respect to the slot time.

A simulation program has been developed by using the QNAP2 simulation software package [1]. The simulation program is attached in appendix A. In QNAP2, parameters such as delays and queues can be represented by the control command called "stations". Packets are routed through a system of stations consisting of servers and related queues according to predefined transitions and service disciplines. In the simulation of TDMA system, the generation of the frame is modelled by the use of a single server successively passing the transmission permission to every station as shown in figure 4.3. When a station receives the permission and has packets queued, it transmits right away. When a station receives its permission and has no packet in its queue, it lets the time slot pass by without being used. Packets arriving during this time slot are not allowed to transmit. After its slot has gone by, the station, no matter whether it has transmitted or not, must wait for the same slot in the next frame. After having passed the slot to every station (i.e., from 1 to M), the single server station will have to restart a frame by passing the transmission permission to station 1 again. Note that for each simulation run, the simulation time is chosen to have a convergent result, and a 95 % confidence interval is also provided automatically at the end of the simulation.

A more detailed discussion of results follows. Figures 4.4 and 4.5 show two examples of normalized mean response time for packet transmissions and message transmissions with respect to utilizations which are defined as $\rho = \lambda\overline{g}(\overline{v} + \overline{b})$. They also include results obtained from [30]. The results from our model, Lam's model and the simulations agree with each other very well. Results are also obtained for other system sizes such as $M = 20$, and different message sizes such as $\overline{g} = 2$, and $\overline{g} = 10$. They are not provided here for reasons of simplicity. From all these examples we observed that the proposed analytical approach gives an exact performance of TDMA systems.

For both the packet response time and the message response time, figures 4.6 to 4.8 show their variances against utilizations ρ. Since no other analytical results are available from Lam's model, simulations are used for verification, and the results are found to match very closely. This further enhances the confidence in our model.

Figure 4.9 shows the normalized mean packet waiting time versus utilizations ρ. Figure 4.10 presents the mean message waiting time versus utilizations ρ. Both are for the system of $M = 10$ stations. Simulation results with 95% confidence intervals are used to verify these results. The comparison shows a striking agreement for all cases. Based on these, more performance studies are carried out as follows.

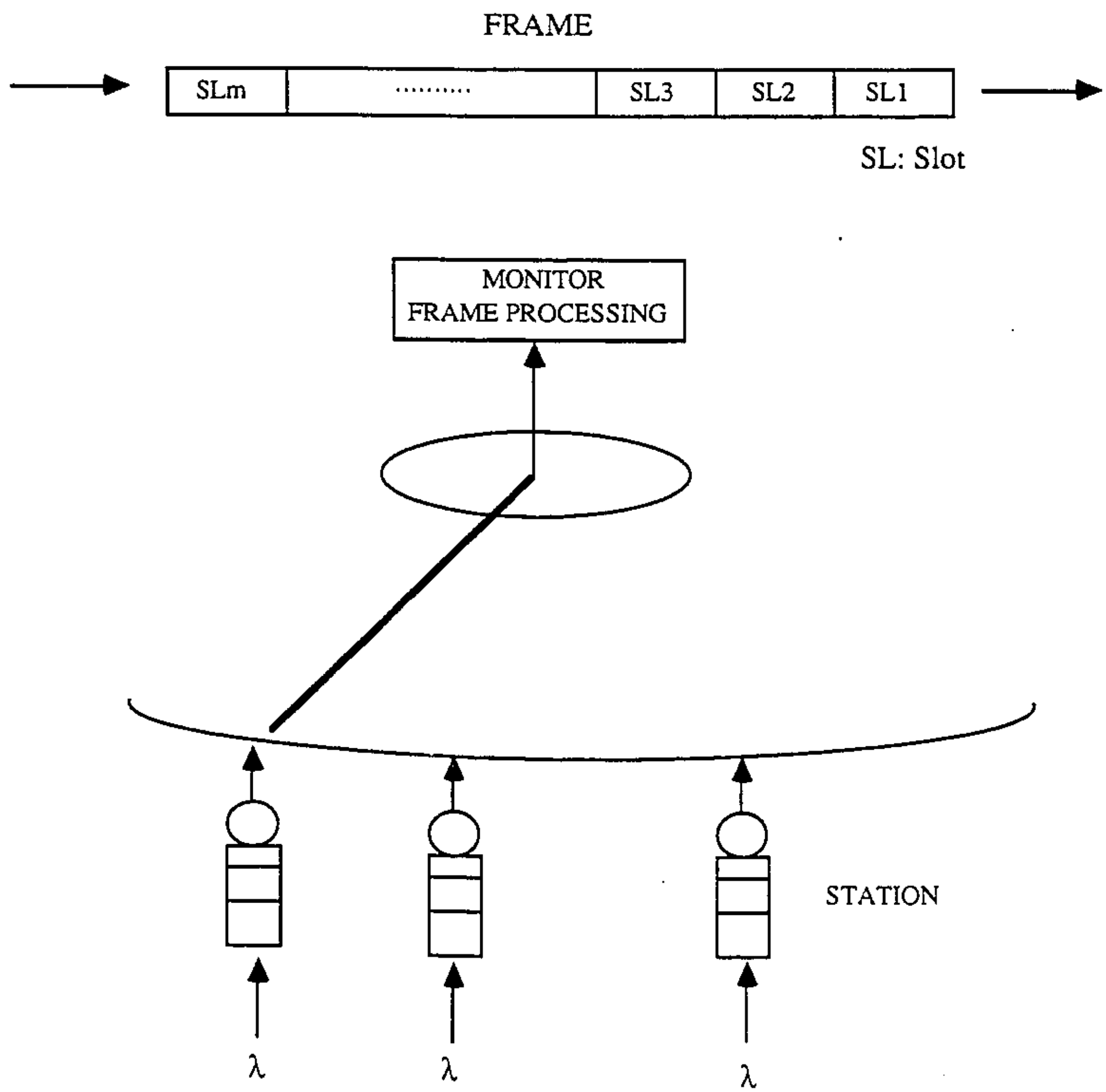


Figure 4.3: Simulation model for a TDMA system.

Figure 4.11 shows the comparison of the mean packet waiting time for $M = 10$ and $M = 20$. Since the slot size is kept the same, the system with a larger M will have a larger delay. In figure 4.12, we show the effect of the message size on the delay performance. In this comparison, we use three different message lengths, namely single packet message, a multiple-packet message in which the number of packets forms a geometric distribution with a mean of 5 and a multiple-packet message with a mean of 10. As expected, the larger the message size is, the larger the mean waiting time will be.

4.6 Summary

In this chapter we have studied the delay performance of TDMA systems by employing the decomposition method. Exact expressions of the LSTs of the pdf of the waiting time, the response time and their moments have been obtained by modelling a station in the system as a queue with non-independent vacations for both the packet and the message transmissions. The mean packet response time and the mean message response time were compared against the results obtained previously by using a different approach as well as simulation results and found to match exactly. The mean waiting time and the variance of the response time were also obtained and validated by simulations. As a result, the proposed analytical approach can give an exact analysis of TDMA systems.

Normalized mean packet response time for M = 10				
	Our Model	Lam's	Simulation	95% interval
utilization	response time	response time	response time	
0.1	6.555	6.555	6.552	0.083
0.2	7.250	7.250	7.240	0.085
0.3	8.143	8.143	8.136	0.096
0.4	9.333	9.333	9.309	0.121
0.5	11.00	11.00	11.08	6.552
0.6	13.50	13.50	13.57	0.273
0.7	17.67	17.67	17.67	6.552
0.8	26.00	26.00	26.18	1.079
0.9	51.00	51.00	51.29	3.220

Figure 4.4: Mean packet response time versus utilizations.

Normalized mean message response time for M = 10 and g = 5				
	Our Model	Lam's	Simulation	95% interval
utilization	response time	response time	response time	
0.1	51.00	51.00	50.68	1.33
0.2	57.25	57.25	57.28	1.25
0.3	65.29	65.29	65.77	1.63
0.4	76.00	76.00	76.06	1.75
0.5	91.00	91.00	90.88	2.31
0.6	113.5	113.5	113.9	3.72
0.7	151.0	151.0	151.8	6.73
0.8	226.0	226.0	225.5	14.26
0.9	451.0	451.0	456.7	18.95

Figure 4.5: Mean message response time versus utilizations for $\overline{g} = 5$.

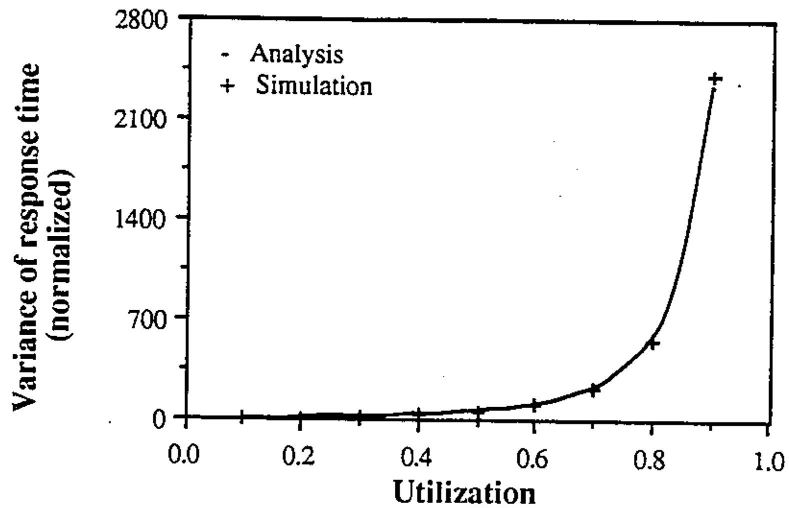


Figure 4.6: Variance of packet response time versus utilizations.

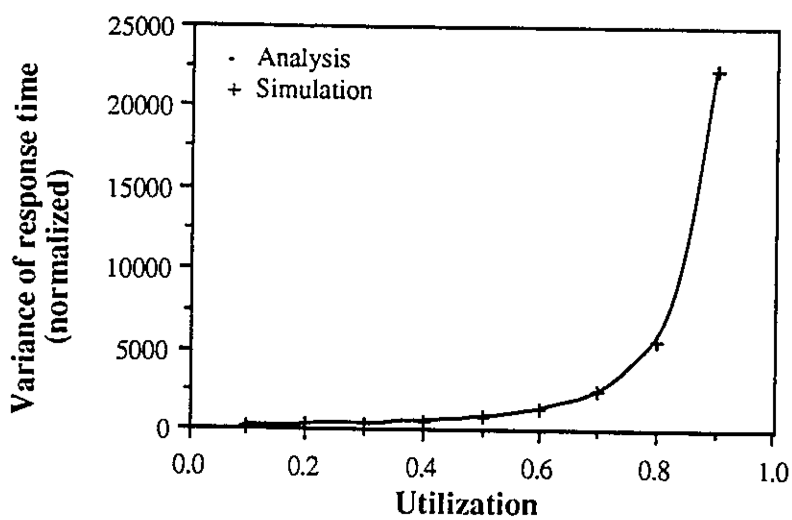


Figure 4.7: Variance of message response time vs utilizations for $\overline{g} = 2$.

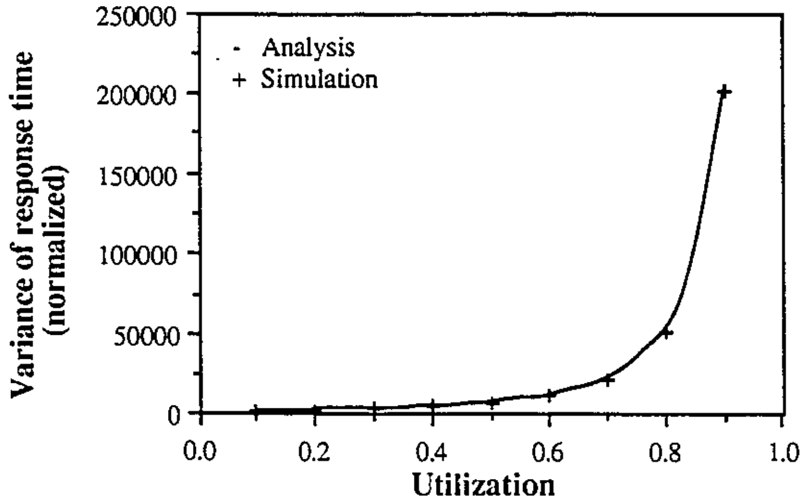


Figure 4.8: Variance of message response time vs utilizations for $\overline{g} = 5$.

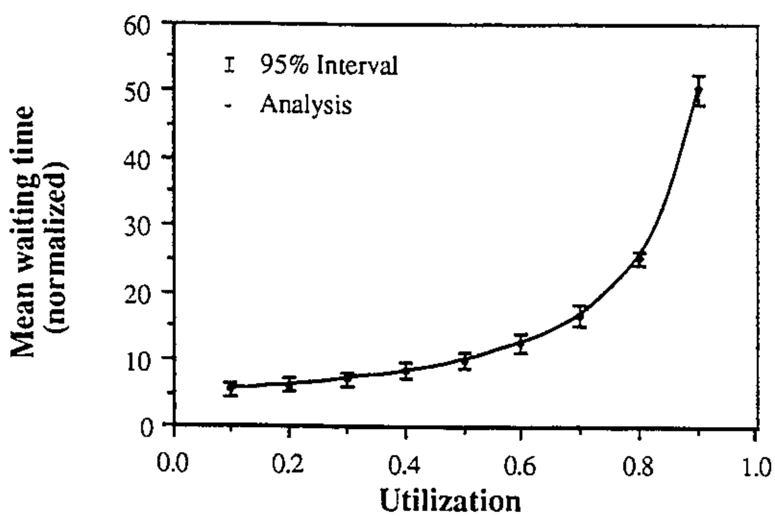


Figure 4.9: Mean packet waiting time versus utilizations.

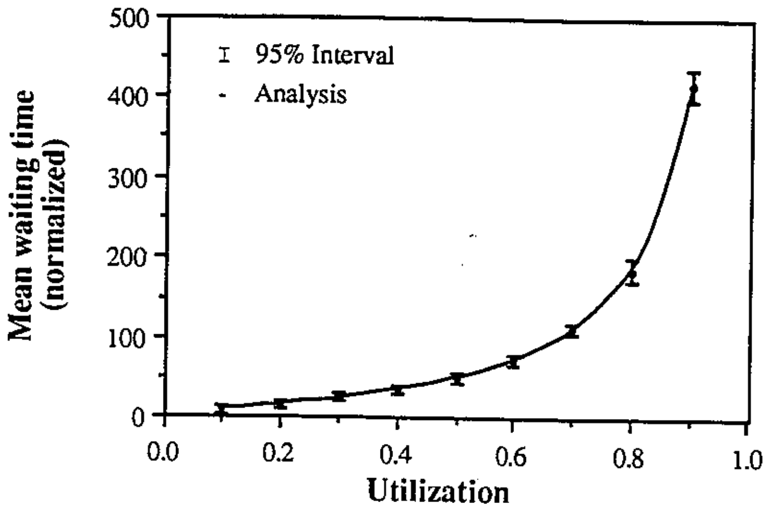


Figure 4.10: Mean message waiting time vs utilizations for $\overline{g} = 5$.

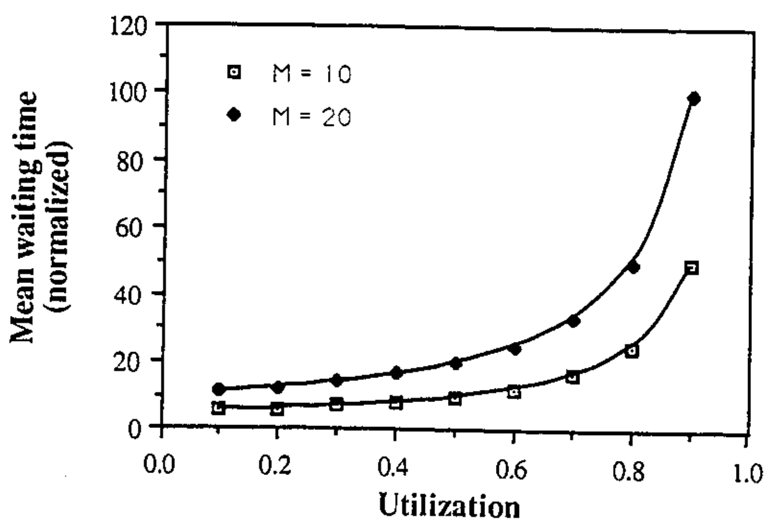


Figure 4.11: The effect of M on mean packet waiting time

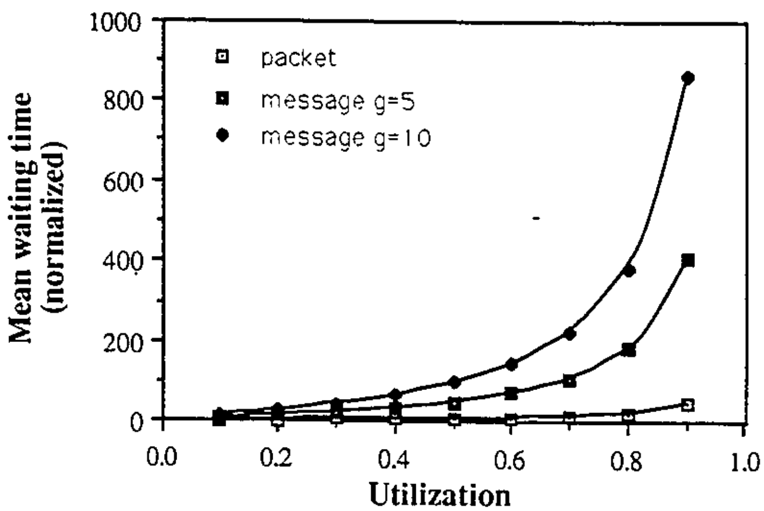


Figure 4.12: The effect of message sizes on mean waiting time for M = 10.

