PERFORMANCE EVALUATION OF THE
MOVABLE-SLOT TDM PROTOCOL
AND ITS APPLICATION
IN METROPOLITAN AREA NETWORKS

by

LENNY KWOK-MING HON
B.Sc., Chinese University of Hong Kong, 1985

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
Department of Electrical Engineering

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA
July, 1987

©Lenny Kwok-Ming Hon, 1987
In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical Engineering
The University of British Columbia
1956 Main Mall
Vancouver, Canada
V6T 1Y3

Date July 30, 1977
Abstract

Movable-slot time-division multiplexing (MSTDM) is a medium access control protocol for the integration of voice and data in local area networks. In this thesis, the performance of this protocol is evaluated through mathematical analysis and simulation. Its application in metropolitan area networks is also studied.

For the performance evaluation, a non-preemptive priority queueing model is first proposed for analysing the mean data delay characteristic of the slotted non-persistent carrier-sense multiple access with collision detection (CSMA/CD) protocol. Then this analytical approach is extended to the slotted MSTDM protocol with non-persistent data packet transmission, and its mean data delay performance is obtained. Numerical results from the analysis are shown and discussed.

Moreover, simulation study of the MSTDM protocol is performed. Through the simulation results, the effects of this protocol on the general delay performances of voice and data are discussed. It is found that if first voice packets, which are generated at the beginning of talkspurts, are given a shorter retransmission delay than data packets, the channel-acquisition delay for voice sources can be reduced without sacrificing the data delay performance significantly. The simulation results are also used to verify the analytical results. As the comparisons show, the accuracy of the analysis is high although it is based on a simple approximate model.

For the application of MSTDM in metropolitan area networks, a scheme which alleviates the distance and transmission rate constraints associated with this protocol is described. The approach is to divide the stations in a large area into regional groups, each operating in a different frequency band. Each group forms a
sub-network which is part of the metropolitan area network. An access protocol is proposed for interconnecting these sub-networks. Also an analysis which finds the optimum number of sub-networks for interconnection is presented. The criterion is to minimize the mean data delay for communications in a sub-network.
# Table of Contents

Abstract ............................................................................................................. ii
List of Figures ..................................................................................................... vi
List of Tables ...................................................................................................... viii
Acknowledgements ............................................................................................ ix

1 Introduction

1.1 Background .................................................................................................. 1
1.2 Objective and Outline of the Thesis ......................................................... 5

2 Performance Analysis of CSMA/CD and MSTDM 8

2.1 CSMA/CD .................................................................................................. 9
  2.1.1 Protocol Description ............................................................................. 9
  2.1.2 Analytical Model ................................................................................. 10
  2.1.3 Delay Analysis ....................................................................................... 12
  2.1.4 Numerical Results ............................................................................... 14

2.2 MSTDM ..................................................................................................... 16
  2.2.1 Protocol Description ............................................................................. 16
  2.2.2 Analytical Model ................................................................................. 21
  2.2.3 Delay Analysis ....................................................................................... 24
  2.2.4 Numerical Results ............................................................................... 30

2.3 Summary ..................................................................................................... 37

3 Simulation of MSTDM 38

3.1 Description of the Simulation Model ....................................................... 39
3.2 Simulation Results and Discussions ....................................................... 42
3.3 Summary

4 MSTDM Network for a Metropolitan Area

4.1 Homenet Network

4.2 Description of the Access Protocol

4.2.1 Interconnection

4.2.2 Other Related Issues

4.3 Optimum Number of Homenets

4.3.1 Analysis

4.3.2 Numerical Results

4.4 Summary

5 Conclusions

Appendix A

Appendix B

References
List of Figures

2.1 Packet transmission using the slotted non-persistent CSMA/CD protocol . 9
2.2 Non-preemptive priority queueing model for CSMA/CD .................. 11
2.3 Data delay-throughput characteristics for CSMA/CD \( k = 2 \) ............... 15
2.4 First voice packet transmission using the slotted non-persistent MSTDM protocol ................................................................. 18
2.5 Subsequent voice packet transmission ........................................ 18
2.6 Packet formats for MSTDM ...................................................... 20
2.7 Periodic transmission of voice packets and their repositioning when delayed by other transmission ................................. 22
2.8 Non-preemptive priority queueing model for MSTDM .............. 23
2.9 Remaining transmission time seen by a class 1 arrival ............... 26
2.10 Data delay-throughput characteristics for MSTDM \( \alpha = 0.0125, k = 3 \) . 32
2.11 Data delay-throughput characteristics for MSTDM \( \alpha = 0.05, k = 2 \) .... 33
2.12 Data delay versus fraction of data load for MSTDM \( \alpha = 0.0125, k = 3 \) . 35
2.13 Data delay versus fraction of data load for MSTDM \( \alpha = 0.05, k = 2 \) ... 36

3.1 Delay characteristics of data packets when both first voice packets and data packets use BEB for retransmission \( \alpha = 0.0125, k = 3 \) .......... 43
3.2 Delay characteristics of voice packets when both first voice packets and data packets use BEB for retransmission \( \alpha = 0.0125, k = 3 \) .......... 44
3.3 Delay characteristics of data packets when both first voice packets and data packets use BEB for retransmission \( \alpha = 0.05, k = 2 \) ............... 45
3.4 Delay characteristics of voice packets when both first voice packets and data packets use BEB for retransmission \( \alpha = 0.05, k = 2 \) ............... 46
3.5 Delay characteristics of data packets when first voice packets use LIB and
data packets use BEB for retransmission ($\alpha = 0.0125, k = 3$) .................. 49
3.6 Delay characteristics of first voice packets when first voice packets use
LIB and data packets use BEB for retransmission ($\alpha = 0.0125, k = 3$) .... 50
3.7 Delay characteristics of voice packets when first voice packets use LIB
and data packets use BEB for retransmission ($\alpha = 0.0125, k = 3$) ........ 51
3.8 Delay characteristics of data packets when first voice packets use LIB and
data packets use BEB for retransmission ($\alpha = 0.05, k = 2$) .................. 52
3.9 Delay characteristics of first voice packets when first voice packets use
LIB and data packets use BEB for retransmission ($\alpha = 0.05, k = 2$) ...... 53
3.10 Delay characteristics of voice packets when first voice packets use LIB
and data packets use BEB for retransmission ($\alpha = 0.05, k = 2$) ........... 54

4.1 Homenet network on a subsplit cable system ........................................ 60
4.2 Inter-homenet packet transfer of bursty data ........................................ 62
4.3 Link set-up for conversation between two stations in different homenets .. 64
4.4 Mean data delay for transmission within homenet as a function of the
number of homenets at $p_v = 0$ ................................................................. 70
4.5 Mean data delay for transmission within homenet as a function of the
number of homenets at $p_v = 0.2$ ......................................................... 71
4.6 Mean data delay for transmission within homenet as a function of the
number of homenets at $p_v = 0.4$ ......................................................... 72
List of Tables

B.1 95% Confidence intervals for the CSMA/CD simulation results shown in 
    Figure 2.3 ($k = 2$) .................................................................80

B.2 95% Confidence intervals for the MSTDM simulation results shown in 
    Figure 2.10 ($\alpha = 0.0125, k = 3$) ........................................83

B.3 95% Confidence intervals for the MSTDM simulation results shown in 
    Figure 2.11 ($\alpha = 0.05, k = 2$) ............................................86
Acknowledgements

I would like to thank my supervisor, Dr. H. W. Lee, for his encouragement and advice. Financial support from his NSERC grant No. A5986 is also gratefully acknowledged.
Chapter 1

Introduction

1.1 Background

The concept of local area networks (LANs) emerged when there was a dramatic increase in the number of small independent computing systems, especially in the form of microcomputers, as a result of the continuing decline in the cost of computing hardware. In an establishment, these machines may be dedicated to perform specific functions. However, there is a need for them to communicate with each other in order to exploit the advantages of functionally distributed computing and share the resources effectively. This communications network is regarded as a LAN which interconnects locally distributed computing systems.

Generally, a LAN should satisfy a broad set of requirements such as the ones listed as below [1]:

- high data rates (typically 1 to 10 Mbps);
- geographic distance spanning from one hundred meters to a few kilometers;
• ability to support a few hundred independent devices;

• good reliability;

• fair access to the network by all devices;

• easy installation of a small system, with graceful growth as the network evolves;

• ease of reconfiguration and maintenance; and

• low cost.

The design of LANs involves two important issues which are the topology of configuration and the coordination of access to the network (or the medium access control protocol) [2]. They are often closely related to one another. Two general classes of physical link topologies are the ring and the bus. In the ring topology, the access methods typically employed are slotted-ring technique [3] and implicit token-passing [4]. In the bus topology, the more common ones are contention technique [5] and explicit token-passing [6]. These protocols provide access by the nodes to the transmission medium in a distributed coordination with packet-switched communications.

One of the most successful designs of LANs, which attempt to meet the requirements listed above, is the Ethernet [7]. It employs a bus communications channel which provides a passive broadcast medium. The access protocol used is the carrier-sense multiple access with collision detection (CSMA/CD) protocol. It is a contention protocol which allows a number of users to share a single channel in a random access manner with no central control.
The LAN was primarily designed for data service. However, office automation has necessitated the integration of voice and data into a single LAN. There are two potential advantages of this implementation:

- cost savings gained from the sharing of the transmission and the switching facilities; and

- provisions of more sophisticated services to the users.

Yet integrating voice and data creates a new set of communications problems, which are mainly due to their different traffic characteristics. Voice is periodic traffic that requires real-time delivery while data is bursty traffic that can tolerate long delays. Therefore, to make the integration possible, new medium access control protocols have to be applied, which can fulfil the different requirements for both voice and data traffic in LANs.

There have been many new protocols proposed in this research area. When applied to a (implicit or explicit) token-passing or a slotted-ring network, they take the advantage of the round-robin approach in the access method, which bounds the channel access delay, to meet the demands of the periodic traffic from voice sources [8][9]. However, because the channel access delay in contention technique is unbounded, it is not suitable for integrating voice and data. Usually it has to form a hybrid with the reservation scheme to suit this task [10].

Because of the popularity of applying the CSMA/CD protocol in LANs, typified by the numerous Ethernet installations, it can be a very attractive feature if these CSMA/CD LANs can be upgraded to provide voice and data services with minimal
CHAPTER 1. INTRODUCTION

changes in the system. The enhancement can be achieved by applying the movable-slot time-division multiplexing (MSTDM) protocol to the network [11][12]. It is a simple variation on CSMA/CD, which enables voice and data integration in a random access manner and has an upper bound on voice packet delay. In this protocol, data sources follow the same set of rules for transmission as in CSMA/CD. Therefore, in migrating from a CSMA/CD LAN to a MSTDM LAN, it is not necessary to modify the data stations. New voice stations and voice/data hybrid stations can be added onto the network simply by tapping to the passive transmission medium. As a result, MSTDM lends itself to be an attractive protocol for the integration of voice and data.

LANs are limited by their geographic distance. They can work well over a distance of a few kilometers. However, as they elongate, problems such as performance degradation and decline in reliability occur [13]. Solving the distance limitation problem can increase the attractiveness of the office information system so that office branches located at distant locations in a metropolitan area can become part of the network and can share information of voice and data [14–17]. Moreover, different companies can participate in the same metropolitan area network (MAN) and exchange information with each other, in addition to communicating within their own premises through the MAN. If the MAN supporting voice and data services can be expanded to the community, the local telephone subscriber loops can even be replaced by the network. Besides, new and advanced services such as home banking and home shopping can be provided in cooperation with the banking and merchandising companies. However, back to the problem addressed at the beginning, we first have to solve the distance constraint associated with the networks before these
new concepts can be put into realization.

The introduction above has briefly reviewed the evolution from LANs providing data-only service to MAN supporting voice and data integration. In the following parts of the thesis, attention is focused on the MSTDM protocol. The performance of this protocol is evaluated through mathematical analysis and simulation, for its application in LANs [18-20]. The method of applying the protocol to the MANs, which have the potentialities to provide the services as described in the previous paragraph, is also studied [21]. Because of the suitability of the analytical model and the close relationship between CSMA/CD and MSTDM, a minor part of the thesis is devoted to the performance analysis of the CSMA/CD protocol.

1.2 Objective and Outline of the Thesis

There have been some analyses on the performance of CSMA/CD in the literature, e.g. [22], [23] and [24]. In the first part of Chapter 2, a non-preemptive priority queueing model is proposed for analysing the mean delay-throughput performance of the slotted non-persistent CSMA/CD protocol. This analytical model presents a simpler way of performance evaluation than the previous works. We shall see later that although it is a simple approximate model, it provides very promising explicit results which are verified by simulation.

Unlike CSMA/CD, there has been no analytical result of the data delay performance of MSTDM before. In the second part of Chapter 2, a queueing model, which is similar to the one for CSMA/CD, is formulated for analysing the mean delay-throughput characteristic of data packets for the slotted MSTDM protocol.
with non-persistent data packet transmission. Numerical results are obtained from the analysis and compared with simulation results. The comparisons show that the analytical model proposed yields accurate results for the data delay performance of MSTDM.

The MSTDM protocol can be considered as a more general case of the CSMA/CD protocol, which allows the integration of voice and data. Without voice, MSTDM degenerates to CSMA/CD. The purposes of including the performance analysis of CSMA/CD here are twofold. One is to look into the data delay performance of this more 'specific' protocol directly in our new approach. Another is to show the accuracy of this modeling approach before we apply it to analyse the MSTDM protocol.

The analysis of MSTDM gives only the mean of data delay, but not its higher moments as well as any information on voice delay characteristics. In order to have a thorough understanding on the characteristics of MSTDM and to provide a means for verifying the delay analysis in Chapter 2, simulation was performed for this protocol.

Since the first voice packet generated at the beginning of a talkspurt from an active voice source uses CSMA/CD as a data packet in the MSTDM system, we may give the former a shorter retransmission delay than the latter so as to reduce the channel-acquisition time for voice sources, as explained in Section 2.2.1. In the simulation studies, the same as well as different retransmission scheduling algorithms were applied to data packets and first voice packets. In each case, the effects of the MSTDM protocol on the delay performances of data packets, first
voice packets and subsequent voice packets were obtained in terms of the mean and the standard deviation of their delay. Chapter 3 shows the simulation model, the simulation results and the discussions.

Because MSTDM is a bus contention protocol, its efficiency decreases inversely with the transmission rate and the channel length. The protocol works well on local area networks where the distance between the stations at the two ends is typically less than a few kilometers and the bit rate is in the order of 1 to 10 Mbps. When it is applied to a metropolitan area network with a distance being an order of magnitude longer, the transmission rate should be decreased by an order of magnitude in order to maintain the same efficiency. However, this is incapable of providing adequate data and voice services to such a large community of users.

In Chapter 4, a scheme which was proposed in [16] to enable the MSTDM protocol to be applied efficiently to a large network is described. The approach is to divide the system into a number of broadband local area networks called homenets. However, the method for inter-homenet communications is not clearly examined in [16]. In this thesis, we enhance the scheme by proposing an access protocol for the interconnection of homenets. It facilitates the communications between stations in different homenets for both voice and data services. The issues about traffic control and addressing are also mentioned. Moreover, based on the analytical results obtained for MSTDM in Chapter 2, we find the optimum number of homenets for interconnection, which minimizes the mean data delay within homenet.

Finally, Chapter 5 contains the conclusions which review the major results of the research work.
Chapter 2

Performance Analysis of CSMA/CD and MSTDM

In this chapter, a non-preemptive priority queueing model is first proposed for the performance evaluation of the slotted non-persistent CSMA/CD protocol. Then it is extended to a more 'general' voice/data model for the MSTDM protocol, a variation on CSMA/CD, which enables voice and data integration in a random access manner with an upper bound on voice packet delay. For each of the two protocols, the analysis yields explicit results for the mean delay-throughput characteristics of data packets. Numerical results are presented and discussed. As comparisons with simulation results show, the accuracy of the analysis is high although it is based on a simple approximate model.
2.1 CSMA/CD

2.1.1 Protocol Description

The slotted non-persistent CSMA/CD protocol is considered here. In this protocol, the time axis is (mini-)slotted with slot size equal to the end-to-end channel propagation delay ($\tau$ second). Transmission can only be started at the beginning of a slot. All data sources are synchronized to follow this restriction. When a data source has a packet ready for transmission, it senses the channel and proceeds as follows (also see Figure 2.1).

1. If the channel is sensed idle, it initiates transmission of the packet.

2. If a collision is detected during transmission, the transmission is aborted and the packet is scheduled for retransmission at some later time which is deter-
mined by a retransmission scheduling scheme. Then the data source repeats the algorithm.

3. If the channel is sensed busy, the data source reschedules the transmission of the packet to some later time using the retransmission scheduling scheme and repeats the algorithm.

### 2.1.2 Analytical Model

We assume, in the following, that the number of data sources in our CSMA/CD system is infinite and they collectively form an independent Poisson source with an aggregate mean packet arrival rate of $\lambda_2$ packets/second. Moreover, the system is stable so that the input rate of data packets is equal to the output rate.

There are two types of events that happen in the channel. They are (successful) data packet transmission and data packet collision. We attempt to consider them as two independent processes and we term packet collision as transmission of collision packet. By means of a suitable retransmission scheduling scheme [25], we assume that the probability of successful transmission in a contention slot is maintained to be constant, which is independent of the number of packets in the system. This constant probability is denoted by a factor $1/\nu$. Therefore, the mean arrival rate of collision packets is $\lambda_1 = (\nu - 1)\lambda_2$ packets/second.

If $\tau$ is so small in the CSMA/CD system that the duration of a collision can be neglected, we can approximate the system by the $M/G/1$ queue because the arrival of data packets is Poisson. As we now consider packet transmission and packet collision as two independent processes, we attempt to model each of them with the
CHAPTER 2. PERFORMANCE ANALYSIS OF CSMA/CD AND MSTDM

M/G/1 queue and they together form a non-preemptive priority queueing system (as shown in Figure 2.2) in which collision packets have a higher priority of transmission than data packets. The mean arrival rate of the former is $\lambda_1$ packets/second whereas that of the latter is $\lambda_2$ packets/second.

As the analytical model is prioritized, (successful) data packet transmission cannot occur while collision packet is present, which implies collision of data packets in the system. However, the model is non-preemptive so that a data packet will be transmitted until completion if it can acquire the channel successfully, i.e. no collision occurs at the beginning of the transmission. Under a stable condition of the model, the ratio of the number of data packet transmissions to the total of the two types of channel events is $1/\nu$ on the average.

In the next section, we shall use this model to find the delay-throughput performance of the slotted non-persistent CSMA/CD protocol.
2.1.3 Delay Analysis

For the M/G/1 model with two non-preemptive priorities like the one shown in Figure 2.2, we denote the packets in the higher priority queue as the class 1 packets and the ones in the lower priority queue as the class 2 packets. The lengths of class 1 packet and class 2 packet are respectively $\tilde{m}_1$ and $\tilde{m}_2$, which may be random variables of any distribution in second. The mean waiting time ($\bar{W}_2$) of class 2 packets in this model (see Appendix A) is given by

$$\bar{W}_2 = \frac{\lambda_1 \tilde{m}_1^2 + \lambda_2 \tilde{m}_2^2}{2(1 - \lambda_1 \bar{m}_1)(1 - \lambda_1 \tilde{m}_1 - \lambda_2 \bar{m}_2)}$$

(2.1)

where $\bar{m}_1$ and $\bar{m}_2$ are the mean lengths of class 1 packets and class 2 packets respectively, and $\bar{m}_1^2$ and $\bar{m}_2^2$ are the second moments.

To model the CSMA/CD protocol, we assume the length of collision packets, i.e. the collision recovery time, is a multiple of the end-to-end channel propagation delay $\tau$. Therefore,

$$\tilde{m}_1 = m_1 = k\tau$$

(2.2)

where $k$ is an integer. Because of propagation delay, the channel will not be sensed idle right after a packet has been successfully transmitted. Thus, the length of class 2 packet is longer than the actual length of data packet by some propagation delay. We assume the difference is one end-to-end channel propagation delay. Therefore,

$$\tilde{m}_2 = \tilde{m} + \tau$$

(2.3)

where $\tilde{m}$ is the length of data packet and is assumed to be a multiple of $\tau$. As the channel is slotted, a packet has an additional waiting time of half a slot, i.e. $\tau/2$, on
the average. This factor is not characterized by our analytical model. To make the correction, we have to add the term \( \tau / 2 \) to the mean waiting time of data packets in (2.1).