Chapter 5

Token Ring Analysis

In this chapter we study the performances of token ring systems. The token ring systems considered here consist of M stations, each of them assumed to have an infinite buffer. Two token holding policies namely, an exhaustive service policy and a single service policy, are analyzed in this chapter. For each service policy, two types of data arrivals, single packets and multi-packet messages, are assumed. The analysis is based on the results obtained in chapter 2 and chapter 3.

5.1 Introduction

The operation of a token ring system is well-known [3] [47, sct 3.4]. Basically access to the transmission channel is controlled by passing around the ring a special signal called the permission token. When a system is initialized, a designated station generates a token which travels around the ring until a station ready to transmit captures it and transmits its data. At the end of its transmission, the station passes the access permission to the next station downstream by generating a new token.

In the standard token ring protocols [3], the length of the time a station is allowed to transmit when in possession of a token is controlled by a so-called token-holding timer. Long token-holding timeouts lead to an exhaustive service policy [3] in one extreme where a station holding the token transmits all packets queued up. On the other hand, short token-holding timeouts yield an operation that may give rise to a single service policy [3], where only a single packet per token is transmitted. Therefore, both of these limiting cases are of interest for the performance evaluation of token rings.

There exists a lot of work in the analysis of token ring systems or their variation polling systems in the literature. For example, the book by Takagi [46] is a tutorial of the analysis of polling systems and contains a list of references. For single buffer systems, analysis results can be found in [17], [22], and [45]. These systems are not realistic as no queueing is allowed.

For infinite buffer systems, three types of service policies have been considered: 1) an exhaustive service, 2) a gated service, and 3) a single service. The analysis of the exhaustive service system with M stations and random switch-over times is available in [10], [12] and [16]. The model of the gated service system with M stations and random switch-over times was studied in [12] and [33]. For the single service system, the mean waiting time for M identical stations with zero switch-over time is given in [44]; approximation with M nonidentical stations and random switch-over times is given in [8], [16], [20] and [29], but no exact analytical results are available for the general case.

Among all the analysis, the M/G/1 vacation model has been used to analyze the performance of timed token protocols, especially in those approximations of the single service policy, because it captures some key features of the token control operations. However, most analyses are dealing with the packet delay performances [16] [20]. The main contribution of this study is to extend the packet delay performances to the message delay performances by using the batch arrival model. In particular, we studied the message response time performances for the single and the exhaustive service policies. It later came to our attention that the message delay performance was also reported last year in [8]. However in those analysis, only the waiting time was derived for the single service policy.

5.2 Model and Assumptions

The principle of the token ring protocol can be modelled by the queueing system as shown in figure 5.1. The active stations are represented by their transmitter queues. These queues are served in a cyclic manner symbolized by the rotating token.

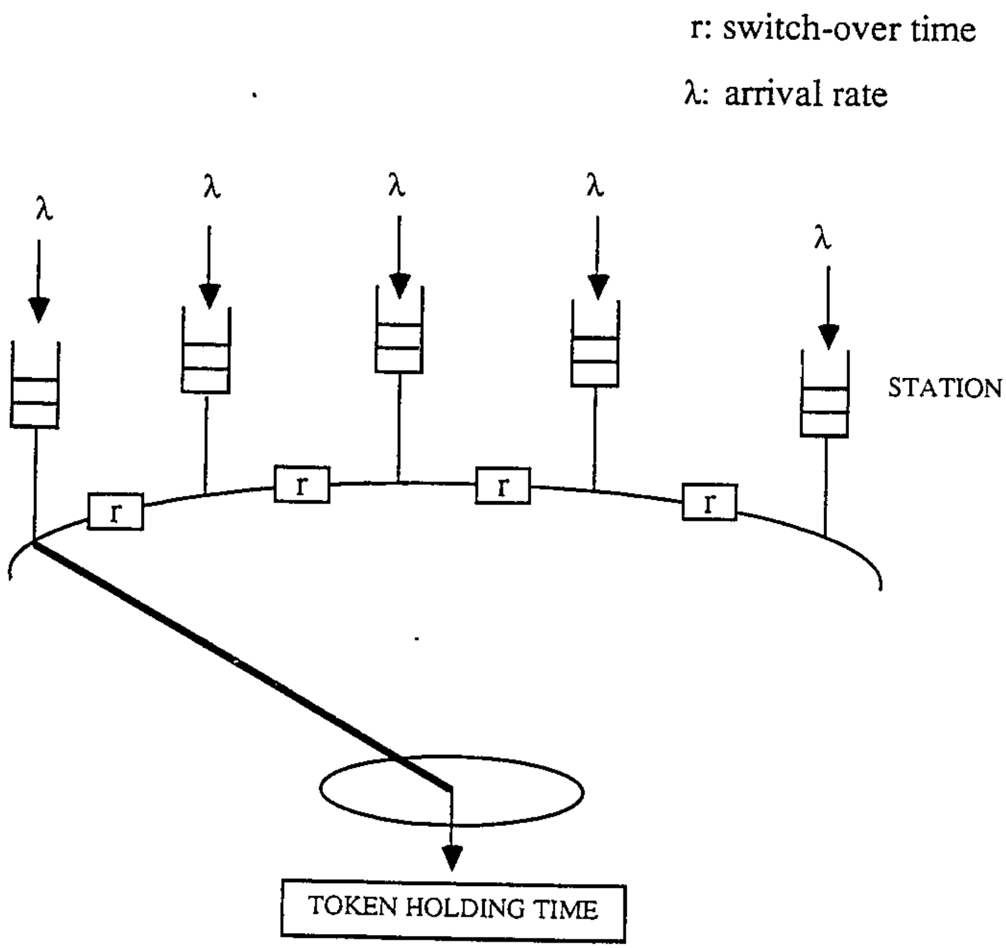


Figure 5.1: queueing system model for a token ring.

We consider a system with M stations, each with an infinite buffer. The server (token) serves M queues in a cyclic manner. After completing the service of a queue, the server incurs a switch-over period which in general can be a random length. During this period, none of the queues is served, and it may be considered as the time required to switch from one queue to the next. The switch-over time is also called the walk time in the literature. We shall decompose the system into M independent stations and analyze the delay performance of one station on a token ring by employing the vacation model. To a selected station, the token can be viewed as a server who provides service during its visit and goes on vacation when it is away. Two types of service policies are chosen for analyses in this chapter: the exhaustive service policy, and the single service policy. Because, as mentioned earlier, they are the two limiting cases which are of interest for the performance evaluation of token ring systems.

In the following analysis, we assume a completely symmetrical system, i.e., each station has an identical $B^*(s)$, the LST of the pdf of the service time, an identical independent Poisson arrival process with an arrival rate λ. Let the mean service time be $\overline{b}$ and its second moment $\overline{b^2}$. The server switch-over time, which is the combination of the token propagation time between stations i and $(i+1)$ and the station latency within station i, is assumed to be an independent identical distributed random variable with a LST of $R^*(s)$, a mean $\overline{r}$, and a second moment $\overline{r^2}$. Also we assume that the time a station holds the token for the service of its packets in the queue is independent of the waiting time and the vacation period. Furthermore, the vacation period seen by a general station k in equilibrium is denoted as $\tilde{v}_k$, its LST as $V_k^*(s)$, its mean $\overline{v}_k$ and its second moment $\overline{v^2}_k$.

5.3 Exhaustive Service Systems

In an exhaustive service system, upon the receipt of a free token, a station is permitted to transmit all packets currently in its buffer, as well as all the new packets arriving at the station during its transmission. This operation can be modelled by a server that begins a vacation of a random length each time the queue becomes empty. If the server returns to find one or more packets waiting, it works until the system empties again before taking another vacation. If the server finds no packets waiting, it begins another vacation immediately. This is repeated until it finds at least one packet waiting upon returning from a vacation. We shall use the M/G/1 queue with independent vacations to analyze the packet delay performances. Although there is a certain dependency between two consecutive vacations, due to the operation of the token ring, vacations can still be considered independent in steady state.

5.3.1 Packet Delay Analysis

Let $\tilde{L}(t)$ be a random variable for the number of packets in the station at time t. Define the steady state probability

$$q_n = \lim_{t \to \infty} Prob[\tilde{L}(t) = n], \tag{5.1}$$

and its transform

$$Q(z) = \sum_{n=0}^{\infty} q_n z^n. \tag{5.2}$$

We shall obtain the generating function of q_n by using the queue with independent vacations.

Based on the analytical results for the queue with independent vacations, we have from equation (2.25) the PGF of the queue length distribution for a station in the exhaustive service system to be

$$Q(z) = \frac{1 - V^*(\lambda - \lambda z)}{\overline{v}(1 - z)} Q_{M/G/1}(z) \tag{5.3}$$

where $Q_{M/G/1}(z)$ is given in equation (2.9) for an M/G/1 queueing system. The mean queue length is given by

$$E[Q] = \lambda \left(\frac{E[V^2]}{2E[V]} + \frac{\lambda \overline{b^2}}{2(1 - \lambda \overline{b})} + \overline{b} \right). \tag{5.4}$$

Accordingly, the LST of the pdf of the packet waiting time for a station in the exhaustive service system is given by

$$W^*(s) = \frac{1 - V^*(s)}{s\overline{v}} W^*_{M/G/1}(s), \tag{5.5}$$

where $W^*_{M/G/1}(s)$ is given by equation (2.10). The LST of the pdf of the response time can be derived to be

$$T^*(s) = W^*(s)B^*(s). \tag{5.6}$$

Their means can be obtained like before,

$$E[W] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1}], \tag{5.7}$$

$$E[T] = E[W] + \bar{b}, \tag{5.8}$$

where $E[W_{M/G/1}]$ is given in equation (2.11).

5.3.2 Vacation Period Analysis

We now proceed to analyze $V_i^*(s)$, the LST for the pdf of the vacation period for station i. Since the system is completely symmetric, the vacation period for each station is the same and we can drop the subscript i and use $V^*(s)$ instead. The approach used in [46] for the analysis of the vacation period is adopted and summarized as follows.

Let $\tilde{N}_i(t)$ be the number of packets at station i at time $t = \tau_m(i)$ when the server arrives at station i for the mth time, and define the joint and marginal generating functions for $[\tilde{N}_1(t), \tilde{N}_2(t), \cdots, \tilde{N}_M(t)]$ at time t to be

$$F_i(z_1, z_2, \cdots, z_M) = E[\ \prod_{j=1}^{M} z_j^{\tilde{N}_j(t)}\ ], \tag{5.9}$$

$$F_i(z) = E[\ z^{\tilde{N}_i(t)}\ ] = F_i(1, \cdots, 1, z, 1, \cdots, 1). \tag{5.10}$$

Thus, at $t = \tau_m(i+1)$ when the server arrives at station $i+1$ for the mth time, we have the joint generating function $F_{i+1}(z_1, z_2, \cdots, z_M)$. The $F_i(z_1, z_2, \cdots, z_M)$ and $F_{i+1}(z_1, z_2, \cdots, z_M)$ can be related by

$$\begin{aligned} F_{i+1}(z_1, z_2, \cdots, \quad & z_M) = R^*(\sum_{j=1}^{M}(\lambda - \lambda z_j)\ ) \\ & F_i(z_1, \cdots, z_{i-1}, B^*(\sum_{j \neq i}^{M}(\lambda - \lambda z_j)), z_{i+1}, \cdots, z_M), \end{aligned} \tag{5.11}$$

where $R^*(\sum_{j=1}^{M}(\lambda - \lambda z_j))$ is the joint generating function for the number of packets arrived during the switch-time interval between station i and station $i+1$, and $B^*(\sum_{j\neq i}^{M}(\lambda-\lambda z_j))$ is the joint generating function (except for station i) for the number of arrivals during the service time for station i. Define

$$f_i(j) = \frac{\partial F_i(z_1, \cdots, z_M)}{\partial z_j} \Big|_{z_1=z_2=\cdots=z_M=1}, \tag{5.12}$$

and

$$f_i(j,k) = \frac{\partial^2 F_i(z_1, \cdots, z_M)}{\partial z_j \partial z_k} \Big|_{z_1=z_2=\cdots=z_M=1} . \tag{5.13}$$

By solving a set of equations for i, j, $k = 1, 2, \cdots, M$, we can obtain the following first and the second factorial moments of the probability distribution in steady state,

$$f_i(i) = \frac{M\lambda\overline{r}(1-\rho)}{1-M\rho}, \tag{5.14}$$

$$f_i(i,i) = \frac{M(1-\rho)\lambda^2\sigma_r^2}{1-M\rho} + \frac{M(M-1)\lambda^3\overline{b^2}\overline{r}}{(1-M\rho)^2} + \frac{M^2\lambda^2\overline{r}^2(1-\rho)^2}{(1-M\rho)^2}, \tag{5.15}$$

where $\rho = \lambda\,\overline{b}$, $\sigma_r^2 = \overline{r^2} - \overline{r}^2$.

Since the number of packets arrived during the vacation is the number of packets found at the instant when the server arrives at the station, we have the relation

$$V_i^*(\lambda - \lambda z) = F_i(z). \tag{5.16}$$

Consider that the system is symmetric, we can drop the subscript i and write

$$V^*(\lambda - \lambda z) = F(z), \tag{5.17}$$

from which we have

$$E[V] = \frac{M\overline{r}(1-\rho)}{1-M\rho}, \tag{5.18}$$

$$Var[V] = \frac{M(1-\rho)\sigma_r^2}{1-M\rho} + \frac{M(M-1)\lambda\overline{r}\overline{b^2}}{(1-M\rho)^2}. \tag{5.19}$$

By replacing E[V] and E[V^2] in equation (5.8), the mean response time can be obtained as

$$E[T] = \overline{b} + \frac{\sigma_r^2}{2\overline{r}} + \frac{M\overline{r}(1-\rho)}{2(1-M\rho)} + \frac{M\lambda\overline{b^2}}{2(1-M\rho)}, \tag{5.20}$$

where $\rho = \lambda\overline{b}$.

5.3.3 Message Delay Analysis

Consider that stations on the ring have data messages arrival according to the Poisson process. From each message, one or more packets are formed. Assume that the number of packets required to dispatch a message is given by a general PGF

$$G(z) = \sum_{k=1}^{\infty} g_k z^k, \tag{5.21}$$

its mean $\overline{g}$ and its second moment $\overline{g^2}$. Since in the exhaustive service policy, the station is permitted to transmit all packets currently in its buffer, as well as all the new packets arriving at the station during its transmission, and the service time of each packet in a message is independent, also the service time of a message is independent of the behaviors of other stations, the message delay performance can be analyzed by using the results of the queue with independent vacations and batch arrivals. Recall all the notations used before, $B^*(s)$ be the LST of the pdf of the service time of a packet, $B_g^*(s)$ be the LST of the pdf of the service time of a message, and $V^*(s)$ be the LST of the pdf of the vacation time.

From the assumption that the service time of each packet in a message is independent, each message can be viewed as a superpacket. Therefore the service time of a message can be obtained as the service time of a batch in chapter 2. From equation (5.21), we have the LST of the pdf of the service time of a message to be

$$B_g^*(s) = \sum_{k=1}^{\infty} (B^*(s))^k g_k = G(B^*(s)). \tag{5.22}$$

By using the results in chapter 2, the PGF of the queue length distribution in terms of the number of messages is readily available

$$Q_g(z) = \frac{q_0(1 - V^*(\lambda - \lambda z))G(B^*(\lambda - \lambda z))}{G(B^*(\lambda - \lambda z)) - z}, \tag{5.23}$$

where q_0 is given by

$$q_0 = \frac{1 - \lambda \overline{g} \overline{b}}{\overline{v}}. \tag{5.24}$$

The mean queue length is given by

$$E[Q_g] = \lambda \left(\frac{E[V^2]}{2E[V]} + \frac{\lambda E[B_g^2]}{2(1-\lambda E[B_g])} + E[B_g] \right), \tag{5.25}$$

where $E[B_g]$ and $E[B_g^2]$ can be obtained from equation (5.22)

$$E[B_g] = \overline{g}\overline{b} \tag{5.26}$$

$$E[B_g^2] = \overline{g}\overline{b^2} + (\overline{g^2} - \overline{g})\overline{b}^2. \tag{5.27}$$

Define the waiting time of a message to be the time lapse measured from the arrival of a message till the start of the service of the first packet in the message. From equation (2.41), we have the LST of the pdf for the waiting time of a message

$$W_g^*(s) = \frac{1 - V^*(s)}{sE[V]} \frac{s(1-\rho)}{s - \lambda + \lambda G(B^*(s))}, \tag{5.28}$$

where $\rho = \lambda \overline{g}\overline{b}$. The mean waiting time of a message can be computed as

$$E[W_g] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1} \mid_{B^*(s)=G(B^*(s))}]. \tag{5.29}$$

The response time of a message is defined to be the time interval starting from the arrival of a message until the service completion of the last packet of the message and it is the sum of the waiting time of the message and the service time of the message. As before, the LST of the pdf for the response time of a message is given by

$$T_g^*(s) = W_g^*(s) B_g^*(s), \tag{5.30}$$

and its mean is

$$E[T_g] = E[W_g] + E[B_g]. \tag{5.31}$$

The vacation time distribution for the message delay analysis can be analyzed by the same approach as in the packet delay analysis. The mean and the variance are given by

$$E[V] = \frac{M\overline{r}(1 - \lambda \overline{b}\overline{g})}{1 - M\lambda \overline{b}\overline{g}}, \tag{5.32}$$

$$Var[V] = \frac{M(1-\lambda\overline{b}\overline{g})\sigma_\tau^2}{1-M\lambda\overline{b}\overline{g}} + \frac{M(M-1)\lambda\overline{\tau}((\overline{g^2}-\overline{g})\overline{b}^2+\overline{g}\overline{b^2})}{(1-M\lambda\overline{b}\overline{g})^2}. \tag{5.33}$$

Replacing $E[V]$ and $Var[V^2]$ in equation (5.31), the mean response time of a message can be obtained as

$$E[T_g] = \frac{\sigma_\tau^2}{2\overline{\tau}} + \frac{M\overline{\tau}(1-\rho)}{2(1-M\rho)} + \frac{M\lambda((\overline{g^2}-\overline{g})\overline{b}^2+\overline{g}\overline{b^2})}{2(1-M\rho)} + \overline{g}\overline{b}, \tag{5.34}$$

where $\rho = \lambda\overline{g}\overline{b}$.