The mean delay of data packets is defined as the sum of the mean packet waiting time and the mean packet transmission time. For simplicity, here we let the length of data packets be constant, i.e. \( \bar{m} = m \). Using (2.1) to (2.3) and replacing \( \lambda_1 \) with \( \lambda_2 (\nu - 1) \), we obtain the normalized mean delay \( (\overline{D}_n) \) of data packets after dividing the mean delay by \( m \) and it is given by

\[
(\overline{D}_n) = \left[ 1 + \frac{\rho_d [k^2 \alpha^2 (\nu - 1) + (1 + \alpha)^2]}{2[1 - k \rho_d \alpha (\nu - 1)][1 - \rho_d (1 + \alpha + k \alpha (\nu - 1))]} + \frac{\alpha}{2} \right] \]  

where \( \rho_d = \lambda_2 m \) is the data throughput and \( \alpha = \tau / m \) is the normalized end-to-end channel propagation delay. Note that as \( \alpha \) goes to zero, (2.4) can be approximated by the M/D/1 delay equation (see e.g. [26]) which is given by

\[
(\overline{D}_n) = 1 + \frac{\rho_d}{2(1 - \rho_d)} \]  

To find the maximum data throughput that the CSMA/CD system can achieve, we have, from our queueing model, that

\[
\lambda_1 \bar{m}_1 + \lambda_2 \bar{m}_2 < 1 \]  

if the system is stable. This implies that for the CSMA/CD system,

\[
\rho_d < \frac{1}{1 + \alpha [k (\nu - 1) + 1]} \]  

If retransmission is optimized by means of a suitable adaptive retransmission scheduling scheme, the probability of successful transmission in a contention slot is
$1/e$, i.e. $1/\nu = 1/e$, which is the maximum for the slotted random access protocol [22][25]. Therefore, substituting $\nu = e$ into (2.7), we have

$$P_d < \frac{1}{1 + \alpha [k(e - 1) + 1]}.$$  \hspace{1cm} (2.8)

### 2.1.4 Numerical Results

We use (2.4) to obtain the mean delay-throughput characteristics for data packets of constant length. In the following examples, the collision recovery time is twice the end-to-end channel propagation delay, i.e. $k = 2$. Also we assume that retransmission is optimized so that we have $\nu = e$.

Figure 2.3 shows the analytical results together with the simulation results for $\alpha=0.001$, 0.01, 0.05, 0.1 and 0.2 respectively. We see from the figure that the curves obtained from both methods are very close to one another for each value of $\alpha$. The results show that the performance of the CSMA/CD protocol declines with $\alpha$ being increased. From light to medium load (relative to the maximum throughput achievable for a particular $\alpha$) of data traffic, the mean delay of data packets is quite small. However, it increases rapidly as the traffic becomes heavy. This is a well-known characteristic for the performance of data packets using random access protocol for transmission. Moreover, the maximum throughput that can be achieved increases with $\alpha$ being reduced because less channel capacity is wasted in collisions and removals of signal propagation after successful packet transmissions. When $\alpha$ is equal to 0.001, the delay curve of the CSMA/CD protocol is almost identical to the one obtained from the M/D/1 delay equation in (2.5).

Our simulation model for the slotted non-persistent CSMA/CD assumes an in-
Figure 2.3: Data delay-throughput characteristics for CSMA/CD ($k = 2$).
finite population of data sources which collectively form an independent Poisson source. To obtain the simulation results shown in Figure 2.3, we assume the length of data packets is constant and it takes the time of one end-to-end channel propagation delay to have the effects of signal propagation removed from the channel after a successful packet transmission. The adaptive retransmission scheduling algorithm that we use in the simulation is binary exponential backoff. As comparisons with the simulation results show, the accuracy of our analysis turns out to be high although it is based on a simple queueing model. The 95% confidence intervals for the CSMA/CD simulation results shown in Figure 2.3 are listed in Appendix B.1, together with the analytical results for easy comparisons.

2.2 MSTDM

2.2.1 Protocol Description

Speech from an active voice source consists of alternating intervals of talkspurt and silence. During talkspurt, voice packets are generated from digitized voice; however, there are none in a silent period. The MSTDM protocol enables voice and data integration in a random access manner [11][12]. As it is a variation on CSMA/CD, data packets are treated exactly the same in this protocol as in CSMA/CD. Moreover, the first voice packet generated at the beginning of a talkspurt from an active voice source follows the CSMA/CD protocol as a data packet. We consider here the slotted non-persistent MSTDM which has the same set of rules for both data packets and first voice packets as those given in section 2.1.1
When speech energy is detected at the beginning of a talkspurt, transmission of the first voice packet is activated as soon as voice bits of the first voice sample arrive. Depending on the waiting time of this packet, there is a variable number of voice bits in it. Yet there is a maximum number of voice bits, say S, that can be placed in the packet, because the packets generated by voice sources are constrained to have a fixed length as explained later. If there are more than S voice bits coded when the channel is acquired, only the last S bits are transmitted and the earlier ones are discarded; thus, the speech at the beginning is clipped. However, we shall see in the next chapter that we can give first voice packets a shorter retransmission delay than data packets in order to reduce the channel-acquisition time for voice sources. In this way, the chance of voice bits being discarded is considerably decreased, yet the performance of data packets is not deteriorated significantly.

After a voice source has successfully transmitted its first packet, it plans on transmitting its next packet in a fixed time $T$ which is its period of transmission (Figure 2.4). Within a period, new voice bits arrive and fill up the data area of the voice packet. Subsequent voice packets in a talkspurt after the first one follow a subset of the CSMA/CD rules. When a voice source has a packet (other than the first one in a talkspurt) ready for transmission, it senses the channel and proceeds as follows (also see Figure 2.5).

1. If the channel is sensed idle, it initiates transmission of the packet without listening to the channel.

2. If the channel is sensed busy, the voice source refrains from transmission and keeps on listening to the channel. When the channel becomes idle, it initiates
CHAPTER 2. PERFORMANCE ANALYSIS OF CSMA/CD AND MSTDM

Figure 2.4: First voice packet transmission using the slotted non-persistent MSTDM protocol.

Figure 2.5: Subsequent voice packet transmission.
transmission of the packet without listening to the channel.

(Note that the rules above represent the 1-persistent CSMA without collision detection protocol.)

For the slotted MSTDM, all voice sources are synchronized to start transmission only at the beginning of a slot as data sources. In the following, we call the subsequent voice packets in a talkspurt after the first one simply as voice packets and that first one as the first voice packet. We find that a voice packet may experience collision with data packet(s) and/or first voice packet(s) in the vulnerable interval. Moreover, extra voice bits keep on arriving during the waiting time of a voice packet, which cannot be carried in the data area of the packet. To overcome these problems, voice packets have a specific packet format which is shown in Figure 2.6, together with the format of data packets for comparison. It consists of a preempt area, a header area, a data area and an overflow area. When a voice source initiates transmission of a voice packet, it places a signal (which forms the preempt area of the packet) on the transmission media but does not send information during the vulnerable interval. This interval is long enough for data sources as well as voice sources transmitting their first voice packets to detect a collision, stop transmitting, and have the effects of their transmission removed from the channel before the voice source begins transmitting useful data. As first voice packets use CSMA/CD, they do not have a preempt area.

If we restrict the length of data packets to be equal to or less than that of voice packets, the maximum waiting time incurred by a voice source is never greater than one voice packet transmission time after it has successfully transmitted the first
voice packet in a talkspurt. As voice bits arrive continuously, the overflow area is used for accommodating those voice bits which arrive while a voice source is waiting for transmission, and it is long enough to carry all of the extra voice bits while a voice source is delayed by as much as one voice packet transmission time. The voice source loses the privilege of not listening while transmitting after a talkspurt is over. In the next talkspurt after a silent period, the first voice packet has to be transmitted according to the whole set of the CSMA/CD rules.

Also, collision among voice packets never occurs because voice sources plan on transmitting their next packet one period after they have successfully acquired the channel. For the transmission of the subsequent voice packets in a talkspurt after the first one, a data source cannot delay a voice source enough to make it collide with
another voice source. Figure 2.7 shows the situation when the periodic transmission of voice packets is delayed by a data source. We see that voice sources appear to acquire TDM slots for transmission. Yet these periodic slots are repositioned when other transmissions interfere with them. That is why it is called movable-slot TDM protocol.

### 2.2.2 Analytical Model

In the following, the phrase ‘data packets’ includes both actual data packets and first voice packets. We assume the former and the latter have the same retransmission delay distribution so that they behave without any difference from each other in the system.

There are three types of events in the MSTDM channel. They are voice packet transmission, (data) packet collision and (successful) data packet transmission. We know from the protocol description that collision of voice packet with data packet has no effect on the voice packet transmission because useful information is not sent in the preempt area of voice packet. Thus, a voice source transmitting voice packet is oblivious of data packet transmission and once the channel becomes idle, a ready voice source can acquire the channel successfully. Voice packet transmission has, therefore, the highest priority of acquiring the channel among the three types of channel events, but it is non-preemptive as voice sources listen before transmitting.

From section 2, we see that the performance of data packets for CSMA/CD can be approximately analysed by a non-preemptive priority queueing model. Here we attempt to extend this model to include voice packets for formulating an approx-
Figure 2.7: Periodic transmission of voice packets and their repositioning when delayed by other transmission.
imate model for the MSTDM protocol. Then we solve for the delay performance of data packets. Our extended model as shown in Figure 2.8 is the M/G/1 system with three non-preemptive priorities. Queue 1 which has the highest priority of transmission contains voice packet. The next priorities are queue 2 and then queue 3, which contain collision packets and data packets respectively. Note that queue 1 is single-buffered whereas the other two have infinite storage. Without queue 1, the model in the present case is identical to the one in Figure 2.2. We have the same assumptions for data packets and collision packets here as those given in section 2.1.2.

Assuming the number of active voice sources in our system is large (but finite), we attempt to replace the process of voice packet generation by an independent
Poisson source. However, not all the voice packets generated by this source contribute to the voice traffic in our model as we shall see below. For the MSTDM protocol, we know that there is no contention of voice packet transmissions, since at most one voice packet can be waiting for transmission at a time. To characterize this property in our model, we assume that queue 1 is single-buffered so that any voice packet arrival from the Poisson source is blocked off and lost while one voice packet is waiting for transmission in the queue. Only those voice packets that get into the queueing system for transmission form the voice traffic for our analysis.

In the next section, we shall make use of this model to analyse the data delay performance of the MSTDM protocol with voice throughput in the channel as a system parameter.

### 2.2.3 Delay Analysis

For the queueing model shown in Figure 2.8, we denote the packet in queue $i$ as the class $i$ packet where $i = 1, 2$ or 3. We assume that the lengths of the three classes of packets, which are denoted by $m_1$, $m_2$ and $m_3$ for classes 1, 2 and 3 respectively, are constant for simplifying the analysis. Moreover, the mean packet arrival rates at the queues are $\lambda_1$, $\lambda_2$ and $\lambda_3$.

Under a stable condition, the throughputs of class 2 and class 3 packets are $\lambda_2 m_2$ and $\lambda_3 m_3$ respectively. To calculate the throughput of class 1 packets, we first denote their mean waiting time in queue 1 by $\bar{W}_1$. Since queue 1 has only one buffer, all packet arrivals are simply blocked off if it is occupied. From the time that a class 1 packet leaves the queue for transmission to the next arrival, the
CHAPTER 2. PERFORMANCE ANALYSIS OF CSMA/CD AND MSTDM

Figure 2.9: Remaining transmission time seen by a class 1 arrival.

mean duration is $1/\lambda_1$ because the interarrival time is exponentially distributed. The mean number of class 1 arrivals at the queue is, by expectation, given by

$$\frac{1}{W_1 + \frac{1}{\lambda_1}} = \frac{\lambda_1}{1 + \lambda_1 W_1}.$$  \hspace{1cm} (2.9)

Thus, the throughput of class 1 packets is $\lambda_1 m_1/(1 + \lambda_1 W_1)$.

Define

$$\rho_1 = \frac{\lambda_1 m_1}{1 + \lambda_1 W_1}, \quad \rho_2 = \lambda_2 m_2, \quad \rho_3 = \lambda_3 m_3.$$  \hspace{1cm} (2.10)

We are next going to find the mean waiting times of class 1, 2 and 3 packets in the queues, which are respectively denoted by $W_1, W_2$ and $W_3$.

Class 1

Assume there is a class 1 arrival while another class 1 packet is being transmitted as shown in Figure 2.9. Let $\tilde{Y}_1$ denote the remaining transmission time seen by the
arriving class 1 packet. Now,

\[
\text{Prob}\left[ \hat{Y}_1 \leq y | \text{packet length} = m_1 \right] = \frac{e^{-\lambda_1 (m_1 - y)} (1 - e^{-\lambda_1 y})}{1 - e^{-\lambda_1 m_1}}
\]  

(2.11)

and its probability density function is

\[
f(y|\text{packet length} = m_1) = \frac{\lambda_1 e^{-\lambda_1 (m_1 - y)}}{1 - e^{-\lambda_1 m_1}}.
\]  

(2.12)

Thus,

\[
\bar{Y}_1 = \int_0^{m_1} y f(y|\text{packet length} = m_1) \, dy
\]

\[
= \frac{m_1}{1 - e^{-\lambda_1 m_1}} - \frac{1}{\lambda_1}.
\]  

(2.13)

Similarly, the mean remaining transmission times of class 2 and class 3 packets seen by a class 1 arrival (which successfully goes into queue 1) are, respectively,

\[
\bar{Y}_2 = \frac{m_2}{1 - e^{-\lambda_1 m_2}} - \frac{1}{\lambda_1}
\]  

(2.14)

and

\[
\bar{Y}_3 = \frac{m_3}{1 - e^{-\lambda_1 m_3}} - \frac{1}{\lambda_1}.
\]  

(2.15)

Thus, the mean waiting time of class 1 packets is given by

\[
\bar{W}_1 = \rho_1 \bar{Y}_1 + \rho_2 \bar{Y}_2 + \rho_3 \bar{Y}_3.
\]  

(2.16)

From (2.16), we have, after some manipulations,

\[
(\bar{W}_1)^2 - \beta \bar{W}_1 + \gamma = 0
\]  

(2.17)

where

\[
\beta = \frac{\lambda_2 m_2^2}{1 - e^{-\lambda_1 m_2}} + \frac{\lambda_3 m_3^2}{1 - e^{-\lambda_1 m_3}} - \frac{1}{\lambda_1} \left(1 + \lambda_2 m_2 + \lambda_3 m_3\right)
\]  

(2.18)
\[
\gamma = \frac{1}{\lambda_1} \left( m_1 + \frac{\lambda_2 m_2}{\lambda_1} + \frac{\lambda_3 m_3}{\lambda_1} \right) - \left[ \frac{m_1^2}{1 - e^{-\lambda_1 m_1}} + \frac{\lambda_2 m_2^2}{\lambda_1 (1 - e^{-\lambda_1 m_2})} + \frac{\lambda_3 m_3^2}{\lambda_1 (1 - e^{-\lambda_1 m_3})} \right].
\]

From (2.17), we have
\[
\bar{W}_1 = \frac{\beta + \sqrt{\beta^2 - 4\gamma}}{2}. \tag{2.20}
\]

**Class 2**

We have, by expectation, that the mean waiting time of class 2 packets is given by (see e.g. [26])
\[
\bar{W}_2 = \text{mean remaining transmission time seen by the class 2 arrival} + \text{mean transmission time for the packets in peer and higher priority classes, which have already been in the queues at the class 2 arrival} + \text{mean transmission time for the packets in higher priority class, which arrive during the waiting time of the class 2 packet}. \tag{2.21}
\]

This implies
\[
\bar{W}_2 = \frac{\rho_1 m_1}{2} + \frac{\rho_2 m_2}{2} + \frac{\rho_3 m_3}{2} + \sum_{i=1}^{2} \bar{N}_i m_i + \frac{\lambda_1 \bar{W}_2 m_1}{1 + \lambda_1 \bar{W}_1}. \tag{2.22}
\]

where \(\bar{N}_i\) represents the mean number of class \(i\) packets in queue \(i\) for \(i=1, 2\) or \(3\).

By Little's Formula, we have
\[
\bar{N}_1 = \frac{\lambda_1 \bar{W}_1}{1 + \lambda_1 \bar{W}_1}, \quad \bar{N}_2 = \lambda_2 \bar{W}_2. \tag{2.23}
\]
CHAPTER 2. PERFORMANCE ANALYSIS OF CSMA/CD AND MSTDM

Note that the term \( \frac{\lambda_1 W_2 m_1}{(1 + \lambda_1 W_1)} \) in (2.22) is an approximation for the third component of \( W_2 \) in (2.21), since the arrival distribution of class 1 packets (those which get into queue 1 successfully) is not Poisson because of the blocking. To simplify the analysis, we assume the arrival is Poisson with mean \( \frac{\lambda_1}{1 + \lambda_1 W_1} \) here. From (2.22), we have

\[
W_2 = \frac{\Phi + \lambda_1 W_1 (\Phi + m_1)}{(1 + \lambda_1 W_1)(1 - \lambda_2 m_2 - \lambda_1 m_1)} \tag{2.24}
\]

where

\[
\Phi = \frac{1}{2} \left( \frac{\lambda_1 m_1^2}{1 + \lambda_1 W_1} + \lambda_2 m_2^2 + \lambda_3 m_3^2 \right) \tag{2.25}
\]

Class 3

Similar to class 2, the mean waiting time of class 3 packets is given by

\[
W_3 = \frac{\rho_1 m_1}{2} + \frac{\rho_2 m_2}{2} + \frac{\rho_3 m_3}{2} + \sum_{i=1}^{3} \sum_{i=1}^{3} N_i m_i + \frac{\lambda_1 W_3 m_1}{1 + \lambda_1 W_1} + \lambda_2 W_3 m_2 \tag{2.26}
\]

where by Little's Formula,

\[
N_3 = \lambda_3 W_3 \tag{2.27}
\]

After some manipulations, we have, from (2.26),

\[
W_3 = \frac{\Phi + \lambda_1 W_1 (\Phi + m_1) + \lambda_2 m_2 W_2 (1 + \lambda_1 W_1)}{(1 + \lambda_1 W_1)(1 - \lambda_2 m_2 - \lambda_3 m_3) - \lambda_1 m_1} \tag{2.28}
\]

Recall that \( W_3 \) denotes the mean waiting time of data packets for the MSTDM protocol. To be specific for modeling the performance measure, we have to incorporate the system characteristics into the equations we have obtained above. For MSTDM, the length of data packets cannot be longer than that of voice packets. In
our model, as data packets represent a mixture of actual data packets and first voice packets, the assumption of constant packet length, which is made at the beginning of this section, has implicitly set the length of data packets to be equal to that of voice packets because first voice packets and voice packets are of the same fixed length. Here we denote their length by $m$ which is assumed to be a multiple of the end-to-end channel propagation delay $\tau$.

Define

$$\rho_d = \lambda_3 m, \text{ the data throughput} \tag{2.29}$$

$$\hat{\rho}_v = \lambda_1 m \tag{2.30}$$

$$\rho_v = \frac{\lambda_1 m}{1 + \lambda_1 W_1} = \frac{\hat{\rho}_v}{1 + \hat{\rho}_v W_1}, \text{ the voice throughput} \tag{2.31}$$

where $W_{1n}$ is the normalized value of $W_1$ w.r.t. $m$; and $W_{2n}$ and $W_{3n}$ have the similar meaning in the following. Using the same assumptions and symbols we have made for CSMA/CD in section 2.1.3, we have

$$W_{1n} = \frac{\beta_n + \sqrt{\beta_n^2 - 4\gamma_n}}{2} \tag{2.32}$$

where

$$\beta_n = \rho_d \left[ \frac{k^2 \alpha^2 (\nu - 1)}{1 - e^{-k\alpha \hat{\rho}_v}} + \frac{(1 + \alpha)^2}{1 - e^{-(1 + \alpha)\hat{\rho}_v}} \right] - \frac{1}{\hat{\rho}_v \{1 + \rho_d[1 + \alpha + k\alpha(\nu - 1)]\}} \tag{2.33}$$

$$\gamma_n = \frac{1}{\hat{\rho}_v} \left\{ 1 + \alpha + \frac{\rho_d}{\hat{\rho}_v} [1 + \alpha + k\alpha(\nu - 1)] \right\}$$

$$- \left\{ \frac{(1 + \alpha)^2}{1 - e^{-(1 + \alpha)\hat{\rho}_v}} + \frac{\rho_d}{\hat{\rho}_v} \left[ \frac{k^2 \alpha^2 (\nu - 1)}{1 - e^{-k\alpha \hat{\rho}_v}} + \frac{(1 + \alpha)^2}{1 - e^{-(1 + \alpha)\hat{\rho}_v}} \right] \right\}, \tag{2.34}$$

$$W_{2n} = \frac{\Phi_n + \hat{\rho}_v W_{1n}(1 + \alpha + \Phi_n)}{(1 + \hat{\rho}_v W_{1n})[1 - k\alpha(\nu - 1)\rho_d] - \hat{\rho}_v(1 + \alpha)} \tag{2.35}$$
where
\[ \Phi_n = \frac{1}{2} \left( \hat{\rho}_v (1 + \alpha)^2 + \rho_d [(1 + \alpha)^2 + k^2 \alpha^2 (\nu - 1)] \right) \cdot (2.36) \]

\[ W_{3n} = \frac{\Phi_n + \hat{\rho}_v W_{1n} (1 + \alpha + \Phi_n) + k \alpha (\nu - 1) \rho_d W_{2n} (1 + \hat{\rho}_v W_{1n})}{(1 + \hat{\rho}_v W_{1n}) [1 - \rho_d [1 + \alpha + k \alpha (\nu - 1)]] - \hat{\rho}_v (1 + \alpha)} \cdot (2.37) \]

We have mentioned before that because the channel is slotted, a packet has an additional waiting time of half a slot on the average. We neglected this factor in our analysis. Thus, we have to add the term \( r/2 \) to the mean waiting time of data packets in (2.37). Therefore, the normalized mean delay of data packets is, after some manipulations, given by
\[ \overline{D}_n = 1 + W_{3n} + \frac{\alpha}{2} \cdot (2.38) \]

We notice that in (2.38), the normalized mean data delay \( \overline{D}_n \) is expressed in terms of \( \rho_d \) and \( \hat{\rho}_v \). To compute the voice throughput \( \rho_v \) from given values of \( \rho_d \) and \( \hat{\rho}_v \), we first use (2.32) to find \( W_{1n} \). Then we can take (2.31) and the value of \( W_{1n} \) just obtained to calculate the voice throughput \( \rho_v \) to which the given pair of \( \rho_d \) and \( \hat{\rho}_v \) corresponds.