5.4 Single Service Systems

In a single service system, a station receiving a free token is allowed to transmit only one packet if the queue is non-empty. Then it has to pass the token to the next station. To a station on the ring, the token can be treated as a server which provides service during its visit and goes on vacation when it is away. If there are no packets waiting when the server (token) returns from a vacation, it will take another vacation. Otherwise, one packet will be served from the waiting queue. By assuming that the vacation period is independent of the queue length, we can use a queue with non-independent vacations to analyze the delay performances of a station on the token ring. Unlike the analysis in chapter 4, the vacation period initiated by an empty queue is the same as the one initiated by a non-empty queue, so the condition $V_0^*(s) = V^*(s)$ will be used in this study.

5.4.1 Packet Delay Analysis

As in the analysis of the exhaustive service system, let $\tilde{L}(t)$ be a random variable of the number of packets in the station at time t and

$$q_n = \lim_{t\to\infty} Prob[\tilde{L}(t) = n]. \tag{5.35}$$

Define the transform

$$Q(z) = \sum_{n=0}^{\infty} q_n z^n. \tag{5.36}$$

Recall equation (3.7) for the PGF of the queue length distribution of the system with non-independent vacations and let $V_0^*(s) = V^*(s)$, we have

$$Q(z) = \frac{1-\lambda(\bar{b}+E[V])}{E[V]} \frac{B^*(\lambda-\lambda z)(V^*(\lambda-\lambda z)-1)}{z-V^*(\lambda-\lambda z)B^*(\lambda-\lambda z)}, \tag{5.37}$$

and its mean

$$E[Q] = \lambda\left(\frac{E[V^2]}{2E[V]} + \frac{\lambda(\overline{b^2}+2\bar{b}E[V]+E[V^2])}{2(1-\lambda(\bar{b}+E[V]))} + \bar{b}\right). \tag{5.38}$$

Thus, the LST of the pdf of the packet waiting time is given as

$$W^*(s) = \frac{1-\lambda(\bar{b}+E[V])}{E[V]} \frac{1-V^*(s)}{s-\lambda+\lambda V^*(s)B^*(s)}. \tag{5.39}$$

Rearrange this equation we have

$$\begin{aligned} W^*(s) &= \frac{1-V^*(s)}{E[V]} \frac{1-\lambda(\bar{b}+E[V])}{s-\lambda+\lambda V^*(s)B^*(s)} \\ &= \frac{1-V^*(s)}{sE[V]} W^*_{M/G/1}(s) \mid_{B^*(s)=B^*(s)V^*(s)} . \end{aligned} \tag{5.40}$$

From that we have the mean

$$E[W] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1} \mid_{B^*(s)=B^*(s)V^*(s)}]. \tag{5.41}$$

The LST of the pdf of the response time and its mean can be obtained accordingly

$$T^*(s) = W^*(s)B^*(s), \tag{5.42}$$

$$E[T] = E[W] + \bar{b}. \tag{5.43}$$

5.4.2 Vacation Period Approximation

Due to the interdependency between the vacation duration and the queue length at each station on the ring, the analysis of the vacation time distribution becomes very difficult and no exact solution has been obtained yet. However, the LST of the pdf of the vacation time $V^*(s)$ seen by a station on a completely symmetric ring can be approximated by the independent assumption and obtained [16] as

$$V^*(s) = ((1-\alpha)+\alpha B^*(s))^{M-1}(R^*(s))^M. \tag{5.44}$$

In the equation, α is the probability that the server finds at least one packet at a station upon its arrival. There are several ways to approximate this α [8] [16] [20] [29]. Here we chose the one which gives the best approximation with respect to simulation results. That is [8]

$$\alpha = \frac{C_0 + \overline{r^2}/\overline{r}}{1 - \rho_T + \rho}\,\lambda, \tag{5.45}$$

where $\rho_T = M\rho = M\lambda\bar{b}$, and C_0 is a ring latency (i.e. the total switch-over time of the token). The moments of the vacation period can be obtained as

$$E[V] = M\overline{r} + (M-1)\alpha\bar{b}, \tag{5.46}$$

$$Var[V] = (M-1)(\alpha\overline{b^2} - (\alpha\bar{b})^2) + M\sigma_r^2. \tag{5.47}$$

By substituting equations into equations (5.39) to (5.43), the packet delay performances can be obtained.

5.4.3 Message Delay Analysis

Now we study the message delay performance as an extension of the packet delay performance. In the exhaustive service system, the long token holding time makes it possible to transmit all the packets in a message per token. However, in the single service system, only one packet per token is transmitted. If a data message has more than one packets, then the entire message can only be transmitted after several token rotation cycles. This makes the performance of the message delay different from the packet delay.

We shall model a station on the ring as a queue with non-independent vacations and batch arrivals. Recall all the notations used before, $B^*(s)$ be the LST of the pdf of the service time of a packet, $B_g^*(s)$ be the LST of the pdf of the service time of a message, and $V^*(s)$ be LST of the pdf of the vacation time. Following the same discussions as in previous chapters, we have the LST of the service time distribution of a message with its PGF $G(z)$ to be

$$B_g^*(s) = \frac{G(B^*(s)V^*(s))}{V^*(s)}. \tag{5.48}$$

Again, define the waiting time of a message to be the time lapse measured from the arrival of the message till the start of service of the first packets of the message. Under the assumption that the packet service time is independent of the vacation duration, applying the results of the queue with non-independent vacations with batch arrivals and letting $V_0^*(s) = V^*(s)$, we have the LST of the pdf of the message waiting time to be

$$W_g^*(s) = \frac{1 - V^*(s)}{E[V]} \cdot \frac{(1-\rho')}{s - \lambda + \lambda G(V^*(s)B^*(s))} \tag{5.49}$$

where $\rho' = \lambda \overline{g}(\overline{v} + \overline{b})$. Its mean is given by

$$E[W_g] = \frac{E[V^2]}{2E[V]} + E[W_{M/G/1} \,|_{B^*(s)=G(B^*(s)V^*(s))}]. \tag{5.50}$$

The message response time is defined to be the time interval starting from the arrival of a message until the service completion of the last packet of the message. The response time of a message is composed of two independent parts. One is the waiting time of the message, and the other is the service time of all the packets in the message. So we have the LST of the pdf of the response time of a message and its mean,

$$T_g^*(s) = W_g^*(s)B_g^*(s), \tag{5.51}$$

$$E[T_g] = E[W_g] + E[B_g], \tag{5.52}$$

where $B_g^*(s)$ is given in equation (5.48) and its mean is

$$E[B_g] = \overline{g}(\overline{b} + \overline{v}) - \overline{v}. \tag{5.53}$$

Note that the LST of the pdf of the vacation time used above is different from the one used in the packet delay analysis. However, the LST of the pdf of the vacation time $V^*(s)$ for the message delay analysis can be approximated by the same approach as in the packet delay analysis. The LST of the pdf of the vacation time $V^*(s)$ seen by a station on a completely symmetric ring can be approximated by,

$$V^*(s) = ((1-\alpha) + \alpha B^*(s))^{M-1}(R^*(s))^M, \tag{5.54}$$

as before, except α is now given by

$$\alpha = \frac{C_0 + \overline{r^2}/\overline{r}}{1 - \rho_T + \rho}\, \lambda \overline{g}, \tag{5.55}$$

where $\rho_T = M\rho = M\lambda\overline{g}\overline{b}$, and C_0 is a ring latency. Based on these, the message delay performances can be obtained.

5.5 Numerical Results

In this section we examine the delay performances of the token ring system through some numerical examples. A ring size of M = 20 stations is considered. The station-to-station switch-over times are assumed to be constant and all equal to C_0/M where C_0 is the total ring latency due to the switch-over time around the ring. By choosing a ring length of 40 km, and using a propagation speed of $5\mu s/km$, one obtains a ring latency C_0 of 0.2 ms. For simplicity, we have ignored the station latencies (in the order of μs) which are small in comparison with 0.2 ms.

Two service policies that were described in sections 5.3 and 5.4 are considered. For each service policy, both the packet delay performance and the message delay performance are evaluated.

In figures 5.2 and 5.3, the delay performances versus system utilizations are presented for a system with the exhaustive service policy. The system utilization is defined as $\rho = M\lambda\overline{b}\overline{g}$. In figure 5.2, we assume that the packet length forms an exponential distribution with a mean of 10 $k$bits. For a transmission rate of 100 $Mbps$, this corresponds to an exponential service time distribution with a mean of 0.1 ms. The mean packet response time versus system utilization is presented. Figure 5.3 shows the mean message response time versus system utilization. Here, a constant packet length of 10 $k$bits is used instead, and this gives a packet service time of 0.1 ms. The number of packets comprising a message is assumed to have a geometric distribution with a mean of $\overline{g} = 5$.

In figures 5.4 and 5.5, the delay performance versus station utilization is presented for a system with the single service policy. For the single service policy, the station utilization is defined as $\rho = \lambda(\overline{b}+\overline{v})$ for the packet delay performance and $\rho = \lambda\overline{g}(\overline{b}+\overline{v})$ for the message delay performance. Figure 5.4 shows the mean packet response time for the system with a constant packet service time of 0.1 ms and figure 5.5 gives the mean message response time performance for the system with a constant packet service time of 0.1 ms and a message with a geometric distribution and a mean of $\overline{g} = 5$.

Simulation programs have been developed by using the QNAP2 simulation software package and attached in appendices B1 and B2. Simulations have been run for systems with parameters described above. Different simulation time durations have been used for different utilization values to get the simulation result converged. The 95% confidence intervals have also been obtained and shown in figures 5.2 to 5.5. In every case, the simulation results agree very well with our analytical results.

Consequent to this good agreement, we may use our analytical results for further performance studies. First we study the effect of the number of stations on the delay performance. We increase the number of stations on a ring to $M = 40$, while keeping C_0 the same and a constant packet service time of 0.1 ms. The comparisons of $M = 20$ and $M = 40$ for the mean packet response time versus system utilizations for two service policies are presented in figures 5.6 and 5.7 respectively. From the figures we can find that for a given policy the number of stations on the ring does not affect the delay performance much so long the system utilization is kept the same. However in figure 5.7, we can find that for the same values of system utilizations $M = 20$ has larger delays than $M = 40$ when ρ is larger than 0.7. This is because for the same system utilization, the station on the ring with 20 stations will have a larger λ. This means it will have a larger station utilization and a longer delay. For the exhaustive service, the delay depends only upon the system utilization as shown in the analysis, therefore for the same ρ, we have the same delay.

In figures 5.8 and 5.9, we analyze the effect of the packet size on the packet delay performance. In order to have fair comparisons, the mean response time is

normalized with respect to the mean packet service time. Figure 5.8 shows the comparison for the single service policy while figure 5.9 is for the exhaustive service policy. From the comparison for both policies, we can see that the larger the packet size (the larger the mean packet service time), the smaller the mean response time. When the packet size is reduced to (or below) the value which is comparable to the switch-over time ($\bar{b} = 0.01$ in our case), the delay will increase very fast as shown in figures. This observation suggests that a relatively large packet size should be used for the token ring. This is because a large packet size can reduce the effect of the overhead caused by the ring latencies of token rotations.

The effect of the packet size on the message delay performance is also studied. In the study we use the same message. Its length has an exponential distribution with a mean of 50 $k$bits. Two cases were considered, 1) the message can be transmitted in one packet ($\bar{b} = 0.5$); 2) the message has to be packetized before it can be transmitted. In the second case, we use a constant packet size of 10 $k$bits which corresponds to $\bar{b} = 0.1$. So we have a $\bar{g} = 5$. The results are shown in figure 5.10 for the exhaustive service policy and in figure 5.11 for the single service policy. From the figure we can see that the packet size has little effect on the message delay performance for the exhaustive service policy while the packet size will affect the message delay performance greatly for the single service policy. This is because of the effect of the overhead caused by token rotations.

In figures 5.12 and 5.13, we examine the effect of the distribution of the number of packets which comprises a message for both the single and the exhaustive service policies. Two distributions were chosen, one is the geometric distribution with a mean of $\bar{g} = 2$ and the other is the uniform distribution. For the uniform distribution, we consider that messages arriving at the station consist of either one packet, two packets or three packets with an equal probability. Figure 5.13 shows the comparison between two distributions for the single service policy and figure 5.12 presents the one for the exhaustive service policy. From the comparison we can find that the uniform distribution has a smaller mean response time for both service policies. This is because the uniform distribution has a smaller variance for the same mean value.

Figures 5.14 to 5.16 show the comparison of the delay performances of the two service policies. In order to show fair comparisons, system utilizations are used. The system utilization was defined before $\rho = M\lambda\bar{b}\bar{g}$. Also for the packet delay performance, a constant packet length is used for the exhaustive service policy. It can be seen that the exhaustive service discipline can always perform better. This is because the exhaustive service policy reduces to a minimum the overhead caused by the ring latencies of token rotations.

5.6 Summary

The performances of a token ring network under the single and the exhaustive service policies have been analyzed. Closed form expressions for the packet delay and the message delay of a symmetric system under the exhaustive service policy have been presented. By employing the approximation of the vacation time distribution, the packet delay as well as the message delay expressions have also been obtained for a symmetric system under the single service policy. The analytical results were validated by simulations. The performance with respect to the number of stations on the ring, the effect of the packet size, and the effect of the message length distribution were also studied. Comparison between these two service policies was made. From the numerical results, the exhaustive service policy is found to always perform better than the single service policy. However note that a station in an exhaustive service system with heavy incoming traffics might monopolize the token, this may be unfair to other stations and cause dead-lock [3].

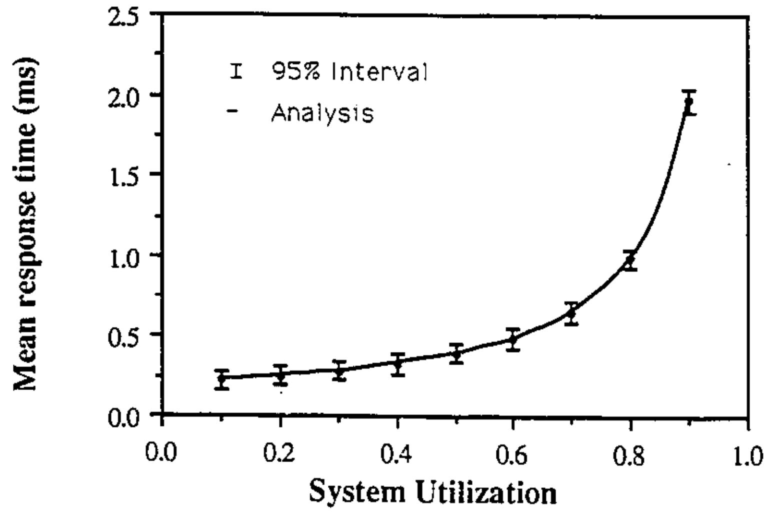


Figure 5.2: Mean packet response time for exhaustive service with $\overline{b} = 0.1$.

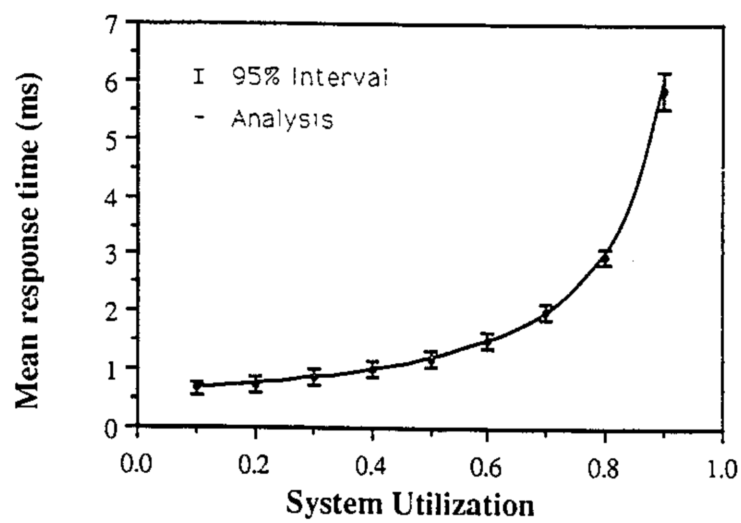


Figure 5.3: Mean message response time for exhaustive service with $\overline{g} = 5$.

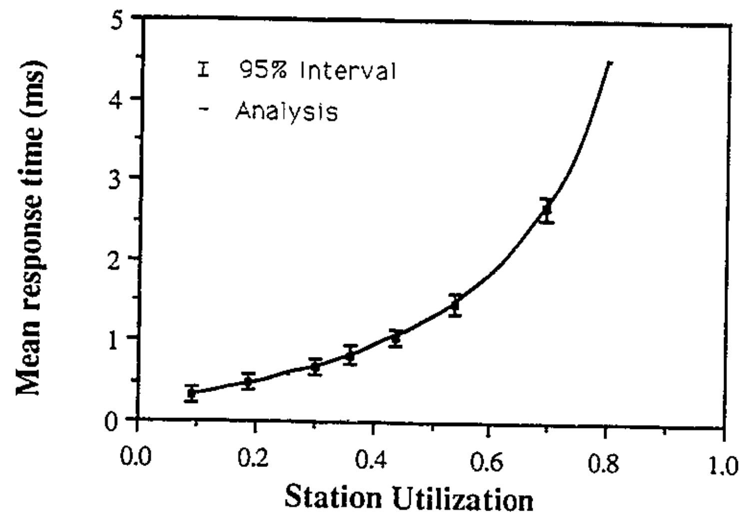


Figure 5.4: Mean packet response time for single service with b = 0.1.

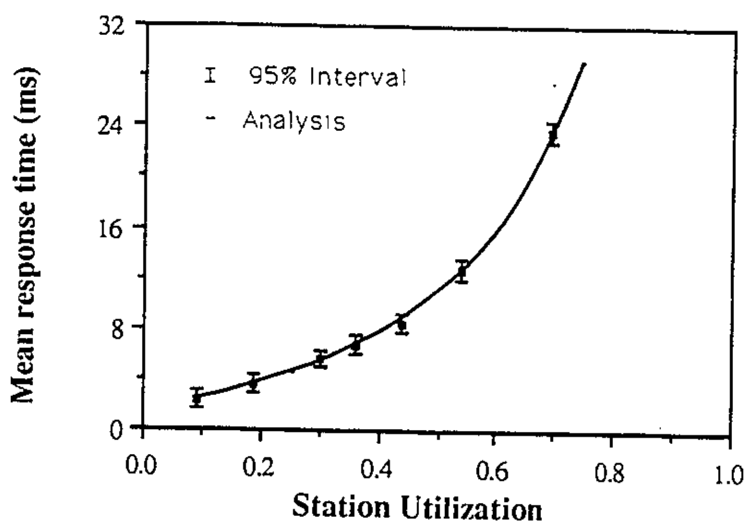


Figure 5.5: Mean message response time for single service with $\overline{g} = 5$.

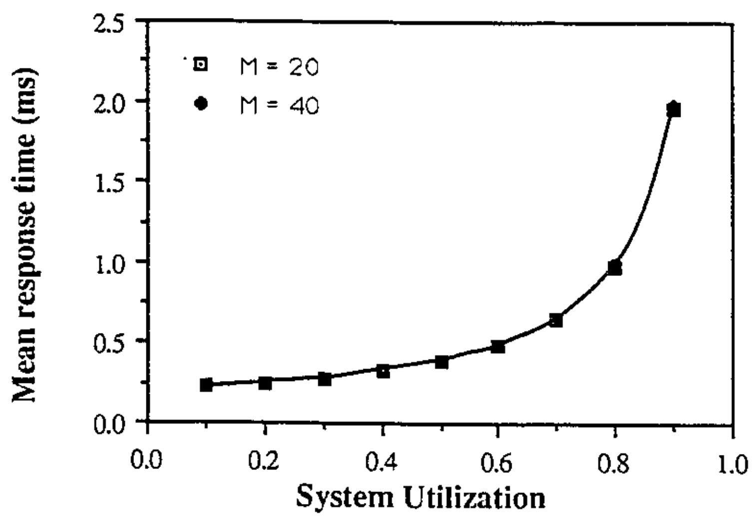


Figure 5.6: The effect of M for exhaustive service with $b = 0.1$.

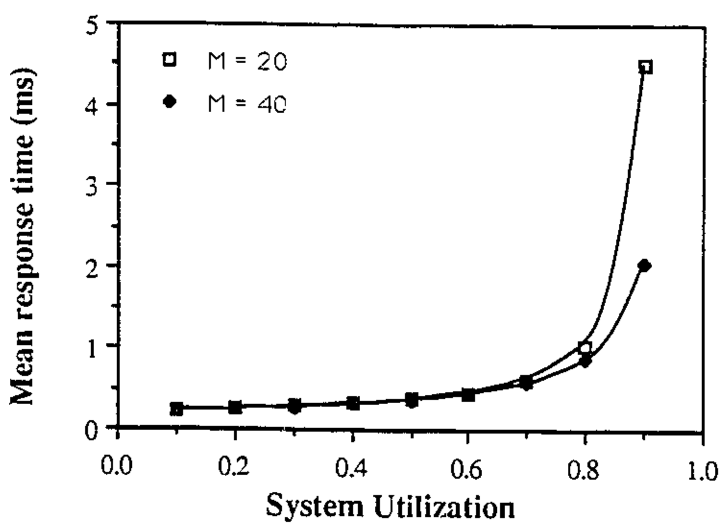


Figure 5.7: The effect of M for single service with $b = 0.1$.

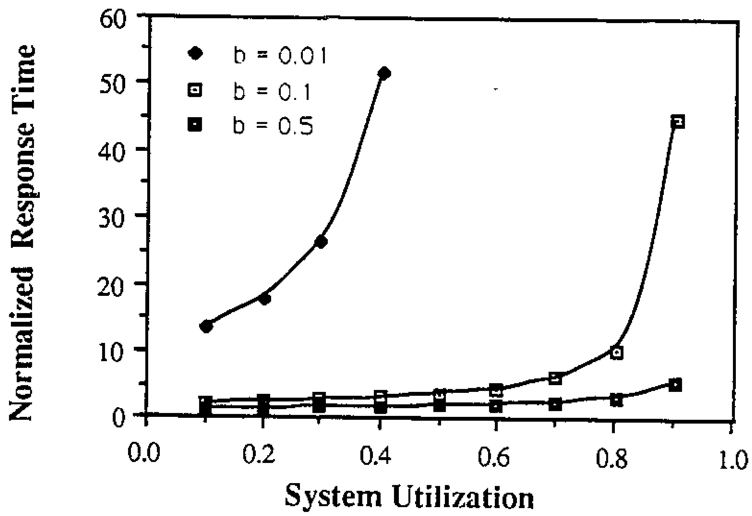


Figure 5.8: The effect of packet size for single service with M = 20.

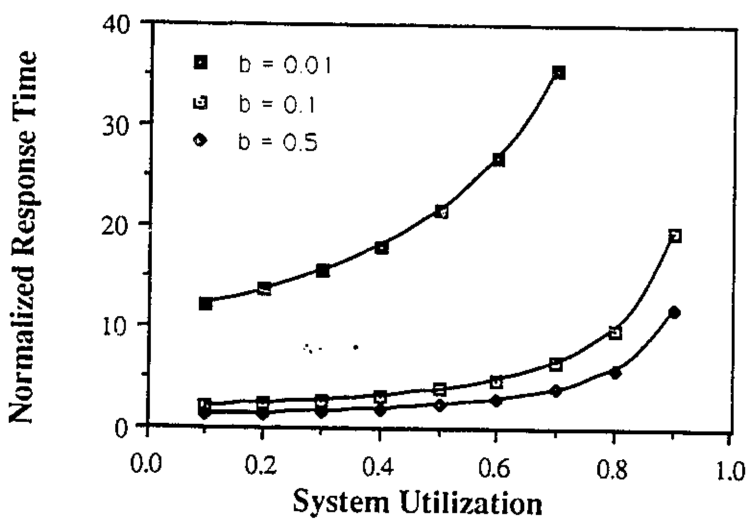


Figure 5.9: The effect of packet size for exhaustive service with M = 20.

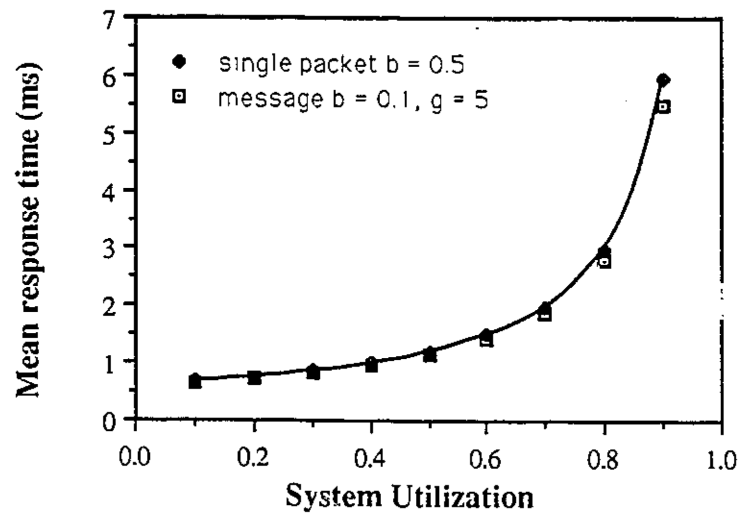


Figure 5.10: The effect of packet size on message delay for exhaustive service.

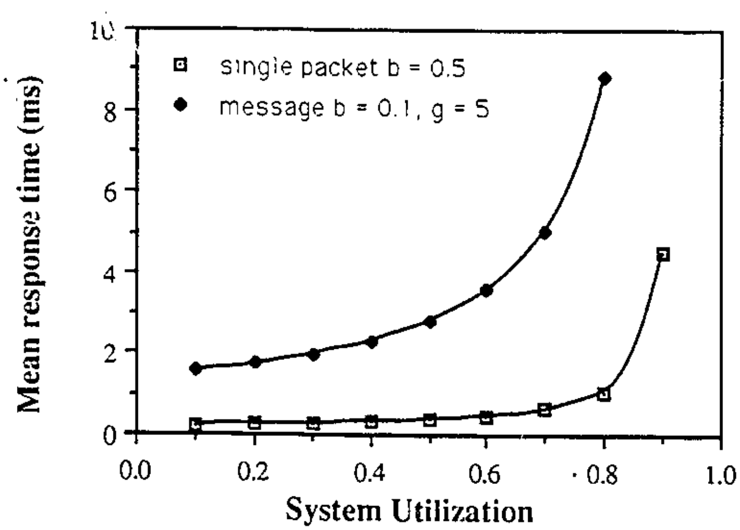


Figure 5.11: The effect of packet size on message delay for single service.

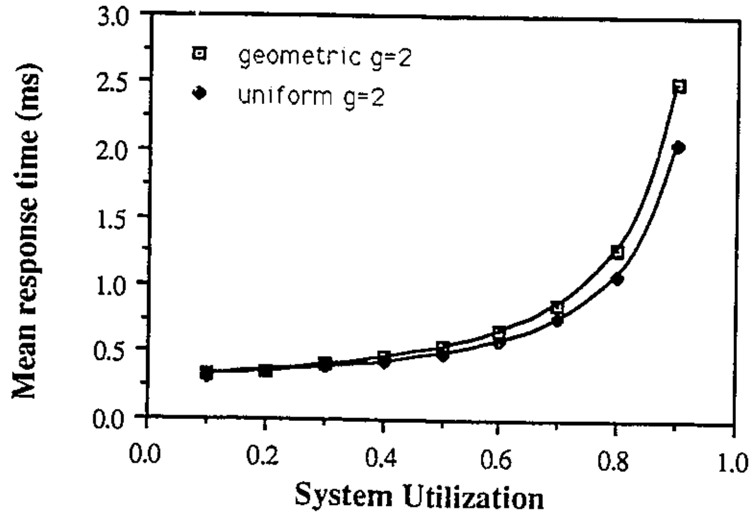


Figure 5.12: The effect of message length distribution on delay for exhaustive service

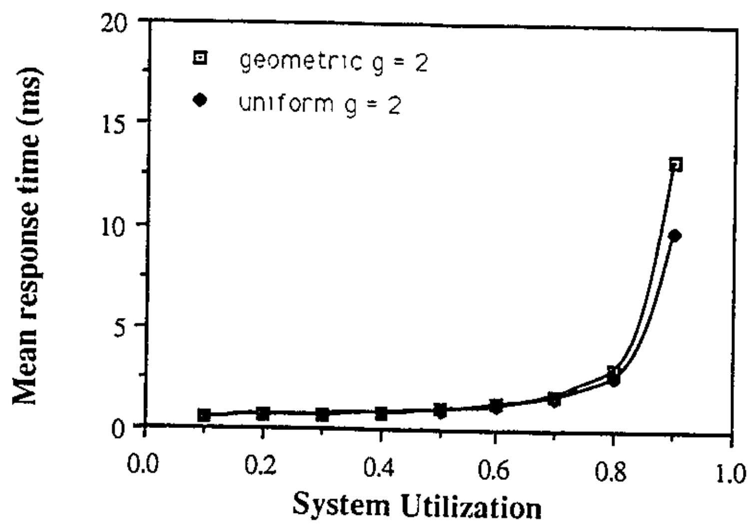


Figure 5.13: The effect of message length distribution on delay for single service

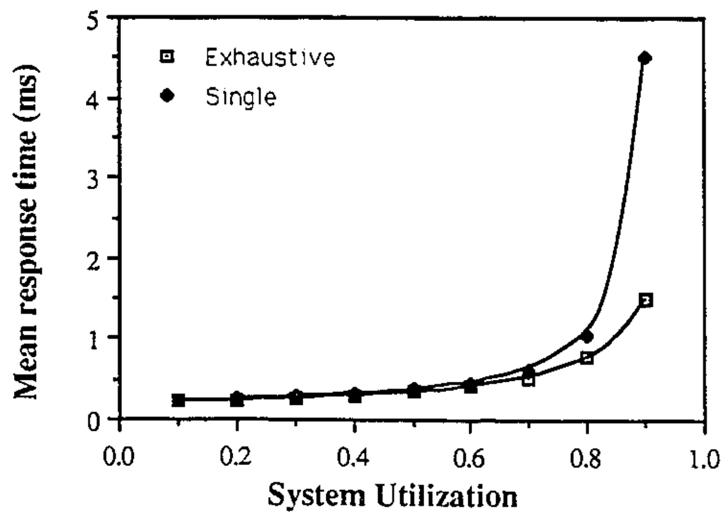


Figure 5.14: Comparison of two policies for packet delay with $b = 0.1$.

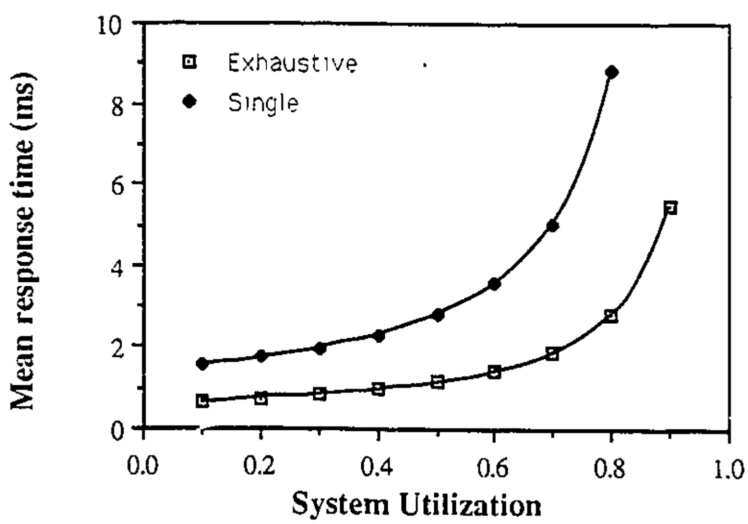


Figure 5.15: Comparison of two policies for message delay with $b = 0.1$ and $\overline{g} = 5$.

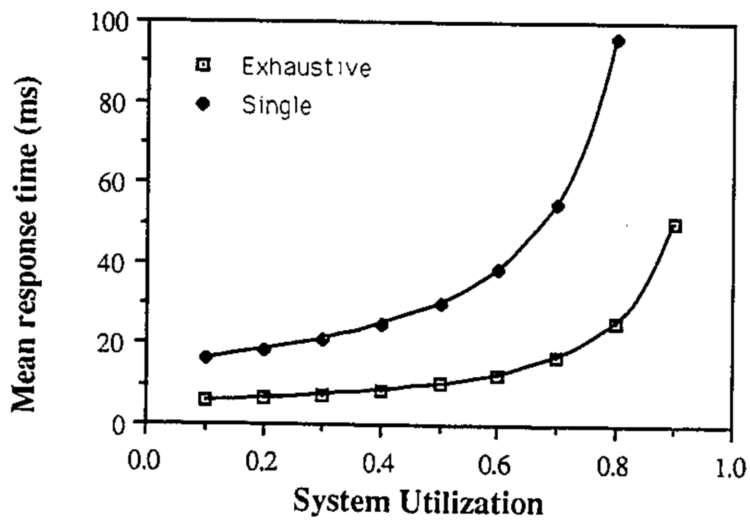


Figure 5.16: Comparison of two policies for message delay with $b = 0.1$ and $\bar{g} = 50$.