2.2.4 Numerical Results

In the following numerical results, we assume that retransmission of data packets is optimized by means of a suitable adaptive retransmission scheduling scheme so that the probability of success in contending for an empty slot among data sources
is $1/e$, i.e. $\nu = e$. In Figure 2.10 and Figure 2.11, we plot the data delay-throughput characteristics at various values of $\rho_v$ using numerical results as well as simulation results. The values of the system parameters used in Figure 2.10 are $\alpha=0.0125$ and $k=3$ whereas they are respectively $0.05$ and $2$ in Figure 2.11.

From the figures, we observe that the data delay-throughput characteristic for MSTDM displays the same trend as that for CSMA/CD. Also, the delay performance declines apparently when the voice throughput $\rho_v$ in the channel increases. This is obvious because voice packets occupy part of the channel capacity for transmission. With a higher voice throughput, the maximum of the data throughput that can be achieved is reduced relatively. When voice throughput decreases, the data delay curve approaches that of CSMA/CD as expected.

In our simulation model for the slotted MSTDM protocol, we assume there is an infinite population of data sources which collectively form an independent Poisson source. The number of voice sources is, on the other hand, finite and we assume all of them are active. By varying the number of voice sources, we can change the level of voice throughput in our system. As we have done for the analysis, the first voice packets are considered as part of the data packets in the simulation. The actual data packets have the same length as the packets generated by voice sources. Moreover, it takes the time of one end-to-end channel propagation delay to have the effects of signal propagation removed from the channel after a packet transmission. The adaptive retransmission scheduling algorithm used is binary exponential backoff which is applied to both data packets and first voice packets. A more detailed description of the simulation model is given in the next chapter. The 95% confidence intervals for the MSTDM simulation results shown in Figure 2.10
Figure 2.10: Data delay-throughput characteristics for MSTDM ($\alpha = 0.0125$, $k = 3$).
Figure 2.11: Data delay-throughput characteristics for MSTDM ($\alpha = 0.05$, $k = 2$).
and Figure 2.11 are listed in Appendix B.2, together with the analytical results for easy comparisons. Finally, we see from the figures that the curves obtained from the analysis and the simulation match very well with each other. This shows that our analysis gives very good approximate results for the data delay performance of the MSTDM protocol.

Putting the delay-throughput characteristics (from the numerical results) in Figure 2.10 and Figure 2.11 in another form, we have Figure 2.12 and Figure 2.13. They show the data delay versus the fraction of data load at constant channel throughput (sum of the throughputs of voice and data) for the two sets of parameters. There are some interesting points noted on these curves. We see from Figure 2.12 that the data delay is reduced as the fraction of data load increases. This is because if the fraction of the load generated by voice sources decreases (or the fraction of data load increases), the number of sources which have a higher priority of transmission than data sources decreases. As a result, the delay experienced by data sources decreases. In Figure 2.13, the data delay also shows the same characteristic at low channel throughput, but the decrease is comparatively smaller. As the channel throughput becomes rather high, it changes from decrease to increase. The reason for this is that when the load of channel traffic is heavy and collisions of data packets waste a significant portion of the channel capacity (the collision recovery time is 1/10 of the packet transmission time in Figure 2.13 whereas it is only 3/80 in Figure 2.12), the increase in the number of data collisions counteracts the effect of having fewer number of voice sources as the fraction of data load increases. Therefore, the increase in the data delay is a result of the increased-channel contention.
Figure 2.12: Data delay versus fraction of data load for MSTDM ($\alpha = 0.0125$, $k = 3$).
Figure 2.13: Data delay versus fraction of data load for MSTDM ($\alpha = 0.05, k = 2$).
2.3 Summary

We have proposed a non-preemptive priority queueing model for evaluating the mean delay-throughput performance of the slotted non-persistent CSMA/CD protocol. This analytical approach has also served as a stepping stone for formulating a similar approximate model for the MSTDM protocol. The analysis here is the first piece of work done on the data delay performance of MSTDM and yields explicit results for the mean data delay-throughput characteristics. For each analysis of the two protocols, numerical results have been presented and discussed. As comparisons with simulation results show, the accuracy of the analysis is high although it is based on a simple approximate model.
Chapter 3

Simulation of MSTDM

The analysis in the previous chapter provides a method for evaluating the mean delay-throughput performance of data packets for MSTDM. However, it does not give the higher moments of data delay as well as any information on voice delay characteristics. To gain a deeper understanding on the characteristics of MSTDM, simulation was done extensively to evaluate the delay performance of voice and data for this protocol. In this chapter, we first present the simulation model for the MSTDM protocol which is described in Section 2.2.1. Then we show the simulation results and discuss the delay characteristics of both voice and data. Moreover, we find that if we give the first voice packet generated at the beginning of a talk-spurt a shorter retransmission delay than data packet, we can considerably reduce the channel-acquisition delay for voice sources without sacrificing the data delay performance significantly.
3.1 Description of the Simulation Model

In our simulation model for the MSTDM protocol, we assume there is an infinite population of data sources which collectively form an independent Poisson source. The number of voice sources is, on the other hand, finite and we assume all of them are engaged in talking activities. By varying the number of voice sources in the MSTDM simulator, we can change the level of the throughput of voice packets\(^1\).

The durations of the talkspurt and the silent period are assumed to be exponentially distributed \([27]\) with means 1 sec and 1.4 sec respectively. The period of transmission for voice sources is 28 msec. Therefore, first voice packets contribute to 2.8% of the total number of packets generated by voice sources. The coding rate for speech is 64 Kbps. Thus, the number of voice bits which arrive in 28 msec is 1792. This is the size of the data area of voice packet. Including the space required for preempt area, packet header and overflow area, we let the voice packet length be 2000 bits. The channel transmission rate in our system is assumed to be 10 Mbps and so the transmission time of a voice packet is 200 \(\mu\)sec. The number of voice bits which arrive in the interval of one voice packet transmission time is 12.8 bits. The size of the overflow area is, therefore, 13 bits. In this simulation, we assume data packets have a fixed length which is the same as that of first voice packets and voice packets.

As the slotted MSTDM protocol is considered here, we assume the time axis is divided into slots. The size of a slot is equal to the end-to-end channel propagation

\(^1\)If we set the number of voice sources to be zero, the MSTDM simulator will become a CSMA/CD one. We do this to run simulation for the CSMA/CD protocol. The results obtained are used for verifying the analytical results in Section 2.1.4.
delay. Since transmission can only be started at the beginning of a slot, the packet transmission time is assumed to be a multiple number of slots in our MSTDM simulator. Because of the effect of signal propagation, the channel will not be sensed idle right after a packet has been completely transmitted. We assume it takes the time of one slot to have this effect removed from the channel for each packet transmission. Moreover, when data packets and/or first voice packets have a collision, the time that it takes for the channel to recover from busy state to idle state is a multiple number of slots.

There are two parameters that specify the simulated system. The first is the normalized end-to-end channel propagation delay \( \alpha \) which is defined as the quotient of the end-to-end channel propagation delay divided by the packet transmission time. Another parameter is the normalized collision recovery time which is given by \( k\alpha \), where \( k \) is the number of slots required to recover from a collision. As data packets and first voice packets may experience collision in the vulnerable interval, they have to be retransmitted according to a retry scheme. In our simulation studies, we have attempted to apply the same as well as different retry schemes to data packets and first voice packets.

In the first case, we use the binary exponential backoff (BEB) algorithm to schedule retransmission for both of them. In our BEB, if a packet has just experienced its \( C \)th collision, it will be transmitted at the point \( R \) slots from the time the source senses the channel idle, where \( R \) is picked up uniformly from the interval \( \{0, ..., 2^i - 1\} \) slots and \( i = \min\{C, 8\} \). If the channel is busy at the time of retransmission, the rescheduling procedure above will be repeated at the same interval for the selection of retransmission time. Note that if a packet has suffered eight collisions or
more, the retransmission interval will still remain at 255 slots until it is successfully transmitted.

In the second case, we use the BEB algorithm for data packets and the linear incremental backoff (LIB) algorithm [28] for first voice packets. LIB implies that the retransmission time is picked up uniformly from the interval \( \{0, \ldots, C\} \) slots, where \( C \) is the number of collisions a packet has suffered. For other details, LIB is the same as BEB. But we do not put an upper limit on the retransmission interval for LIB. We notice that the retransmission delay in LIB should be much shorter than that in BEB on the average.

The MSTDM simulator is implemented in the high-level language ‘C’. Our prime objective of running the simulator is to evaluate the delay performances of the three types of packets. We have run the simulator for two pairs of system parameters. The first pair are \( \alpha = 0.0125 \) and \( k = 3 \) whereas another pair are \( \alpha = 0.05 \) and \( k = 2 \). If the speed of signal propagation in the channel is assumed to be two-third of that of light, the first \( \alpha \) implies a slot size of 2.5 \( \mu \)sec which corresponds to a channel length of 500 m and the other is 10 \( \mu \)sec, a channel length of 2 Km. For each of the two simulated systems, we have applied the same as well as different retry schemes to data packets and first voice packets as described above. By comparing these two cases of retransmission scheduling, we can find out the effects on the delays of first voice packets and data packets if the former is given a shorter retransmission delay than the latter.

A time span of about five minutes in the real MSTDM system is simulated for each run. We believe that this is sufficient for obtaining reasonable statistical results.
CHAPTER 3. SIMULATION OF MSTDM

for our performance evaluation. The simulation results obtained verify this claim as they show definite trends consistently.

3.2 Simulation Results and Discussions

In the following, when data packets and first voice packets adopt the BEB algorithm for retransmission, the word 'data' is used to include both of them. This is because if the former and the latter have the same retransmission delay distribution, they will behave without any difference from each other in the system. However, when data packets use BEB and first voice packets use LIB, we have to mention them distinctly. In the simulation, the delay performances of the three types of packets were respectively obtained in terms of the mean and the standard deviation of their delay. The packet delay in the following figures is in the unit of packet transmission time and the throughput is in the unit of packet per packet transmission time.