Chapter 6

An Analysis of Slotted ALOHA Systems

The slotted ALOHA scheme is characterized by using a distributed random access algorithm to control the transmission of M stations in a single channel environment. Thus the activity of one station in the system will affect the behavior of queues at other stations, and this gives rise to a statistical dependence among queues in the system. In this chapter we study the performance of slotted ALOHA systems by focussing on the queueing behavior of one station and model it as an M/G/1 queue in which the first packet of each busy period receives an exceptional service. The interactions among all the stations are taken into account through the service time distributions. Simulations are run to verify the analytical results.

6.1 Introduction

The classic ALOHA access scheme [2] is a method which can be employed in shared-channel networks (i.e., in packet radio networks or in satellite networks, also reported recently in fiber optical based LANs [15]). In this access scheme a station transmits a packet as soon as the packet arrives, and packets transmitted concurrently will collide and cannot be received correctly at the receiving stations. Collided packets are retransmitted after a randomly selected period of time, until they are successfully received.

The slotted ALOHA scheme [37] is similar to the ALOHA scheme, but in addition, the time axis is divided into equal length slots (each of them equals in length the transmission time of a packet) and stations are allowed to start transmission only at the beginning of every time slot as shown in figure 6.1. Usually, a geometrically distributed delay process is employed in the retransmission of collided packets.

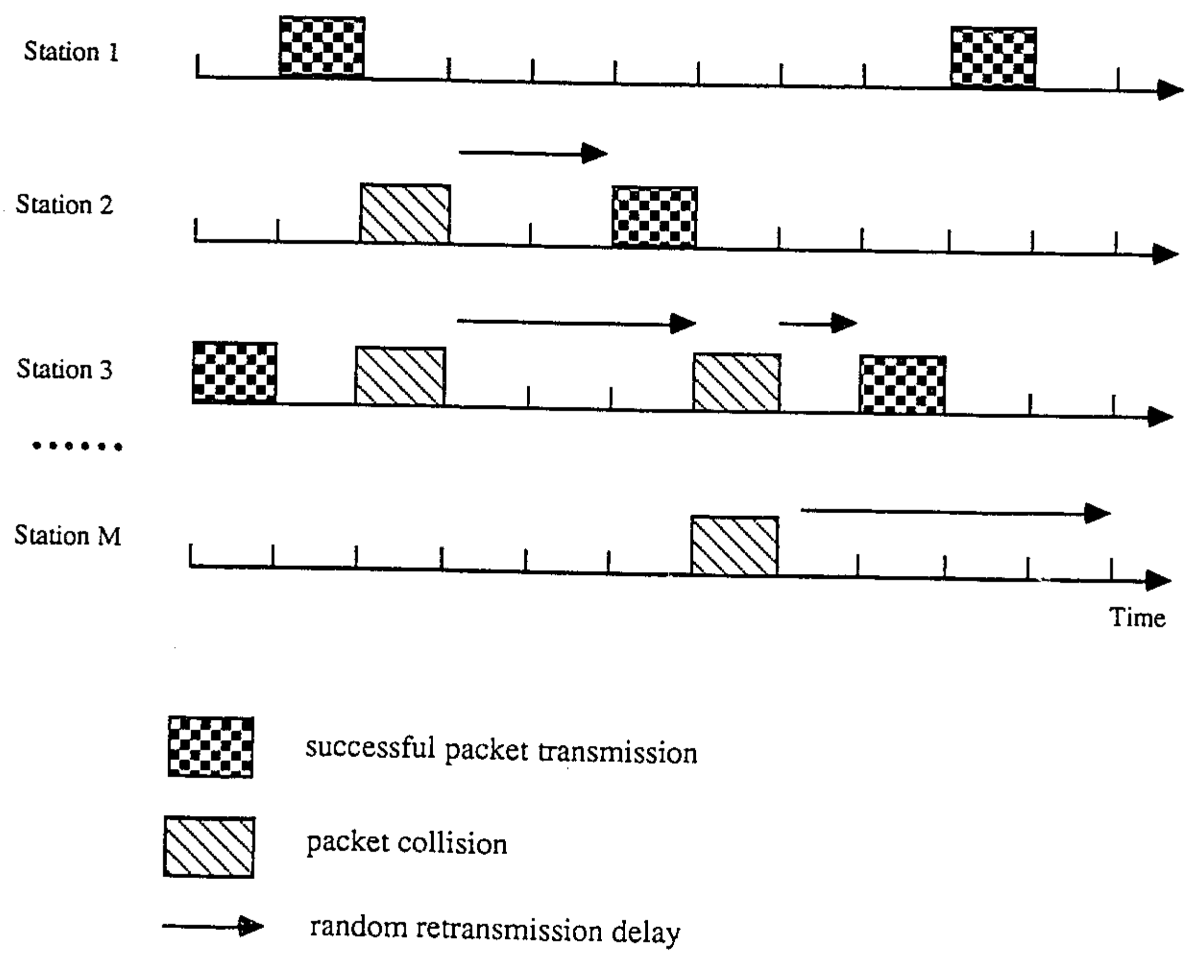


Figure 6.1: Slotted ALOHA random access.

The ALOHA scheme has been extensively studied in the past. However, most of them either concentrate on capacity analysis or assume an infinite user population that implies no queueing of packets at the terminals. Naturally, the main result derived by these studies is the throughput of the system, rather than the station, as a function of the offered load [2], [26] and [37]. Several studies (see [24], for example) analyzed the delay in the slotted ALOHA system. However, these studies do not model any queueing behavior, so the expressions derived in these reports mainly represent the retransmission delay for a packet and not any queueing delay.

For the analysis of a system of a finite number of stations, each with an infinite buffer, the first attempt was made in [21]. That paper takes for granted an infinite buffer capacity and ignores any dependence of the throughput of a station upon the number of stations attempting access. In [11], an approximation for a slotted ALOHA system of a finite number of stations with Bernoulli arrival processes was analyzed by coupled Markov chains, but the analytical approach was very complicated.

For very simple systems, the exact queueing analysis of the slotted ALOHA system has been done in [42]. In that report, based on the symmetric property, the mean response time in a two-station slotted ALOHA system with Bernoulli arrival processes was obtain. Recently, a variation of this model was analyzed by the same approach in [4]. Unfortunately, this method can not easily be extended to an M-station system.

In contrast to the previous studies, this chapter attempts to derive an expression for the queue length distribution as well as for the LST of the pdf of the response time for a slotted ALOHA system of M symmetric stations each with an infinite buffer. Additional contributions come from the modelling of the system as a simple M/G/1 queue in which the first packet of each busy period receives an exceptional service and the derivation of the expression of the LST of the pdf of the service time.

The analysis done in this chapter is divided into two main parts. First, in section 6.3 we analyze the queue length distribution of a tagged station in the system. The analysis is based on the results derived in chapter 2 for the M/G/1 queue in which the first packet of each busy period receives an exceptional service. In the second part of this chapter we present an approach for analyzing the packet service time distributions at each station. This approach analyzes the system by constructing a Markov chain which is formed by the steady state probability of having i busy stations (having at least one packet in the queue) in the system. By employing the steady state analysis and constructing a state transition matrix, we derive the service time distributions. This is done in section 6.4. The results derived are discussed in section 6.5 through some numerical examples.

6.2 Model and Assumptions

The system considered consists of M symmetric stations which transmit packets to each other over a single channel. In this channel, we assume that if the transmissions of two (or more) stations overlap all transmitted packets get garbled, and we call this a collision. Therefore, a station can transmit a packet successfully if and only if no other stations attempt to use the channel at the same time.

The system works synchronously, in that the time axis is divided into slots with the slot size equal to the transmission time of a fixed length packet, which, without loss of generality, is assumed to be unity. Each station is considered to have an infinite buffer. The arrival process to each station is an identical independent Poisson process with an arrival rate of λ packet/ s. Packets which arrive at a given station are served according to a FCFS order.

The medium access scheme studied in this chapter is assumed to be slotted ALOHA. More specifically, at the beginning of a slot a packet, at head of queue, is transmitted with probability p if it has arrived at a non-empty queue; otherwise it is transmitted with probability 1. A station involved in a collision will continue to retransmit the packet. To avoid a deadlock, this station will either transmit in any of the succeeding slots with probability p, or stay quiet with probability $1 - p$ (geometric retransmission delay process). This collision resolution scheme is used until a packet is transmitted successfully. We assume that the propagation delay is zero, so that at the end of the slot, all stations know the status of the channel in that slot. In addition, it is assumed that the channel is error free.

The assumption that the retransmission process has a memoryless geometric distribution permits a simple description for the analysis. However, the assumption that the propagation delay is zero is obviously not true for satellite channels. For a satellite channel the transmission propagation R cannot be neglected. However, simulation results have shown that the delay performance of slotted ALOHA systems are dependent primarily upon the average retransmission delay and quite insensitive to the exact probability distributions [24]. For a slotted ALOHA system with a uniform retransmission process with a parameter of K and a propagation delay R, the average retransmission delay is $R + (K + 1)/2$. Since the average retransmission delay from a geometric distribution (with probability p) is $1/p$, one can find a proper p to match the retransmission delay. Therefore, to use the analytical results of the Markovian model here to predict the delay performance of a slotted ALOHA system with a nonzero propagation delay, we may choose

$$\frac{1}{p} = R + (K+1)/2. \tag{6.1}$$

We shall model this slotted ALOHA system by a system of M interferring queues served by a common server shown in figure 6.2. By assuming that the behavior of one station can be analyzed by taking into account the influence of other stations in the system through the service time distributions, each station in the system can be viewed as an M/G/1 queue in which the first packet of each busy period receives an exceptional service. As a result, the system is decomposed into M individual queues. It is obvious that this model is an approximation of the real situation. However, it allows us to evaluate the performance of the whole system through the analysis of one of the M stations in the system. The reason to choose the model of the M/G/1 queue in which the first packet of each busy period receives an exceptional service instead of the M/G/1 queue or the M/G/1 vacation model is that the packet initiating a busy period is treated differently from others in this protocol. As stated in the protocol, the packet that initiates a busy period will be transmitted with probability 1 at the first attempt. After the successful transmission of the first packet, the next packet in the queue, if there is one, enters the transmission process immediately and will be transmitted with probability p. Therefore, two service time random variables are needed and the M/G/1 queue in which the first packet of each busy period receives an exceptional service is chosen to model this protocol.

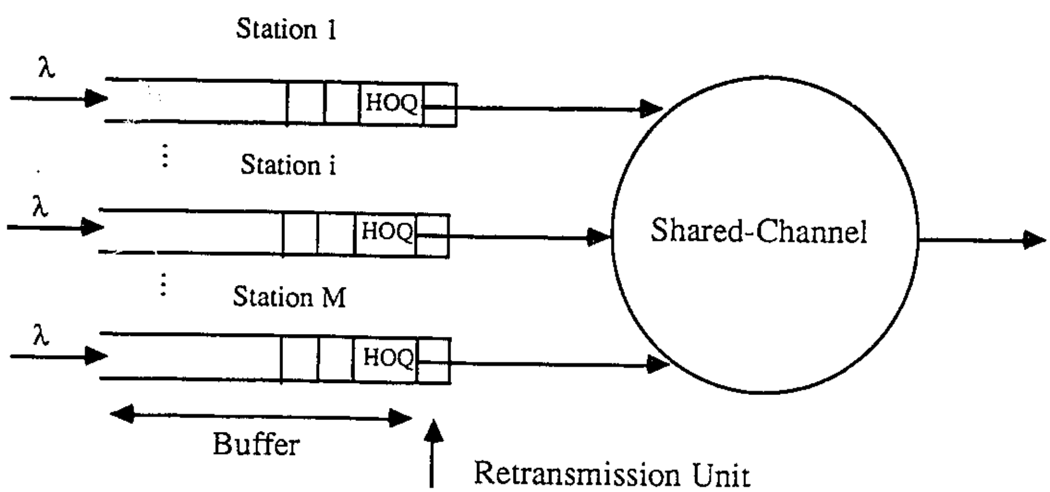


Figure 6.2: Queueing model of a slotted ALOHA system.

6.3 Number of Packets in the Queue

Let $\tilde{L}(t)$ be the number of packets in the station (both in the queue and in service) at time t. The steady state PGF of the queue length distribution can be found by using the M/G/1 queue in which the first packet of each busy period receives an exceptional service. Recall that a station in the slotted ALOHA system has a Poisson arrival. The first packet of each busy period will get service given by $B_1^*(s)$, the LST of the pdf of the service time. The server will provide service to other packets in the queue according to a LST of the pdf of the service time $B^*(s)$. Let

$$q_n = \lim_{t \to \infty} Prob[\tilde{L}(t) = n], \tag{6.2}$$

and define the PGF

$$Q(z) = \sum_{n=0}^{\infty} q_n z^n. \tag{6.3}$$

From the results obtained in chapter 2, we have the queue length distribution to be

$$Q(z) = \sum_{n=0}^{\infty} q_n z^n = q_0 \frac{z B_1^*(\lambda - \lambda z) - B^*(\lambda - \lambda z)}{z - B^*(\lambda - \lambda z)}, \tag{6.4}$$

where q_0 is given by

$$q_0 = \frac{1 - \lambda \bar{b}}{1 + \lambda(\bar{b_1} - \bar{b})}, \tag{6.5}$$

and $B^*(s)$ and $B_1^*(s)$ are to be obtained in section 6.4.2 after the following analysis.

6.4 Service Time Distribution Analysis

An important step in the analysis of this chapter is the derivation of the service time distributions. Define the service time to be a random variable which represents the time duration from the instant a packet arrives at the head of the queue until it is successfully transmitted. In steady state, the distribution function of this random variable is generally dependent on the steady state PGF denoted as $F(\mathbf{S})$, where $\mathbf{S}$ is the state vector $(S_1, S_2, \cdots, S_M)$ and S_i is the number of packets in the ith station's buffer. However, due to the operation of the slotted ALOHA protocol, the service time distribution is independent of the exact queue length distributions of the busy stations, and only depends on k, the number of busy stations (having at least one packet in the queue). This characteristic is due to the fact that during its operation the slotted ALOHA protocol only needs to know the number of stations having a packet to transmit, and not the actual number of packets in the queue for each busy station. This observation simplifies the analysis if a further assumption is imposed. This assumption is that the steady state probability π_{i0}, that the departure of a packet from the ith station leaves the ith station's buffer empty, is also independent of the exact queue length distributions of the other stations. Since all stations receive packets at the same rate λ, and the slotted ALOHA protocol is fair to every station, it is assumed that the probabilities π_{i0} are all equal to π_0.

6.4.1 State Probability

Let $\tilde{N}(t)$ be a random variable representing the number of busy stations in the system at time t. In steady state, the probability of having i busy stations in the system at time t slots is defined as

$$P_i = \lim_{t\to\infty} Prob[\tilde{N}(t) = i] \qquad i = 0, 1, 2, \cdots \tag{6.6}$$

In this case, $\tilde{N}(t)$ forms a Markov chain on the slot boundary and gives a state description for the system. The state space is discrete and consists of a set of integers $\{ 0, 1, 2, \cdots, M \}$. Define the one step stationary state transition probabilities of $\tilde{N}(t)$ to be

$$p_{ij} = Prob[\tilde{N}(t+1) = j \mid \tilde{N}(t) = i] \qquad i, j = 0, 1, 2, \cdots, M. \tag{6.7}$$

Let σ be the probability of having at least one packet arriving in a slot at one station, then under a Poisson arrival assumption

$$\sigma = 1 - e^{-\lambda}. \tag{6.8}$$

Recall that p is the probability for resolving the collision. That is a station will transmit a queued packet at the head of the queue with probability p in the next slot. The same p is also used for the geometric retransmission process in the next time slot.

Since a packet arriving at an empty queue will be transmitted with probability 1 in the coming slot, the probability of having packets arrival in a slot for an empty station is the probability of having a packet to transmit at the next slot. Also as discussed before, packets arrived at a non-empty queue do not affect the number of busy stations in the system. Based on these assumptions a state transition matrix can be derived as follows.

Since there can be no more than one successful transmission in a slot, a transition from state i to state $j < i - 1$ is impossible, and $p_{ij} = 0$ for $j < i - 1$. Next consider the transition from state i to state $i-1$. A reduction in the number of busy stations can only come about if there are no new packets arriving at empty stations in the system, and only one of the i busy stations transmits a packet such that the departure of that packet leaves an empty queue.

The transition from state i to the very same state can happen due to three distinct reasons. The first results from the circumstance in which no new packets arrived at empty stations while none or two or more of the i stations transmitted. The transmitting stations clearly collided and the number of busy stations remains the same. This has a probability of $(1-ip\,(1-p)^{i-1})\,(1-\sigma)^{M-i}$. The second reason for this transition results from a situation in which no new packets arrived at empty stations while one out of the i busy stations transmitted successfully, but the departure of the packet does not leave an empty queue. This contributes a probability of $ip\,(1-\pi_0)\,(1-p)^{i-1}\,(1-\sigma)^{M-i}$. The third reason for this transition results from the case in which one empty station got one packet arrival (and transmitted) and none of the i busy stations transmitted. In this case the empty station got a packet and transmitted successfully and therefore the number of busy stations remains the same. It has a probability of $(1-p)^i\,(M-i)\lambda\, e^{-\lambda}\,(1-\sigma)^{M-i-1}$.

The next transition to consider is from state i to state $j = i + 1$. Since the number of busy stations increased, one of the empty stations must have packets arrival. The transition results from two reasons. The first reason is that the empty station had more than one packets arrival. So after the transmission, no matter whether the transmission is successful or not, the station is non-empty and the number of busy stations increased by one. This event contributes a probability of $(M-i)\,(1-\sigma)^{M-i-1}\,(1-e^{-\lambda}-\lambda e^{-\lambda})$. The second reason is that the empty station had one packet arrival and at least one of the i busy stations transmitted, so a collision happened, the station remains non-empty after transmission and the number of busy stations increased by one. This event has a probability of $(M-i)\ \lambda e^{-\lambda}\ (1-\sigma)^{M-i-1}\ (1-(1-p)^i)$.

The last case is the transition from state i to state $j \geq i+2$. Here the number of busy stations increased by two or more meaning that there are $j-i$ stations out of $M-i$ empty stations had arrival and transmitted. Collision is generated. The number of busy stations increased by $j-i$. The activity of the busy stations is immaterial in this case.

Summarizing all the events discussed above, the general transitional probability p_{ij} can be expressed as

$$p_{ij} = \begin{cases} 0 & j \leq i-2 \\ ip\ \pi_0\ (1-p)^{i-1}\ (1-\sigma)^{M-i} & j = i-1 \\ ip\ (1-\pi_0)\ (1-p)^{i-1}\ (1-\sigma)^{M-i} & \\ \quad +\ (1-ip\ (1-p)^{i-1})\ (1-\sigma)^{M-i} & \\ \quad +\ (1-p)^i\ (M-i)\lambda\ e^{-\lambda}\ (1-\sigma)^{M-i-1} & j = i \\ (M-i)\ \lambda e^{-\lambda}\ (1-\sigma)^{M-i-1}\ (1-(1-p)^i) & \\ \quad +\ (M-i)\ (1-\sigma)^{M-i-1}\ (1-e^{-\lambda}-\lambda e^{-\lambda}) & j = i+1 \\ \begin{pmatrix} M-i \\ j-i \end{pmatrix} \sigma^{j-i}(1-\sigma)^{M-j} & j \geq i+2 \end{cases} \tag{6.9}$$

The stationary probability distribution $\{P_i\}_{i=0}^{M}$ of $N(t)$ can be computed by solving the following set of linear simultaneous equations in terms of π_0:

$$P_j = \sum_{i=0}^{M} P_i\ p_{ij}, \tag{6.10}$$

for $j = 0, 1, 2, \cdots, M$. Of course, the steady state probabilities must also satisfy for the normalizing condition

$$\sum_{i=0}^{M} P_i = 1. \tag{6.11}$$

Note that one of the equations in (6.10) is linearly dependent on others with the constraint of equation (6.11).

6.4.2 Service Time Distributions

There are two service time distributions, $B_1(t)$ and $B(t)$. $B_1(t)$ is the service time distribution for packets that initiate a busy period for the tagged station, and $B(t)$ is the service time distribution for the remaining packets in the queue of the station. This distinction is necessary, because a packet arriving at an empty queue has to wait for transmission at the beginning of the next slot. Also a packet that initiates a busy period will be transmitted with probability 1 at the first attempt. However, after the successful transmission of the first packet, the next packet in the queue, if there is one, enters the transmission process immediately and will be transmitted with probability p. Since all the stations in the system are symmetric and the slotted ALOHA protocol is fair to all the stations, the service time distributions can be considered the same for all the stations.

Let $\tilde{b}_1$ be a random variable with a probability distribution function $B_1(t)$. From the previous discussion, we know $\tilde{b}_1$ is the sum of two independent random variables $\tilde{x}$ and $\tilde{y}$, that is

$$\tilde{b}_1 = \tilde{x} + \tilde{y}, \tag{6.12}$$

where $\tilde{y}$ is the residual life time required for the synchronization of the transmission. It is distributed between 0 and 1, with a LST of

$$Y^*(s) = \frac{1 - e^{-s}}{s}. \tag{6.13}$$

The random variable $\tilde{x}$ represents the time duration from the instant the first attempt is made to transmit the packet until the time the packet is successfully transmitted. The random variable $\tilde{x}$ takes an integral value of time slots.

Recall that the retransmission process of each collided packet is assumed to be geometrically distributed with probability p. Let Ω_1 be the probability that a packet which initiates a busy period is transmitted successfully in the first attempt by a given station, and Ω be the probability that a packet is retransmitted successfully by the station. To obtain Ω_1, we have to condition and uncondition on having i busy stations in the system. The probability that a packet which initiates a busy period is transmitted successfully in the first attempt by a given station conditioning on having i busy stations in the system is $(1-p)^i\ (e^{-\lambda})^{M-i-1}$, where $(e^{-\lambda})^{M-i-1}$ is the probability that there is no arrival in the remaining $M-i-1$ idle stations, and $(1-p)^i$ is the probability that there is no transmission among the i busy stations. Since the number of busy stations can be from 0 to M - 1, we have Ω_1 by unconditioning on having i busy stations in the system.

$$\Omega_1 = \sum_{i=0}^{M-1} (1-p)^i (e^{-\lambda})^{M-i-1} P_i, \tag{6.14}$$

where P_i is the probability of having i busy stations and can be obtained from the state transition matrix derived before. By the same approach, we have

$$\Omega = p \sum_{i=1}^{M} (1-p)^{i-1} (e^{-\lambda})^{M-i} P_i, \tag{6.15}$$

Based on that, we have the distribution function of the random variable $\tilde{x}$ to be

$$x(t) = Prob[t=1] + Prob[t=2] + \cdots + Prob[t=k] + \cdots \tag{6.16}$$

where

$$Prob[t=1] = \Omega_1, \tag{6.17}$$

$$Prob[t=k] = (1-\Omega_1)\Omega(1-\Omega)^{k-2} \qquad k > 1. \tag{6.18}$$

The LST of the pdf of $\tilde{x}$ can be obtained by

$$X^*(s) = \int_0^\infty e^{-st} x(t) dt. \tag{6.19}$$

$$X^*(s) = \Omega_1 e^{-s} + (1-\Omega_1)e^{-s}\Omega e^{-s} + (1-\Omega_1)e^{-s}\Omega(1-\Omega)e^{-2s} + \cdots \tag{6.20}$$

By simplifying that, we have

$$X^*(s) = \Omega_1 e^{-s} + \frac{(1-\Omega_1)\Omega e^{-2s}}{1-(1-\Omega)e^{-s}}. \tag{6.21}$$

Since the service time random variable $\tilde{b}_1$ is the sum of two independent variables $\tilde{x}$ and $\tilde{y}$, the LST of the pdf of $\tilde{b}_1$ can be obtained by

$$B_1^*(s) = Y^*(s)X^*(s) = \frac{1-e^{-s}}{s}\left(\Omega_1 e^{-s} + \frac{(1-\Omega_1)\Omega e^{-2s}}{1-(1-\Omega)e^{-s}}\right). \tag{6.22}$$

The moments of the service time $\tilde{b}_1$ can easily be obtained as

$$\overline{b_1} = \frac{1}{2} + 1 + \frac{1-\Omega_1}{\Omega}, \tag{6.23}$$

$$\overline{b_1^2} = \frac{1}{3} + (1 + \frac{1-\Omega_1}{\Omega}) + \Omega_1 + (1-\Omega_1)\frac{\Omega^2+\Omega+2}{\Omega^2}. \tag{6.24}$$

The service time distribution $B(t)$ can be analyzed by the same approach. Recall that a queued packet is transmitted with probability p for the first attempt as well as for the retransmission. So the probability that a station transmits a queued packet successfully is Ω for the first attempt and for the retransmission. Ω has been given in equation (6.15). The distribution function of $B(t)$ can be obtained from

$$b(t) = Prob[t=1] + Prob[t=2] + \cdots + Prob[t=k] + \cdots, \tag{6.25}$$

where

$$Prob[t=1] = \Omega, \tag{6.26}$$

$$Prob[t=k] = \Omega(1-\Omega)^{k-1} \quad k > 1. \tag{6.27}$$

The LST of the service time distribution $B(t)$ can be obtained by

$$B^*(s) = \int_0^\infty e^{-st} b(t)dt. \tag{6.28}$$

$$B^*(s) = \Omega e^{-s} + (1-\Omega)\Omega e^{-2s} + \Omega(1-\Omega)^2 e^{-3s} + \cdots. \tag{6.29}$$

By simplifying that, we have

$$B^*(s) = \frac{\Omega e^{-s}}{1-(1-\Omega)e^{-s}}. \tag{6.30}$$

The moments can be obtained easily

$$\overline{b} = \frac{1}{\Omega}, \tag{6.31}$$

$$\overline{b^2} = \frac{2-\Omega}{\Omega^2}. \tag{6.32}$$

The mean service time of a packet can be obtained by

$$\overline{b_s} = (1-q_0)\overline{b} + q_0\overline{b_1} = \frac{\overline{b_1}}{1-\lambda(\overline{b}-\overline{b_1})}. \tag{6.33}$$

Note that up till now, the previously defined probability π_0 still remains to be evaluated. Under the Poisson arrival assumption the probability that a queue has an empty buffer is equal to the probability that a packet departure leaves an empty buffer [23, chpt 5]. So we have

$$q_0 = \pi_0. \tag{6.34}$$

This gives another independent equation for π_0 via

$$q_0 = \pi_0 = \frac{1 - \lambda\bar{b}}{1 - \lambda(\bar{b} - \overline{b_1})}. \tag{6.35}$$

The analysis of the service time is now complete from which the performance of the system can be evaluated.

6.5 Delay Performance

Since each station in the system can be analyzed independently by employing the M/G/1 queue in which the first packet of each busy period receives an exceptional service, the LST of the pdf of the response time distribution can be obtained by using the results in chapter 2. Namely,

$$T^*(s) = \frac{1 - \lambda\bar{b}}{1 + \lambda(\overline{b_1} - \bar{b})} \frac{\lambda B^*(s) - (\lambda - s)B_1^*(s)}{s - \lambda + \lambda B^*(s)}. \tag{6.36}$$

Its mean is given by

$$E[T] = \frac{\overline{b_1}}{1 + \lambda(\overline{b_1} - \bar{b})} + \frac{\lambda\overline{b^2}}{2(1 - \lambda\bar{b})} + \frac{\lambda(\overline{b_1^2} - \overline{b^2})}{2(1 + \lambda\overline{b_1} - \lambda\bar{b})}. \tag{6.37}$$

The average queue length can easily be obtained from Little's rule

$$E[L] = \lambda E[T]. \tag{6.38}$$

The steady state throughput S of the system is defined as the average number of successfully transmitted packets per slot. Since the system is in equilibrium, no packet is rejected and all the received packets will be eventually transmitted successfully. In view of this the input rate equals the output rate per slot, and

$$S = M\lambda. \tag{6.39}$$

6.6 Numerical Examples

In order to evaluate the performance measures, the service time distributions and the steady state probabilities of the number of busy stations in the system must be computed. The following simple iteration scheme may be used in order to obtain the service time distribution and the steady state probabilities P_i.

1. Assign an arbitrary initial value to the probability π_0, $0 < \pi_0 < 1$.

2. Solve the set of linear equations (6.9) with respect to the steady state probability π_0.

3. Compute a new estimate for the value of the probability π_0 from equation (6.5).

4. Repeat steps 2 and 3 until a convergence criterion is satisfied.

Thus the steady state probabilities P_i can be computed by the solution of series of linear equations. Based on that the performance measures, namely, the mean service time and the mean response time versus system throughput (S) can be computed.

Results have been obtained for a population of $M = 5$, 10 and 20 stations. The arrival process to each station is an identical independent Poisson process. The transmission time of the fixed length packet is equal to the slot time, which is normalized to one. A retransmission process with a geometric distribution is chosen to resolve the collision.

A simulation program using the QNAP2 simulation software package has been developed and the program is attached in appendix C. In general the simulation time is proportional to the value of M and the value of ρ. For certain cases, the simulation has to be run for days to get a convergent result. In this study, simulations have been run for systems with parameters described below in order to verify the analytical results. For each set of system parameters, proper simulation time values are selected to obtain convergent results. The 95% confidence intervals have also been obtained.

Figures 6.3, 6.5 and 6.7 depict the mean service time performance versus system throughput for the systems of $(M, p) = (5, 0.2)$, $(10, 0.1)$ and $(20, 0.05)$ respectively. Figures 6.4, 6.6 and 6.8 show the mean response time versus system throughput for the corresponding systems with the same parameters. All time measures are normalized with respect to the slot time. From the figures we can find that all analytical results agree with simulation results well in the range of low system throughputs. As the system throughput gets high, for example, the system throughput (S) is greater than 0.3, we found that they cannot match at all. From the simulation, when the system throughput goes beyond 0.3, the 95% confidence intervals of QNAP2 simulation results become very large and simulation results do not converge. This can probably be suggested by the fact that the maximum system throughput in a slotted ALOHA system is 0.368 [47, sct 3.2]. Together with the use of an infinite buffer, our system may become unstable.

6.7 Summary

In this chapter an approximate queueing model for an M-station slotted ALOHA system with infinite buffer capacities was studied. This is achieved by examining the queueing behaviors of one queue and incorporating the interference from other queues in the packet service time. By employing the M/G/1 queue in which the first packet of each busy period receives an exceptional service model, we derived the service time distributions and the response time distribution for a queue in the system under steady state. The analytical results are compared against simulation results and proved to be very good in the regions of stability.

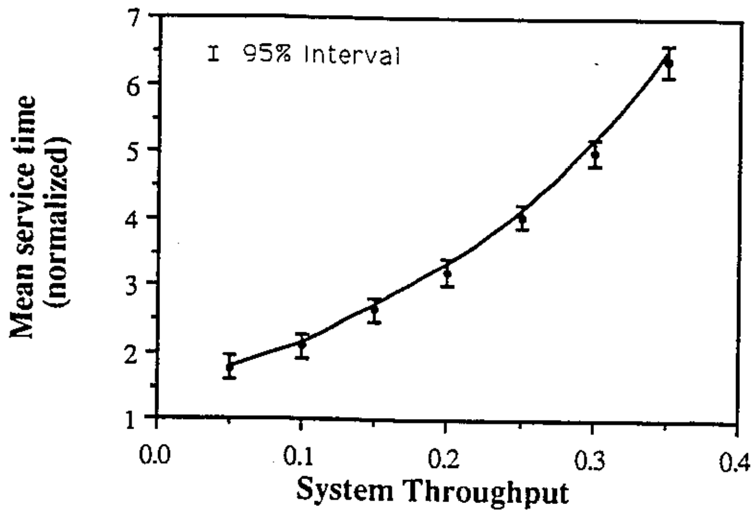


Figure 6.3: Mean service time versus throughput for M = 5, p = 0.2.

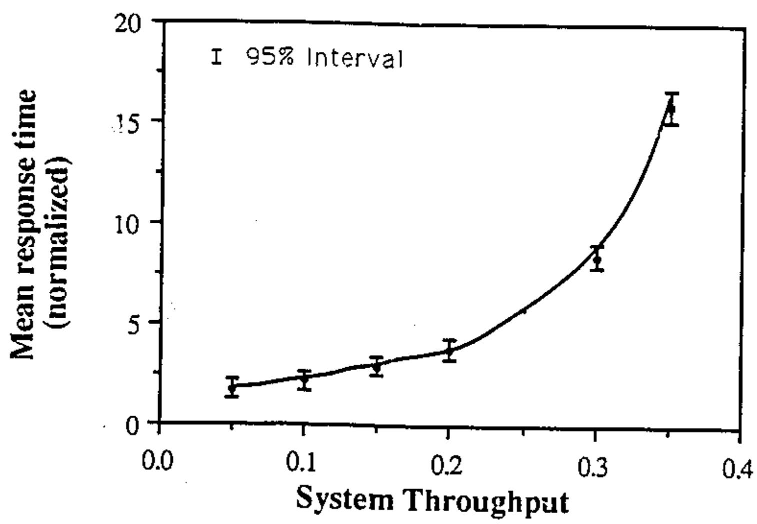


Figure 6.4: Mean response time versus throughput for M = 5, p = 0.2.

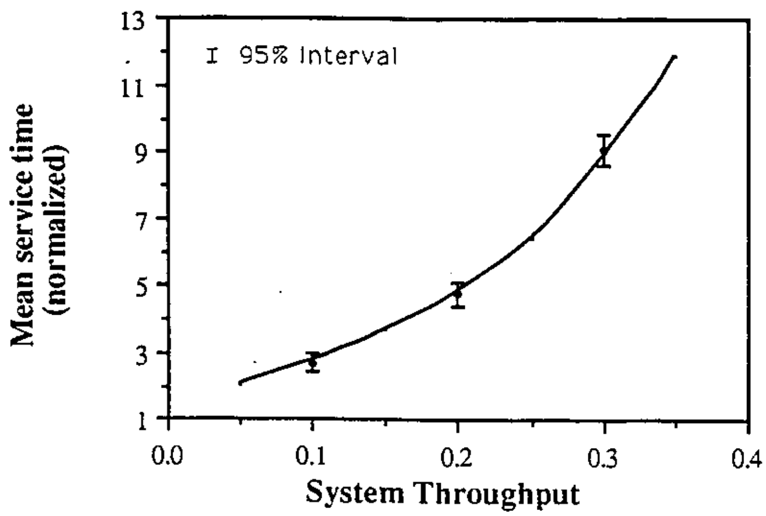


Figure 6.5: Mean service time versus throughput for M = 10, p = 0.1.