Figure 3.1 through Figure 3.4 show the simulation results for the two sets of system parameters when both data packets and first voice packets use the BEB algorithm. The mean and the standard deviation of packet delay versus the data throughput are plotted at constant voice throughput ($\rho_v$) for data packets and voice packets in each simulated system.\footnote{The mean data delay of the simulation results obtained in this case has been used to verify the analysis on the performance of MSTDM in Chapter 2.}

From Figure 3.1 and Figure 3.3, we see that the data delay performance declines apparently when the voice throughput $\rho_v$ in the channel increases. This is obvious because voice packets occupy part of the channel capacity for transmission and
(a) Mean data delay

(b) Standard deviation of data delay

Figure 3.1: Delay characteristics of data packets when both first voice packets and data packets use BEB for retransmission ($\alpha = 0.0125$, $k = 3$)
CHAPTER 3. SIMULATION OF MSTDM

Data throughput

\( \text{Mean voice delay} \)

\[ \begin{align*}
\text{So} &= p - 0.044 \\
\text{Or} &= p - 0.147 \\
\text{O} &= p - 0.293 \\
\text{O} &= p - 0.437 \\
\text{O} &= p - 0.585 \\
\text{O} &= p - 0.738
\end{align*} \]

(a) Mean voice delay

\( \text{Standard deviation of voice delay} \)

\[ \begin{align*}
\text{SD} &= p - 0.044 \\
\text{SD} &= p - 0.147 \\
\text{SD} &= p - 0.293 \\
\text{SD} &= p - 0.437 \\
\text{SD} &= p - 0.585 \\
\text{SD} &= p - 0.738
\end{align*} \]

(b) Standard deviation of voice delay

Figure 3.2: Delay characteristics of voice packets when both first voice packets and data packets use BEB for retransmission \((\alpha = 0.0125, k = 3)\)
Figure 3.3: Delay characteristics of data packets when both first voice packets and data packets use BEB for retransmission (α = 0.05, k = 2)
Figure 3.4: Delay characteristics of voice packets when both first voice packets and data packets use BEB for retransmission ($\alpha = 0.05, k = 2$)
have a higher priority of transmission than data packets. With a higher voice throughput, the maximum of the data throughput that can be achieved is reduced relatively. From light to medium load of data traffic (relative to the maximum data throughput achievable), the mean and the standard deviation of data delay are quite small except at heavy load of voice traffic. However, they increases rapidly when the data throughput is high. This is a well-known characteristic for the performance of data packets using random access protocol for transmission.

Comparing the curves in Figure 3.1 and Figure 3.3, we find that the delay performance is better for $\alpha = 0.0125$ and $k = 3$ than for $\alpha = 0.05$ and $k = 2$. This is because collisions of data packets and removals of signal propagation after packet transmissions waste a more significant portion of the channel capacity in the latter case. Note that when the voice throughput is zero, the curves in Figure 3.1 and Figure 3.3 show the data delay performance of the slotted non-persistent CSMA/CD protocol.

In Figure 3.2 and Figure 3.4, we show the voice delay performance. We see that the mean delay of voice packets increases almost linearly with the data throughput when we keep the voice throughput constant. At the same channel throughput (sum of the throughputs of voice and data), the voice delay is much smaller when the channel is dominated by voice load than by data load. The reason is that voice packet transmissions do not interfere with each other as they are periodic and separable. On the other hand, data packet transmissions are bursty and so they may interfere with the periodic transmissions of voice sources. Therefore, the overflow area in voice packets is more utilized when the data throughput becomes higher. Yet we observe from the results that the mean waiting time of voice packets
is well below one packet transmission time in all cases of our simulation. For the standard deviation of voice delay, we find that it increases rapidly with the data throughput at the beginning, but its characteristics become leveled or even dropping slightly at high data throughput. This is in sharp contrast to the standard deviation characteristics of data delay. It is because data delay is unbounded while there is an upper limit of one packet transmission time on voice packet waiting time.

When BEB is used for data packets and LIB for first voice packets, the simulation results for the two sets of system parameters are shown from Figure 3.5 to Figure 3.10. In this case, we have separate figures for data packets, first voice packets and voice packets. The characteristics of the packet delay versus the throughput of data plus first voice packets are plotted.

Comparing the curves in Figure 3.5 and Figure 3.8 with the ones in Figure 3.1 and Figure 3.3, we see that the data delay performance is nearly unaffected when first voice packets are given a shorter retransmission delay than data packets, except at very high voice throughput. The reason for this is that the load of first voice packets is light in the channel as there is only one first voice packet in each talkspurt. They represent 2.8% of the total number of packets generated by voice sources in the simulation. Therefore, as comparison with the previous case shows, giving first voice packets a faster retransmission does not deteriorate the data delay performance significantly unless the channel is highly utilized by voice sources, i.e. the situation where the channel is congested and the load of first voice packets is larger than or comparable to that of data packets.

However, by doing so, we can improve the delay performance of first voice pack-
CHAPTER 3. SIMULATION OF MSTDM

Figure 3.5: Delay characteristics of data packets when first voice packets use LIB and data packets use BEB for retransmission ($\alpha = 0.0125$, $k = 3$)
(b) Standard deviation of first voice packet delay

Figure 3.6: Delay characteristics of first voice packets when first voice packets use LIB and data packets use BEB for retransmission ($\alpha = 0.0125$, $k = 3$)
CHAPTER 3. SIMULATION OF MSTDM

Figure 3.7: Delay characteristics of voice packets when first voice packets use LIB and data packets use BEB for retransmission ($\alpha = 0.0125$, $k = 3$)
CHAPTER 3. SIMULATION OF MSTDM

Figure 3.8: Delay characteristics of data packets when first voice packets use LIB and data packets use BEB for retransmission ($\alpha = 0.05, k = 2$)
Figure 3.9: Delay characteristics of first voice packets when first voice packets use LIB and data packets use BEB for retransmission ($\alpha = 0.05$, $k = 2$)
(a) Mean delay of voice packets

(b) Standard deviation of voice packet delay

Figure 3.10: Delay characteristics of voice packets when first voice packets use LIB and data packets use BEB for retransmission ($\alpha = 0.05$, $k = 2$)
ets considerably. This can be seen by comparing Figure 3.6 and Figure 3.9 with Figure 3.1 and Figure 3.3. Note that the data delay in Figure 3.1 and Figure 3.3 also refers to the delay of first voice packets as mentioned at the beginning of this section. The comparisons show that the delay of first voice packets grows more slowly and more steadily with the increase of data load in the channel when they have a shorter retransmission delay than data packets. Therefore, although both first voice packets and data packets use CSMA/CD, the former can be transmitted in a shorter time than the latter on the average. This gain is more remarkable when data load occupies a considerable portion of the channel capacity.

We know from the protocol description that the first voice packet needs to be transmitted within one period of voice transmission if the speech at the beginning of a talkspurt is not to be clipped. In our MSTDM simulator, the packet transmission time is 200 $\mu$/sec (at a channel transmission rate of 10 Mbps) and the period of voice transmission is 28 msec. From the simulation results, we see that the first voice packet delay is just in the order of millisecond even at high channel throughput when both data packets and first voice packets adopt the BEB algorithm for retransmission. This delay should be well below 28 msec most of the time. Therefore, it may not be needed to give a faster retransmission for first voice packets although it can still help minimize the chance of voice bits being discarded without affecting the data delay performance significantly.

However, if the channel transmission rate is lowered to 1 Mbps, the packet transmission time will become 2 msec. Then the first voice packet delay just mentioned above will correspondingly increase by 10 times going from the order of millisecond to ten milliseconds. But it can be considerably decreased if first voice packets use
LIB and data packets use BEB. In this case, it is, therefore, very favourable to have a shorter retransmission delay for first voice packets in order to reduce the channel-acquisition time for voice sources. This can prevent the speech from being clipped with only little adverse effect on the data delay performance.

Finally, we notice that the delay characteristics of voice packets in Figure 3.7 and Figure 3.10 are almost identical to the ones in Figure 3.2 and Figure 3.4. This shows that their performance remains unchanged no matter first voice packets and data packets have the same or different retry schemes.

### 3.3 Summary

We have presented the performance evaluation of the MSTDM protocol using simulation. The effects of this protocol on the delay performances of data packets, first voice packets and subsequent voice packets have been shown and discussed respectively in terms of the mean and the standard deviation of their delay. Moreover, we find that if first voice packets are given a shorter retransmission delay than data packets, the channel-acquisition delay for voice sources can be considerably reduced without sacrificing the data delay performance significantly.
Chapter 4

MSTDM Network for a Metropolitan Area

The MSTDM protocol supports both voice and data traffic in a local area network. However, its efficiency decreases inversely with the transmission rate and the channel length because of its bus contention nature. In this chapter, a scheme which alleviates the distance and transmission rate constraints associated with this protocol is introduced [16]. The approach is to divide a network spanning a large area into a number of sub-networks called homenets, for improving the MSTDM protocol efficiency. Then we propose an access protocol for the interconnection of homenets. It facilitates the communications between stations in different homenets for both voice and data services. Finally, we find the optimum number of homenets, into which a large area should be partitioned, to minimize the mean delay of data packets within homenet.
CHAPTER 4. MSTDM NETWORK FOR A METROPOLITAN AREA

4.1 Homenet Network

As we see in the previous two chapters, the performance of the MSTDM protocol deteriorates with the increase of \( \alpha \) which is defined as the ratio of the end-to-end channel propagation delay to the packet transmission time. The protocol does not even work if \( \alpha \) is greater than one. These characteristics are, in general, valid for any bus contention protocol with carrier sensing property [13]. In an MSTDM system, the preempt area of voice packets does not carry any information. Its length depends on the maximum time that it takes for data sources and voice sources transmitting first voice packets to recover from a collision. Since the maximum collision recovery time increases with the end-to-end propagation delay, the fraction of the voice packet, which carries useful data, is small when \( \alpha \) is large. As a result, the efficiency of the MSTDM protocol will be low if it is directly applied to a network which spans a large area and has a high transmission rate.

In order to solve this problem, we have to alleviate the distance and transmission rate constraints associated with the MSTDM protocol. The homenet approach presented here is to divide the stations in a network covering a large area, for example a metropolitan area, into regional groups so that the end-to-end channel propagation delay applied to each region is reduced [16][29][30]. The communications channel with a high total bit rate is also divided, and a fraction of the capacity is assigned to each regional group. Stations in a region only transmit in the fraction of the capacity assigned to their group. This decreases the transmission rate of the stations, compared with the case when the total channel capacity is available to them. The overall effect is that the ratio of the end-to-end channel propagation delay in each
region to the packet transmission time is considerably decreased.

Each group forms a local area network called homenet, using the MSTDM protocol. The physical medium can be a conventional subsplit cable system, for example a CATV network which can be found under the ground of many metropolitan areas [31]. The transmission of signal is uni-directional in broadband. Figure 4.1 shows a network with three homenets on a subsplit cable system. The stations in each homenet transmit in frequency band $f_0$ in upstream. There is a local reflection point in each homenet. At the reflection point, the upstream frequency band $f_0$ is prevented from propagating to adjacent homenet by means of notch filter. It is also translated into the downstream frequency band $f_n$ in homenet $n$, where $n$ is an index for one of the homenets in the network. Each station has one frequency agile receiver (or more). The stations in homenet $n$ receive packets for communications within homenet by listening to $f_n$, which is their homenet frequency. Usually, a station listens to its homenet frequency all the time unless it is informed to switch to another frequency. Moreover, the upstream frequency band $f_0$ is translated into frequency band $f_{Dn}$ at the reflection point of homenet $n$, and $f_{Dn}$ propagates towards the head end of the whole network, which is the root of a subsplit cable system. At the head end\(^1\), $f_{Dn}$ is converted to $f_n$ which is carried throughout the network.