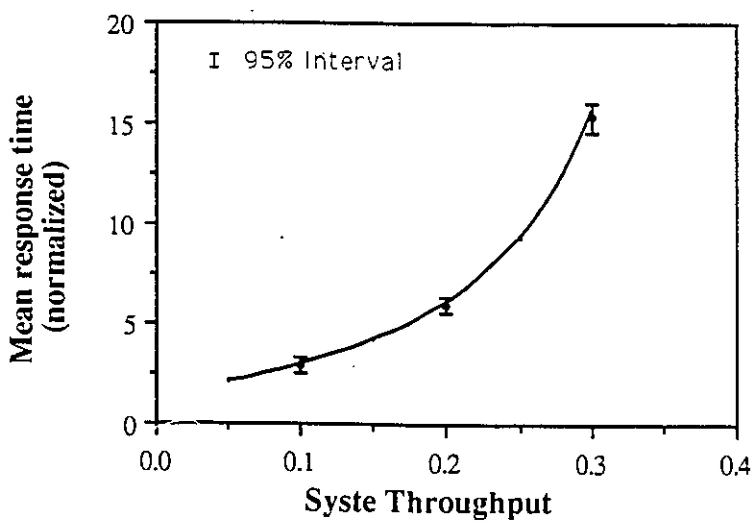


Figure 6.6: Mean response time versus throughput for M = 10, p = 0.1.

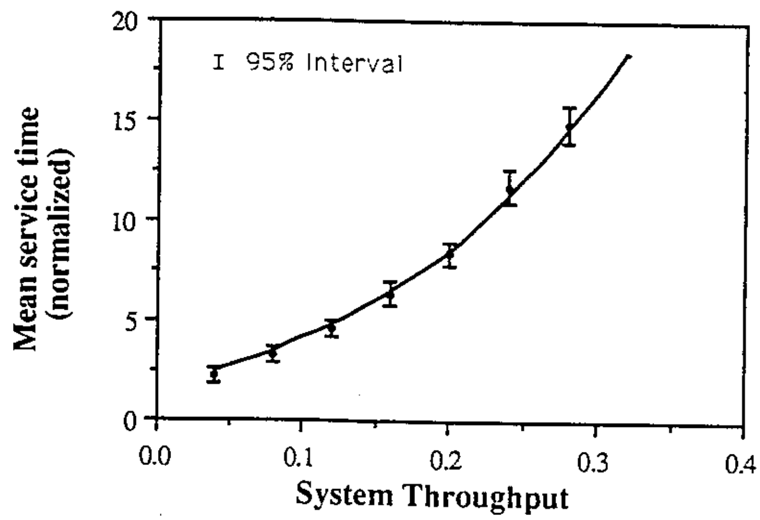


Figure 6.7: Mean service time versus throughput for M = 20, p = 0.05.

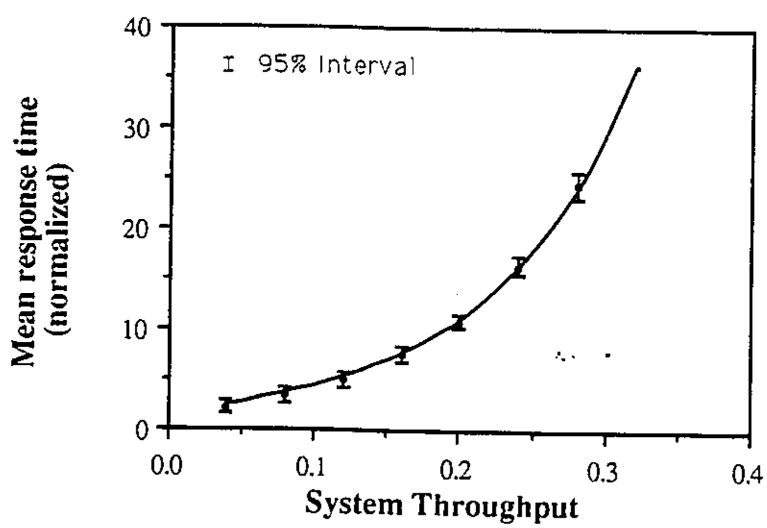


Figure 6.8: Mean response time versus throughput for M = 20, p = 0.05.

Chapter 7

Conclusion

The M-interference queueing problem in shared-channel communication networks has been studied. An effective approach, called the decomposition method, was proposed. This approach decomposes the interaction of M-interference stations into a set of simple queues, and the performance of the symmetric shared-channel communication networks can be evaluated through the analysis of the queueing behavior of only one station using queueing theory. Three typical examples, namely TDMA systems, token ring systems and slotted ALOHA systems, were chosen to illustrate this approach. Simulation programs were also developed and run to verify the analytical results.

For the TDMA system, the analytical results for both the packet delay performance and the message delay performance were obtained. The analytical results were compared to existing results as well as simulation results and found to match exactly. This indicates that the TDMA system is decomposable.

For the token ring system, two service policies were studied. The values predicted by the analysis were compared to simulation results and found to be accurate. This study reveals that the token ring network can be decomposed in the exhaustive service policy and can be approximated as a decomposable system in the single service policy. This suggests that the decomposition method provides an alternative but an effective approach to analyze other polling and time-token systems.

For the slotted ALOHA system with a finite number of stations, each with an infinite buffer, the delay performance was obtained by using the decomposition approach. The values predicted by this approximation were compared to the simulation results and found to be in good agreement.

In summary, our results indicate that the decomposition method is feasible for the analysis of M-interference queueing problem. With the help of more queueing theory tools, we hope to extend this method to other systems, including those networks with asymmetric properties and service characteristics, for example, nonsymmetrical token ring systems.

Bibliography

[1] **QNAP2 Reference Manual**, BULL and INRIA, 1983,1984.

[2] N. Abramson, "The ALOHA system another alternative for computer communications", *AFIPS Conf. Proc.*, Vol. 37, pp 281–285, 1970.

[3] W. Bux, "Token ring local area networks and their performance", *Proceedings of IEEE,* Vol. 77, No. 2, pp 238–256, Feb. 1989.

[4] P. Chen and J.F. Chang, "An improved transmission protocol for two interfering queues in packet radio networks", *IEEE Trans. Commun.*, Vol. COM-39, No. 3 pp 353–355, March 1991.

[5] W.W. Chu and A.G. Konheim, "On the analysis and modeling of a class of computer communication systems", *IEEE Trans Commun.*, Vol. COM-20, No. 3, pp 645–660, June 1972.

[6] R. Cooper, **Introduction to Queueing Theory,** Second Edition, North-Holland Publishing Company, 1981.

[7] L. DeMoraes and I. Robin, "Message delay distributions for a TDMA scheme under a non-preemptive priority discipline", *IEEE GLOBECOM'82,* pp 637–642, November, 1982.

[8] R. H. Deng, X. Zhang and K. Huang, "Token-passing systems with batch arrivals and their application to multimedia file transfer over token ring", *IEEE INFOCOM'91,* pp 1041–1049, April 1991.

[9] B.T. Doshi, "Queueing systems with vacations-a survey", *Queueing Systems,* Vol. 1, No. 1, pp 29–66, June 1986.

[10] M. Eisenberg, "Queues with periodic service and changeover times", *Operations Research,* Vol.20, No.2, pp 440–451, March 1972.

[11] A. Ephremides and R.Z. Zhu, "Delay analysis of interacting queues with an approximate model", *IEEE Trans. Commun.,* Vol. COM-35, No. 2, pp 194–201, Feb. 1987.

[12] M.J. Ferguson and Y.J. Aminetzah, "Exact results for nonsymmetric token ring systems", *IEEE Trans. Commun.,* Vol. COM-33, No. 3, pp 223–231, March 1985.

[13] S. W. Fuhrmann and R. Cooper, "Stochastic decompositions in the M/G/1 queue with generalized vacations", *Oper. Res.,* Vol. 33, pp 1117–1129, 1985.

[14] A. Ganz and I. Chlamtac, "A linear solution to queuing analysis of synchronous finite buffer networks", *IEEE Trans. Commun.,* Vol. 38, No. 4, pp 440–446, April 1990.

[15] I. Habab, M. Kavehrad, and C. Sundberg, "Protocols for very high speed optical fiber local area networks using a passive star topology", *IEEE Journal of Lightwave Technology,* Vol. LT-5, No. 12, Dec. 1987.

[16] O. Hashida and K. Ohara, "Line accommodation capacity of a communication control unit", *Rev. Nippon Elec. Comm. Labs* Vol. 20, pp 231–239, March 1972.

[17] O. Hashida and K. Kawashima, "Analysis of a polling system with single user at each terminal", *Rev. Nippon Elec. Comm. Labs* Vol. 29, pp 245–253, March 1981.

[18] J.F. Hayes, "Performance models of an experimental computer communication network", *Bell System Technical Journal,* Vol.53, pp 225–259, Feb. 1974.

[19] J.F. Hayes, **Modeling and Analysis of Computer Communication Network,** Plenum Press, 1984

[20] O.C. Ibe and X. Cheng, "Approximate analysis of asymmetric single service token passing systems", *IEEE Trans. Commun.,* Vol. COM-37, No.6, pp 572–577, June 1989.

[21] S.S. Kamal and S. A. Mahmoud, "A study of user's buffer variations in random access satellite channels", *IEEE Trans. Commun.,* Vol. COM-27, No. 6, pp 857–868, June 1979.

[22] A.R. Kaye, "Analysis of a distributed control loop data transmission", *Proceedings of the Symposium on Computer-Communications Networks and Teletraffic,* pp 47–58, April 1972.

[23] L. Kleinrock, **Queueing Systems, Vol. 1, Theory.** John Wiley & Sons, New York, 1975.

[24] L. Kleinrock and S.S. Lam, "Packet switching in a multiaccess broadcast channel: performance evaluation", *IEEE Trans. Commun.,* Vol. COM-23, No. 4, pp 410–423, April 1975.

[25] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part I Carrier sense multiple access modes and their throughput-delay characteristics", *IEEE Trans Commun.,* Vol. COM-23, pp 1400–1416, Dec. 1975.

[26] L. Kleinrock, **Queueing Systems, Vol. 2, Computer Applications.** John Wiley & Sons, New York, 1976.

[27] L. Kleinrock and M.O. Scholl, "Packet switching in radio channels: New conflict-free multiple access schemes", *IEEE Trans Commun.,* Vol. COM-28, pp 1015–1029, July 1980.

[28] L. Kleinrock and Y. Yemini, "Inetrfering queueing processes in packet-switched broadcast communication", *Proc. IFIP Congress,* pp 557–562, Oct. 1980.

[29] P. J. Kuehn, "Multiqueue systems with nonexhaustive cyclic service", *Bell Syst. Tech. J.,* Vol. 58, pp 671–698, March 1979.

[30] S.S. Lam, "Delay analysis of a time division multiple access", *IEEE Trans Commun.,* Vol. COM-25, No. 12, pp 1489–1494, Dec. 1977.

[31] R.O. LaMaire, "An M/G/1 vacation model of an FDDI station", *IEEE J. Select. Areas Commun.,* Vol. 9. No. 2, pp 257–264, Feb. 1991.

[32] T.T. Lee, "M/G/1/N queue with vacation time and exhaustive service discipline", *Operations Research,* Vol. 32, No. 4, pp 775–784, July 1984.

[33] K.K. Leung and M. Eisenberg, "A single server queue with vacations and gated time-limited service", *IEEE Trans. Commun.,* Vol. COM-38, No. 9, pp 1454–1462, Sept. 1990.

[34] H. Levy and L. Kleinrock, "A queue with starter and a queue with vacations: delay analysis by decomposition", *Operations Research,* Vol. 34, No. 3, pp 426–436, May 1986.

[35] J.D.C. Little, "Proof of the queueing formula $L = \lambda W$", *Operations Research,* Vol. 9, No. 3, pp 383–386, 1961.

[36] L.W. Miller, **Alternating Priorities in Multiclass Queues,** Ph.D. Dissertation, Cornell University, New York, 1964.

[37] L. Roberts, "Extensions of packet commuincation technology to a hand held personal terminal", *AFIPS Conf. Proc.,* Vol. 40, pp 295–298, 1972.

[38] I. Robin, "Access control disciplines for multi access communication channels: reservation and TDMA schemes", *IEEE Trans. Information Theory,* Vol. IT-25, pp 516–536, Sept. 1979.

[39] M. Scholl and L. Kleinrock, "On the M/G/1 queue with rest periods and certain service-independent queueing disciplines", *Operations Research,* Vol. 31, No. 4, pp 705–719, July 1983.

[40] M. Schwartz, **Telecommunication Networks: Protocols, Modeling and Analysis,** Addison-Wesley Publishing Company, 1986.

[41] L.D. Servi, "Average delay approximation of M/G/1 cyclic service queues with Bernoulli schedules", *IEEE J. Select. Areas Commun.,* Vol. 4. No. 6, pp 813–822, Sept. 1986.

[42] M. Sidi and A. Segall, "Two interfering queues in packet-radio networks", *IEEE Trans. Commun.,* Vol. COM-31, pp 123–129, Jan. 1983.

[43] E.D. Sykas, D.E. Karvelas and E.N. Protonotarios, "Queuing analysis of some buffered random multiple access schemes", *IEEE Trans. Commun.,* Vol. COM-34, No. 8, pp 790–798, August 1986.

[44] H. Takagi, "Mean message waiting times in symmetric multi-queue systems with cyclic service", *IEEE International Conference on Communications,* pp 1154–1157, June 1985.

[45] H. Takagi, "On the analysis of a symmetric polling systems with single-message buffers", *Performance Evaluation,* Vol.5, No.3, pp 149–157, August 1985.

[46] H. Takagi, **Analysis of Polling Systems** MIT Press, Cambridge, MA, 1986.

[47] A.S. Tanebaum, **Computer Netwoks** Englewood Cliffs, NJ: Prentice Hall, 1988.

[48] F.A. Tobagi, "Multiaccess protocols in packet communication systems", *IEEE Trans Commun.,* Vol. COM-28, No. 4, pp 468–488, April 1980.

[49] P.D. Welch, "On a generalized M/G/1 queuing process in which the first customer of each busy period receives exceptional service", *Operations Research,* Vol. 12, No. 5, pp 736–752, Sept. 1964.

[50] T.Y. Yan, "On the delay analysis of a finite capacity TDMA channel", *IEEE Trans. Commun.,* Vol COM-30, No. 8, pp 1937–1942, August 1982.