\(^1\)Besides the frequency translation operations, a variety of user services can be incorporated in the head end. Analog video can be broadcasted from the head end to every station in the network in a different frequency band like a CATV system. A head-end station identical to the ones used at the user nodes can be implemented to perform the functions of an interactive data base providing different kinds of information to its customers. Another head-end station can act as a switching processor for establishing voice links with sources outside the network. A user sends a data packet to this processor giving the number to be called. It then processes the call and connects the user's voice path to the outside line. Similarly, a station at the head end can function as a gate-way which is responsible for data communications with the outside networks. If necessary, the stations providing services at the head end can have receivers listening to all homenet frequencies at the same time and can form a homenet of their own.
two Notch filter bands $f_0$ and $f_2$

$\bigcirc$ local reflection point

$\bullet$ head end

Figure 4.1: Homenet network on a subsplit cable system.
except homenet \( n \) where the \( f_n \) transmitted from the head end is notch filtered at the reflection point to prevent collision with the \( f_n \) localized within homenet \( n \).

Through this configuration, the connectivity between stations in the network is complete. Once a packet has been successfully transmitted in a homenet, it can be received anywhere in the whole network. In this way, a station in homenet \( k \), for example, can receive packets sent from homenet \( l \) by listening to frequency band \( f_l \), instead of its homenet frequency. Therefore, the link for inter-homenet packet transfer is set up. This homenet approach restricts the contention for channel acquisition to be localized among stations within homenet, no matter the transmission is for intra- or inter-homenet packet transfer. The result is a significant improvement in protocol efficiency.

### 4.2 Description of the Access Protocol

A station has to know to which frequency it should listen for getting an inter-homenet packet addressed to it. In this section, we propose an access protocol for inter-homenet communications. This access protocol makes use of a monitor which is implemented in each homenet and basically performs the function of a store-and-forward node.

#### 4.2.1 Interconnection

The monitor in each homenet has receivers listening to all homenet frequencies at the same time. It accepts those packets which are sent from other homenets to the
stations in its homenet. It keeps the packets unchanged and delivers them to their destinations as a store-and-forward node.

For example, station $A_1$ in homenet 1 wants to send a data packet to station $B_2$ in homenet 2. When the transmission is successful, the monitor ($M_2$) in homenet 2 detects this packet in frequency band $f_1$. Recognizing the destination address belongs to one of the stations within its homenet, $M_2$ receives the packet and stores it in the buffer. Then $M_2$ transmits the packet to $B_2$ following the MSTDM protocol in the same way as other stations in homenet 2. Finally $B_2$ receives the packet in frequency band $f_2$, its homenet frequency. The operation of this packet transfer is depicted in Figure 4.2.

This access protocol is suitable for handling bursty data traffic for inter-homenet communications. Yet, for voice packet transmission, we can eliminate the monitor
from being involved after the initial set-up as shown in Figure 4.3. We use the communications between $A_1$ and $B_2$ as an example again. Now $A_1$ wants to have a conversation with $B_2$. So $A_1$ sends a call-connection request packet to $B_2$. This packet is carried through the network to $B_2$ by the method just mentioned. Assuming $B_2$ accepts the connection, it sends a call-confirmation packet back to $A_1$ and its receiver is switched to listen to $f_1$. After receiving the confirmation, $A_1$ switches to listen to $f_2$.

Now $A_1$ and $B_2$ receive voice packets in another's homenet frequency as if they were resided in another's homenet\(^2\). Through the setting of an indication bit in the packet header, the monitors in both homenet 1 and homenet 2 know that the packets sending between $A_1$ and $B_2$ are received by them directly. Therefore, the monitors do not accept these packets. The connection remains throughout the conversation until one side disconnects the link by a clear-connection request. Then $A_1$ and $B_2$ return to their homenet frequencies. In this way, no additional delay is created on top of that due to the MSTDM protocol for voice packet transmission. Thus, MSTDM works well even for inter-homenet voice traffic. This kind of connection between stations in different homenets is also suitable for long file transfer, which can avoid the extra data delay caused when the monitor is involved for every single packet transfer. However, the connection is likely to be one-way in this case. If a station wants to have additional connections simultaneously, it has to be equipped with more than one receiver.

\(^2\)If $A_1$ wants to talk to somebody outside the network, $B_2$ will be one of the functional units of the head-end switching processor described before. These functional units are switching nodes which process the calls and connect the users' voice paths to the outside lines. In this example, the outgoing signal from $A_1$ is carried in frequency band $f_1$ to the switching node $B_2$, and the incoming signal from the outside world is transmitted from $B_2$ to $A_1$ in frequency band $f_2$. At the initial set-up, the number to be called is included in $A_1$'s call-connection request packet.
Figure 4.3: Link set-up for conversation between two stations in different homenets.
4.2.2 Other Related Issues

Inter-homenet packets resulted from bursty data traffic pass through the monitors before reaching their destinations. We may assign priorities to different types of inter-homenet data packets so that the one with highest priority is served by the monitor sooner than the others, instead of being first-come-first-serve. Priority can be made according to the need of a packet for faster response or a toll scheme. This can make the network more adaptable to the users' need.

As a monitor listens to all homenet frequencies, it can observe all of the channel's traffic, construct traffic matrices and perform load analysis to measure the utilization of each homenet. It can then determine which homenet can support additional connections under certain constraint of performance measure. If a connection cannot be made, for example between homenet 1 and homenet 2, because of an overload of input traffic in the former, the monitor in homenet 2 will transmit a clear-connection request stating the reason for rejection to the sending station in homenet 1, after receiving its call-connection request packet.

Addressing is also an important issue in defining the access protocol [1]. We need an efficient way of managing station addresses as machines move and new homenets join in the network. The use of absolute station addresses, each of which is unique over the whole network, in this way, is better than homenet specific station addresses. Two problems arise when the latter addressing method is adopted. The first is that when a station is moved from one homenet to another, it may have to change its address if another station there is using the same one. Moreover, a homenet specific address has to be combined with a unique network address so
that the whole address is unambiguous over the network space for inter-homenet communications. However, the absolute addressing method solves these problems, provided that the address space used is large enough to ensure uniqueness and give adequate room for growth.

To support multicasting communications, we require another addressing mode which makes use of logical station addresses. A logical address can be shared by a group of stations which may be located in more than one homenet. A packet sent with a logical address can, therefore, be received by more than one station. Also, a station with more than one logical address can receive packets from different logical groups of stations. This kind of service is very useful in many distributed applications, such as tele-conferencing and the access and update of distributed data bases. In order to distinguish whether the address is absolute or logical, a specific indication bit has to be included in the packet header. To support the above two types of addressing mode for inter-homenet communications, a monitor has to maintain two lists of addresses for the stations in its homenet, one for absolute addressing and another for logical addressing.

4.3 Optimum Number of Homenets

4.3.1 Analysis

Suppose we have a subsplit cable system with certain channel capacity. If we divide it into a number of homenets and each homenet occupies part of the total channel capacity, the efficiency of the MSTDM protocol increases because (i) the end-to-
end propagation delay within homenet decreases, and (ii) the packet transmission time increases. The result is that we reduce the mean delay of data packets for transmission within homenet, which is normalized w.r.t. the packet transmission time. However, as the system is divided, it takes longer actual time to transmit a packet because only part of the total channel capacity is available to each homenet. Therefore, there is an optimum number of homenets the system should be divided into for networking, in order to minimize the mean data delay within homenet.

Here we assume the stations are uniformly distributed and they are equally divided into $N$ homenets. The area of the resultant homenets is inversely proportional to $N$. It is assumed that the resultant homenets are roughly circles or squares so that the distance between the two ends of the transmission media in a homenet is inversely proportional to $N^{1/2}$. Thus, the end-to-end propagation delay within homenet is reduced by $1/N^{1/2}$, compared with an undivided system. But the division increases the packet transmission time by $N$.

We assume the packet lengths for voice and data are constant and equal in the following. Let $\alpha_N$ be the normalized end-to-end propagation delay w.r.t. the packet transmission time in a homenet. Then $\alpha_N = \alpha/N^{3/2}$ where $\alpha$ is the propagation delay between the two ends of the undivided system, normalized w.r.t. the packet transmission time in that system. In Chapter 2, we obtain analytically the mean delay ($D_n$) of data packets for the slotted MSTDM protocol with non-persistent data packet transmission. Applying the results in Section 2.2.3, we find the mean data delay for transmission within homenet, which is normalized w.r.t. the packet
transmission time in the undivided system. It is given by

\[ D_n = N \left(1 + \overline{W}_{3n} + \frac{\alpha}{2N^{3/2}}\right) \]  

(4.1)

where

\[ \overline{W}_{1n} = \frac{\beta_n + \sqrt{\beta_n^2 - 4\gamma_n}}{2} \]  

(4.2)

\[ \beta_n = \rho_d \left[ \frac{k^2 \alpha^2 (\nu - 1)}{N^3 (1 - e^{-k\alpha / N^{3/2}})} + \frac{(1 + \alpha/N^{3/2})^2}{1 - e^{-(1+\alpha/N^{3/2})\rho_v}} \right] \]

\[ - \frac{1}{\rho_v} \left(1 + \rho_d \left[1 + \frac{\alpha}{N^{3/2}} + \frac{k\alpha}{N^{3/2}} (\nu - 1)\right] \right) \]  

(4.3)

\[ \gamma_n = \frac{1}{\rho_v} \left\{1 + \frac{\alpha}{N^{3/2}} + \frac{\rho_d}{\rho_v} \left[1 + \frac{\alpha}{N^{3/2}} + \frac{k\alpha}{N^{3/2}} (\nu - 1)\right] \right\} \]

\[ - \left\{ \frac{(1 + \alpha/N^{3/2})^2}{1 - e^{-(1+\alpha/N^{3/2})\rho_v}} + \frac{\rho_d}{\rho_v} \left[ \frac{k^2 \alpha^2 (\nu - 1)}{N^3 (1 - e^{-k\alpha / N^{3/2}})} + \frac{(1 + \alpha/N^{3/2})^2}{1 - e^{-(1+\alpha/N^{3/2})\rho_v}} \right] \right\} \]  