Appendix A: Simulation Codes for TDMA

```
&**********************************************
& This program is for the simulation of
& delay performance
& in TDMA network by X. Yao
&                            July, 1991
&**********************************************
/CONTROL/
OPTION =NSOURCE;
OPTION =NRESULT;
MARGINAL = ALL QUEUE;

/DECLARE/

INTEGER             &********************************
    nb_ws=2 ,       & number of nodes in the network.
    i,idle,np,      &
    count_m;        & counter used in slot monitor.
                    &********************************

REAL                &********************************
    t_frame,        & frame duration, t_frame=mt_slot
    t_slot,         & packet transmission time= slot time
    prob,           & prob for the geometric distribution.
    mrt,crt,vrt,    & for the calculation.
    ar_rate;        & arrival rate
                    &********************************

QUEUE INTEGER    id;

QUEUE               &********************************
    init,           & initial queue
    MONITOR,        & frame monitor
    NODE(nb_ws),    & nodes in the network
```

```
    timer(nb_ws),   & to measure the response time
    F_source(nb_ws);& source station
                    &********************************

FLAG                &********************************
    slot(nb_ws),    & slot flag
    frame;          & frame flag
                    &********************************

&--------------------------------------------------------------------
/STATION/ NAME   =timer(1 STEP 1 UNTIL nb_ws);
          TYPE   =RESOURCE;
&--------------------------------------------------------------------
/STATION/ NAME   = init;           & flag initialization
          INIT   = 1;
       SERVICE   = BEGIN
                     SET (frame);
                     count_m := 1;
                     FOR i:=1 STEP 1 UNTIL nb_ws DO
                     BEGIN
                       RESET(slot(i));
                     END;
                     TRANSIT(OUT);
                   END;

&--------------------------------------------------------------------
/STATION/ NAME   = F_source;
          TYPE   = SOURCE;
       SERVICE   = BEGIN
                     EXP(1/ar_rate);
                     TRANSIT(NODE(id));
                   END;
&--------------------------------------------------------------------
/STATION/ NAME   = NODE;
       SERVICE   = BEGIN
```

```
P(timer(id));
np:= 1;
WHILE NOT(DRAW(prob)) DO
BEGIN
  np:= np +1;
END;

FOR i:= 1 STEP 1 UNTIL np DO
BEGIN
  WAIT(slot(id));
  CST(t_slot);
  RESET(slot(id));
  SET(frame);
END;
V(timer(id));
TRANSIT(OUT);
END;
&------------------------------------------------------------------
/STATION/ NAME    = MONITOR;
          INIT    = 1;
       SERVICE    = BEGIN
            WAIT(frame);
            count_m:=count_m+1-INTREAL(count_m/nb_ws)*nb_ws;
            IF ( NODE(count_m).NB >0 ) THEN
              BEGIN
                SET (slot(count_m));
                RESET(frame);
              END
            ELSE
            CST(t_slot);
            idle:=idle+1;
            IF (idle > nb_ws) THEN
            BEGIN
              idle:=0;
            END;
```

```
                    TRANSIT(MONITOR);
                  END;
&------------------------------------------------------------------

/CONTROL/TMAX =10;
     &  TRACE =50,100;
     ACCURACY =timer,NODE;
   ESTIMATION =SPECTRAL;

/EXEC/
&****************
& initilization
&****************
      BEGIN
        FOR i:=1 STEP 1 UNTIL nb_ws DO
        BEGIN
          NODE(i).id:=i;
          F_SOURCE(i).id:=i;
        END;

        prob:= 0.5;
        t_slot:=1.0;
        t_frame:=t_slot*nb_ws;

        PRINT("TDMA CHANNEL", nb_ws, "STATIONS");
        PRINT(" ");
        PRINT("MESSAGE LENGTH = GEOMETRIC WITH MEAN", 1/prob);
        PRINT(" ");
        FOR i:=1 STEP 1 UNTIL 1 DO
        BEGIN
          ar_rate:=0.2*i;
          PRINT(" ");
          PRINT("ARRIVAL RATE = ", ar_rate);

          mrt:=0;
```

```
            crt:=0;
            vrt:=0;

&*****************
& exec simulation
&*****************
         SIMUL;

&*****************
& results
&*****************

         FOR i:= 1 STEP 1 UNTIL nb_ws DO
           BEGIN
             mrt:= mrt + MRESPONSE(NODE(i))/nb_ws;
             crt:= crt + CRESPONSE(NODE(i))/nb_ws;
             vrt:= vrt + VRESPONSE(NODE(i))/nb_ws;
           END;

         PRINT("MEAN RESPONSE TIME = ", mrt);
         PRINT("CONFIDENCE INTERVAL =",crt);
         PRINT("VARIANCE =",vrt);
         END;
         END;
/END/
```

Appendix B1: Simulation Codes for Token Ring 1

```
&***********************************************
& This program is for the simulation of
& delay performance for exhaustive service
& in Token ring network
&                                by X. Yao
&                                March, 1991
&***********************************************

/CONTROL/
OPTION =NSOURCE;
OPTION =NRESULT;
MARGINAL = ALL QUEUE;

/DECLARE/

INTEGER
                    &****************************
     nb_ws=20,      & number of nodes on the ring.
     i,k,m,         & for do loop.
     idle,          & number of idle station.
     count_m;       & counter used in token monitor.
                    &****************************

REAL                &****************************
     mrt,           & mean response time.
     crt,           & confidence interval.
     t_walk,        & walk time between nodes (ms).
     t_packet,      & packet transmission time (ms).
     ar_rate;       & arrival rate
                    &****************************

QUEUE INTEGER    id;
```

```
QUEUE                 &****************************
    init,             & initial queue
    MONITOR,          & token monitor
    NODE(nb_ws),      & nodes on ring
    F_source(nb_ws);& source station
                      &****************************

FLAG                  &*********************************
    token(nb_ws),     & token flag
    circ;             & main flag for token circulation
                      &*********************************

&-------------------------------------------------------
/STATION/ NAME    = init;           & flag initialization
          INIT    = 1;
       SERVICE    = BEGIN
                      SET (circ);
                      count_m := 1;
                      FOR i:=1 STEP 1 UNTIL nb_ws DO
                      BEGIN
                        RESET(token(i));
                      END;
                      TRANSIT(OUT);
                    END;
&-------------------------------------------------------
/STATION/ NAME    = F_source;        & source stations
          TYPE    = SOURCE;
       SERVICE    = BEGIN
                      EXP(1/ar_rate);
                      TRANSIT(NODE(id));
                    END;
&-------------------------------------------------------
/STATION/ NAME   = NODE;
      SERVICE    = BEGIN
                     WAIT(token(id));
```

```
                    EXP(t_packet);
                    IF NODE(id).NB = 1 THEN
                    BEGIN
                      RESET(token(id));
                      SET(circ);
                    END;
                    TRANSIT(OUT);
                  END;
&-------------------------------------------------------
/STATION/ NAME    = MONITOR;
          INIT    = 1;
       SERVICE    = BEGIN
                     WAIT(circ);
                     CST(t_walk);
                     count_m:=count_m+1-INTREAL(count_m/nb_ws)*nb_ws;
                     IF ( NODE(count_m).NB >0 ) THEN
                     BEGIN
                       SET (token(count_m));
                       RESET(circ);
                     END
                     ELSE
                       idle:=idle+1;
                     IF (idle > nb_ws) THEN
                     BEGIN
                       idle:=0;
                     END;
                     TRANSIT(MONITOR);
                   END;
&-------------------------------------------------------

/CONTROL/TMAX =100000;
      & TRACE =50,60;
     ACCURACY =NODE;
   ESTIMATION =SPECTRAL;
```

```
/EXEC/
&****************
& initilization
&****************
      BEGIN
        FOR i:=1 STEP 1 UNTIL nb_ws DO
        BEGIN
          NODE(i).id:=i;
          F_source(i).id:=i;
        END;

        t_walk:=0.2/nb_ws;
        t_packet:=0.1;
        PRINT("SERVICE DISCIPLINE = EXHAUSTIVE");
        PRINT(" ");

        FOR k:=1 STEP 1 UNTIL 9 DO
        BEGIN
        ar_rate:=0.05*k;
        mrt:=0;
        crt:=0;

        PRINT("ARRIVAL RATE PER STATION = ",ar_rate);
        PRINT(" ");
&*****************
& exec simulation
&*****************
        SIMUL;

&*****************
& results
&*****************
        FOR m:= 1 STEP 1 UNTIL nb_ws DO
          BEGIN
            mrt:= mrt + MRESPONSE(NODE(m))/nb_ws;
```

```
        crt:= crt + CRESPONSE(NODE(m))/nb_ws;
      END;
      PRINT("MEAN RESPONSE TIME =", mrt);
      PRINT("CONFIDENCE INTERVAL =",crt);
      PRINT(" ");
    END;
    END;
/END/
```

Appendix B2: Simulation Codes for Token Ring 2

```
&**********************************************
& This program is for the simulation of
& delay performance for single service
& in Token ring network by X. Yao
&                            March, 1991
&**********************************************

/CONTROL/
OPTION=NSOURCE;
OPTION=NRESULT;

/DECLARE/

INTEGER                 &********************************
     nb_ws=10 ,         & number of nodes on the ring.
     i,k,               & for do loop.
     idle,              & number of idle station.
     np,                & number of packets in a message.
     count_m;           & counter used in token monitor.
                        &********************************

REAL                    &********************************
     mrt,               & mean response time
     crt,               & confidence interval
     prob,              & prob for the geometric distribution.
     t_walk,            & walk time between nodes (ms).
     t_packet,          & packet transmission time (ms).
     ar_rate;           & arrival rate
                        &*********************************

CUSTOMER INTEGER SEQ;

REF CUSTOMER    C;
```

```
QUEUE INTEGER   id;  & id number for a particular queue

QUEUE                 &******************************
     init,            & initial queue
     timer(nb_ws),    & to measure service time
     MONITOR,         & token monitor
     NODE(nb_ws),     & nodes on ring
     F_source(nb_ws),& source station
     PKT(nb_ws);      & packetize before transmission
                      &******************************

FLAG                  &******************************

     token(nb_ws),    & token flag
     circ;            & main flag for token circulation

&*****************************************************

&-----------------------------------------------------
/STATION/ NAME   =timer(1 STEP 1 UNTIL nb_ws);
          TYPE   =RESOURCE;
&-----------------------------------------------------
/STATION/ NAME   = init;            & flag initialization
          INIT   = 1;
       SERVICE   = BEGIN
                     SET (circ);
                     count_m := 1;
                     FOR i:=1 STEP 1 UNTIL nb_ws DO
                     BEGIN
                       RESET(token(i));
                     END;
                     TRANSIT(OUT);
                   END;
&-----------------------------------------------------
```

```
/STATION/ NAME    = F_source;
          TYPE    = SOURCE;
       SERVICE    = BEGIN
                      EXP(1/ar_rate);
                      TRANSIT(PKT(id));
                    END;
&--------------------------------------------------------
/STATION/ NAME   = PKT;              & packetization
      SERVICE    = BEGIN
                      P(timer(id));
                      np:= 1;
                      WHILE NOT(DRAW(prob)) DO
                      BEGIN
                      np:= np +1;
                      END;
                      IF (np<>1) THEN
                      BEGIN
                      FOR i:= 1 STEP 1 UNTIL np-1 DO
                      BEGIN
                        C:= NEW(CUSTOMER);
                        C.SEQ:= 0;
                        TRANSIT(C,NODE(id));
                      END;
                      END;
                      SEQ:=1;
                      TRANSIT(NODE(id));
                    END;
&-----------------------------------------------------------
/STATION/ NAME   = NODE;
      SERVICE    = BEGIN
                      WAIT(token(id));
                      CST(t_packet);
                      RESET(token(id));
                      SET(circ);
                      IF (SEQ<>1) THEN
```

```
                    TRANSIT(OUT)
                  ELSE
                    V(timer(id));
                  TRANSIT(OUT);
                END;
&----------------------------------------------------------
/STATION/ NAME    = MONITOR;
          INIT    = 1;
       SERVICE    = BEGIN
                  WAIT(circ);
                  CST(t_walk);
                  count_m:=count_m+1-INTREAL(count_m/nb_ws)*nb_ws;
                  IF ( NODE(count_m).NB >0 ) THEN
                  BEGIN
                    SET (token(count_m));
                    RESET(circ);
                  END
                  ELSE
                    idle:=idle+1;
                  IF (idle > nb_ws) THEN
                  BEGIN
                    idle:=0;
                  END;
                  TRANSIT(MONITOR);
                END;
&----------------------------------------------------------

/CONTROL/TMAX =30000;
      & TRACE =50,60;
     ACCURACY =PKT,timer;
   ESTIMATION =SPECTRAL;


/EXEC/
&****************
```

```
& initilization
&****************
      BEGIN
        FOR i:=1 STEP 1 UNTIL nb_ws DO
        BEGIN
          NODE(i).id:=i;
          PKT(i).id:=i;
          F_source(i).id:=i;
        END;

        prob:= 0.5;
        t_walk:=0.2/nb_ws;
        t_packet:=0.1;
        PRINT("SERVICE DISCIPLINE = SINGLE");
        PRINT(" ");
        PRINT("MESSAGE LENGTH = GEOMETRIC WITH MEAN", 1/prob);
        PRINT(" ");
        FOR k:=1 STEP 1 UNTIL 1 DO
        BEGIN
        ar_rate:=0.2*k;
        mrt:=0;
        crt:=0;
        PRINT(" ");
        PRINT("ARRIVAL RATE =",ar_rate);
&*****************
& exec simulation
&*****************
        SIMUL;

&*****************
& results
&*****************
        FOR i:= 1 STEP 1 UNTIL nb_ws DO
          BEGIN
            mrt:= mrt + MSERVICE(timer(i))/nb_ws;
```

```
            mrt:= mrt + MRESPONSE(PKT(i))/nb_ws;
            mrt:= mrt - MSERVICE(PKT(i))/nb_ws;
            crt:= crt + CSERVICE(timer(i))/nb_ws;
            crt:= crt + CRESPONSE(PKT(i))/nb_ws;
            crt:= crt - CSERVICE(PKT(i))/nb_ws;
          END;
          PRINT("MEAN RESPONSE TIME = ", mrt);
          PRINT("CONFIDENCE INTERVAL =",crt);
        END;
      END;
/END/
```

Appendix C: Simulation Codes for Slotted ALOHA

```
&*****************************************************************
&    Simulation of Slotted-ALOHA.
&    Several Workstations used.
&    Written by X. Yao, Oct.11, 1989.
&    Modified: July, 1991
&*****************************************************************

/CONTROL/
OPTION=NSOURCE;
OPTION=NRESULT;
MARGINAL = ALL QUEUE;

/DECLARE/


INTEGER             & *************************************
        nb_ws=20,   & number of workstations
        i,j,n,m,c,  & used in FOR statement
        br1,        & number of stations attempt to transmit
        number,     & busy station counter
        COLLI1;     & collision counter
                    & *************************************

REAL                &*************************************
        delay,      & average delay
        cdelay,     & confidence interval of delay
        s_time,     & mean service time
        cs_time,    & confidence interval of service time
        t_intarr,   & interarrive time
        prob;       & prob. that a data packet is transmited
                    &*************************************
```

```
CUSTOMER INTEGER    &*********************************************
        n_slot,     & number of slots for a backoff delay process
        m_slot,     &
        SEQ;        &
                    &*********************************************

QUEUE INTEGER id;  & id number for a particular queue

QUEUE               &********************************************

  init,             & init station
  F_source(nb_ws), & source  to generate packets
  MONITOR,          & to monitor the slot time
  THPT,             & to check the throughput
  timer(nb_ws),     & to measure the service time
  NODE(nb_ws);      & stations in the system

                    &********************************************

FLAG  net(nb_ws); & to slot the channel

&***************************************************************
&
& -----------------------------------------------------------------
/STATION/ NAME =timer(1 STEP 1 UNTIL nb_ws);
  TYPE          =RESOURCE;
&------------------------------------------------------------------

/STATION/ NAME=init;
  INIT=1;
  TYPE = INFINITE;
  SERVICE = BEGIN
              FOR i:=1 STEP 1 UNTIL nb_ws DO
              BEGIN
                RESET(net(i));
```

```
        END;
        TRANSIT(CUSTOMER,MONITOR);
      END;
&------------------------------------------------------------------
/STATION/
NAME= MONITOR;
SERVICE=  BEGIN
            br1:=0;
            FOR i:=1 STEP 1 UNTIL nb_ws DO
            BEGIN
              SET(net(i));
            END;
            CST(0.1*t_slot);
            FOR i:=1 STEP 1 UNTIL nb_ws DO
            BEGIN
              RESET(net(i));
            END;
            CST(0.9*t_slot);
            TRANSIT(MONITOR);
          END;
&------------------------------------------------------------------
/STATION/
NAME=Source;
TYPE= SOURCE;
SERVICE= BEGIN
           EXP(t_intarr);
           IF (NODE(id).NB <> 0 ) THEN
           BEGIN
             SEQ:= 1;
           END ELSE
           BEGIN
             SEQ:= 0;
           END;
           TRANSIT(NODE(id));
         END;
```

```
& ------------------------------------------------------------------
&
/STATION/
NAME = NODE(1 STEP 1 UNTIL nb_ws);
TYPE = SERVER;
SERVICE = BEGIN
            P(timer(id));
            IF (SEQ <> 0) THEN
            BEGIN
              m_slot:=0;
              WHILE NOT(DRAW(prob)) DO
              BEGIN
                m_slot:=m_slot + 1;
              END;
              CST(m_slot*t_slot);
            END;

            WAIT(net(id));
            br1:=br1 + 1;
            COLLI1:=br1;
            CST(t_slot);

            WHILE  (COLLI1 > 1) DO
            BEGIN
              n_slot:=0;
              WHILE NOT(DRAW(prob)) DO
              BEGIN
                n_slot:=n_slot+1;
              END;
              CST(n_slot*t_slot);
              WAIT(net(id));
              br1:=br1 + 1;
              COLLI1:=br1;
              CST(t_slot);
```

```
            END;

            V(timer(id));
            TRANSIT(THPT);
          END;
&----------------------------------------------------------------
/STATION/
NAME= THPT;
TYPE= SERVER;
SERVICE= BEGIN
           CST(t_slot);
           TRANSIT(OUT);
         END;

&----------------------------------------------------------------
/CONTROL/
 ACCURACY  = NODE, timer;
 ESTIMATOR = SPECTRUM;
     &RACE = 1,80;
      TMAX = 600000;

&----------------------------------------------------------------

/EXEC/
& *****************
& initialization
& *****************
  BEGIN
  FOR i:=1 STEP 1 UNTIL nb_ws DO
   BEGIN
    F_source(i).id:=i;
    NODE(i).id:=i;
   END;

  FOR j:=1 STEP 1 UNTIL 8 DO
```

```
  BEGIN
   delay:=0;
   cdelay:=0;
   s_time:=0;
   cs_time:=0;

   prob:=0.1;
   t_slot:=1;
   t_intarr:=1000/(2*j);

&*****************
& exec simulation
&*****************
  SIMUL;

&*****************
& results
&*****************
  FOR i:=1 STEP 1 UNTIL nb_ws DO
  BEGIN
    delay:=delay + MRESPONSE(NODE(i))/nb_ws;
    cdelay:=cdelay + CRESPONSE(NODE(i))/nb_ws;
    s_time:=s_time + MSERVICE(timer(i))/nb_ws;
    cs_time:=cs_time + CSERVICE(timer(i))/nb_ws;
  END;
  PRINT("DELAY=",delay,"Confidence Interval=",cdelay);
  PRINT("SERVICE TIME=",s_time,"Confidence Interval=",cs_time);
  PRINT("Throughput=",MTHRUPUT(THPT));
END;
END;
/END/
```