(4.4)

\[ \overline{W}_{2n} = \frac{\Phi_n + \hat{\rho}_v \overline{W}_{1n} (1 + \alpha/N^{3/2} + \Phi_n)}{(1 + \hat{\rho}_v \overline{W}_{1n})[1 - \rho_d k\alpha (\nu - 1)/N^{3/2}] - \hat{\rho}_v (1 + \alpha/N^{3/2})} \]  

(4.5)

\[ \Phi_n = \frac{1}{2} \left\{ \frac{\hat{\rho}_v (1 + \alpha/N^{3/2})^2}{1 + \hat{\rho}_v \overline{W}_{1n}} + \rho_d \left[1 + \alpha/N^{3/2}\right] + \frac{k^2 \alpha^2}{N^{3/2}} (\nu - 1) \right\} \]  

(4.6)

\[ \overline{W}_{3n} = \frac{\Phi_n + \hat{\rho}_v \overline{W}_{1n} (1 + \alpha/N^{3/2} + \Phi_n) + \rho_d k\alpha (\nu - 1) \overline{W}_{2n} (1 + \hat{\rho}_v \overline{W}_{1n}) / N^{3/2}}{(1 + \hat{\rho}_v \overline{W}_{1n})[1 - \rho_d (1 + \alpha/N^{3/2} + k\alpha (\nu - 1)/N^{3/2})] - \hat{\rho}_v (1 + \alpha/N^{3/2})} \]  

(4.7)

Recall that in the equations above,

a. \( \rho_d \) is the data utilization in a homenet;

b. \( \hat{\rho}_v / (1 + \hat{\rho}_v \overline{W}_{1n}) \) is the voice utilization (\( \rho_v \)) in the same homenet;
c. $k$ is the number of the end-to-end propagation delay within the homenet, which is required to recover from a collision of data packets and/or first voice packets; and

d. $1/\nu$ is the probability of successful transmission for data packets and first voice packets.

4.3.2 Numerical Results

Using (4.1), we plot $D_n$ versus $N$. Figure 4.4 through Figure 4.6 show this relationship at different data utilizations $\rho_d$, but we keep the voice utilization $\rho_v$ to be constant at 0, 0.2 and 0.4 of the channel capacity of the homenet in the figures respectively. The values of $\alpha$, $k$ and $\nu$ used are, respectively, 0.9, 2 and $e$. Note that when $\rho_v$ is zero, there is only data traffic in the homenet network and the MSTDM protocol degenerates to CSMA/CD. We see from the figures that as the number of homenets increases, the mean data delay decreases because of the improvement in protocol efficiency. Then it reaches a minimum and starts to increase. The increase occurs because the time it actually takes to transmit a packet increases with the number of homenets. Therefore, there is an optimum number of partitions the system should be divided into for minimizing the mean delay of data packets within homenet. In an actual system, the optimum number of homenets depends on the geometry of the system and the designed utilizations of both voice and data in the homenets.
Figure 4.4: Mean data delay for transmission within homenet as a function of the number of homenets at $\rho_v = 0$. 
CHAPTER 4. MSTDM NETWORK FOR A METROPOLITAN AREA

Figure 4.5: Mean data delay for transmission within homenet as a function of the number of homenets at $\rho_v = 0.2$. 
Figure 4.6: Mean data delay for transmission within homenet as a function of the number of homenets at $\rho_v = 0.4$. 
4.4 Summary

We have introduced the homenet approach for networking in a metropolitan area, using the MSTDM protocol. Also we have proposed an access protocol for the interconnection of homenets and have discussed some of the related issues. Based on the analytical results obtained for the performance of MSTDM in Chapter 2, we find the optimum number of homenets for interconnection, which minimizes the mean delay of data packets within homenet.
Chapter 5

Conclusions

This thesis mainly contains the research work in two areas: performance evaluation of the MSTDM protocol and its application in metropolitan area networks. A minor part is given to the performance analysis of the CSMA/CD protocol.

Two methods are taken for the performance evaluation. They are mathematical analysis and simulation. First, we propose the M/G/1 model with two non-preemptive priorities for the slotted non-persistent CSMA/CD protocol. This analytical approach presents a simpler way of performance evaluation than the previous works in the literature, and yields explicit results for the mean delay-throughput characteristics.

Because MSTDM is a variation on CSMA/CD, we are able to formulate a similar analytical model for the former by extending the previous M/G/1 system to three non-preemptive priorities. One point, which makes the extended model differ from ordinary priority queueing systems, is that the highest priority queue has only one buffer. We solve the model and obtain explicit results for the mean data delay-
throughput performance of the slotted non-persistent MSTDM protocol. This is the first piece of work done on the data delay performance of MSTDM in the literature.

For each analysis of the two protocols, numerical examples are shown, and their delay characteristics are discussed and compared with simulation results. As the comparisons show, the analysis yields very good approximate results. The shortcoming of this analysis is that it is only good for the mean of data delay. To find the general characteristics of the data delay and voice delay of the MSTDM protocol and its stability, further research needs to be performed.

Simulation provides a promising way of performance evaluation when the tractability of the mathematical analysis cannot be reached. We ran simulations for MSTDM, and the delay characteristics of data packets, first voice packets and subsequent voice packets are obtained in terms of the mean and the standard deviation of their delay. These simulation results enrich our understanding on the performance of this protocol, and they are described and discussed in detail in the thesis.

In the simulation, the same as well as different retransmission scheduling algorithms are applied for data packets and first voice packets. In the first case, both of them use the binary exponential backoff algorithm for retransmission; and in the second case, the former uses this algorithm while the latter uses the linear incremental backoff algorithm. Comparing the simulation results from these two cases, we find that if first voice packets are given a shorter retransmission delay than data packets, the channel-acquisition delay for voice sources can be considerably reduced without sacrificing the data delay performance significantly.
Because of its contention nature, the performance of the MSTDM protocol in a network deteriorates when the channel length and transmission rate of the network increase. In order to make this protocol applicable to metropolitan area networks, we have to find a suitable configuration which can alleviate the distance and transmission rate constraints associated.

In the thesis, the homenet approach is introduced for solving this problem. An access protocol for the interconnection of homenets is proposed, which facilitates the communications between stations in different homenets for both voice and data services. The issues about traffic control and addressing are also mentioned. Given the criterion to minimize the mean data delay for transmission within homenet, we perform an analysis and find the optimum number of homenets for interconnection.
Appendix A

In this appendix, we show the analysis that derives (2.1) for the non-preemptive priority queueing model shown in Figure 2.2. The analysis makes use of a two-dimensional Markov chain [32]. A more simplified approach based on expectation can be found, for example, in [26].

A Markov chain is imbedded at transmission completion epoch. Since there are 2 distinct classes of packets, we have to distinguish among the imbedded points as to which class completed service. This is indicated by the term $k$ class epoch, where $k = 1$ or 2.

Let $n_{ik}$ be the number of class $k$ packets in the system at the $i$th departure epoch and $a_{jk}$ be the number of class $j$ packets arriving during the transmission of a class $k$ packet. The $(i + 1)$st departure epoch is class 1 for $n_{i1} > 0$, and

\[
\begin{align*}
n_{i+1,1} &= n_{i1} - 1 + a_{11} \\
n_{i+1,2} &= n_{i2} + a_{21}.
\end{align*}
\] (A.1)

The $(i + 1)$st departure epoch is class 2 for $n_{i2} > 0$ when the $i$th departure leaves
the system devoid of class 1 packet. Then

\[ n_{i+1,1} = a_{12} \]
\[ n_{i+1,2} = n_{i2} - 1 + a_{22} . \]  \hspace{1cm} (A.2)

When the ith departure leaves the system empty, with probability \( \lambda_1/\lambda \),

\[ n_{i+1,1} = a_{11} \]
\[ n_{i+1,2} = a_{21} \] \hspace{1cm} (A.3)

where \( \lambda = \lambda_1 + \lambda_2 \), and with probability \( \lambda_2/\lambda \),

\[ n_{i+1,1} = a_{12} \]
\[ n_{i+1,2} = a_{22} \] \hspace{1cm} (A.4)

Define

\[ \hat{\rho} = \lambda_1 m_1 + \lambda_2 m_2 . \] \hspace{1cm} (A.5)

Then for the following 3 disjoint events in the queue dynamics, we have

\[
\begin{align*}
\text{Prob} [n_{i1} = 0 \text{ and } n_{i2} = 0] &= 1 - \hat{\rho} = P_0 \\
\text{Prob} [n_{i1} > 0] &= \lambda_1 \hat{\rho}/\lambda = P_1 \\
\text{Prob} [n_{i1} = 0 \text{ and } n_{i2} > 0] &= \lambda_2 \hat{\rho}/\lambda = P_2 .
\end{align*}
\] \hspace{1cm} (A.6)

Conditioning on the 3 events to calculate the two-dimensional probability generating functions of \( n_{i+1,1} \) and \( n_{i+1,2} \), we have

\[
E [Z_1^{n_{i+1,1}} Z_2^{n_{i+1,2}}] =
\begin{align*}
P_0 \{ (\lambda_1/\lambda)E [Z_1^{a_{11}} Z_2^{a_{21}} | n_{i1} = n_{i2} = 0] + \\
(\lambda_2/\lambda)E [Z_1^{a_{12}} Z_2^{a_{22}} | n_{i1} = n_{i2} = 0] \} +
\end{align*}
\]
\[ P_1 E[Z_1^{n_{i1} - 1 + a_{11}} Z_2^{n_{i2} + a_{21}} | n_{i1} > 0] + \]
\[ P_2 E[Z_1^{a_{12}} Z_2^{n_{i2} - 1 + a_{22}} | n_{i1} = 0, n_{i2} > 0] . \] (A.7)

Solving (A.7) by differentiating w.r.t. \( Z_1 \) and \( Z_2 \), and after some manipulations, we obtain

\[ \bar{n}_{22} = 1 + \frac{\lambda_2(\lambda_1 \bar{m}_1^2 + \lambda_2 \bar{m}_2^2)}{2 \hat{\rho}(1 - \lambda_1 \bar{m}_1)(1 - \lambda_1 \bar{m}_1 - \lambda_2 \bar{m}_2)} \] (A.8)

where \( \bar{n}_{22} \) is the mean number of class 2 packets in the system given that one of them is beginning transmission. The average number of class 2 packets which arrive during the waiting time of the one being transmitted is \( \bar{n}_{22} - 1 \). Since packets arriving to an empty system suffer no queueing delay, the mean waiting time \( (\bar{W}_2) \) of class 2 packets is given by

\[ \bar{W}_2 = (\hat{\rho} / \lambda_2)(\bar{n}_{22} - 1) \quad \text{by Little's Formula.} \] (A.9)

Substituting (A.8) into (A.9), we obtain

\[ \bar{W}_2 = \frac{\lambda_1 \bar{m}_1^2 + \lambda_2 \bar{m}_2^2}{2(1 - \lambda_1 \bar{m}_1)(1 - \lambda_1 \bar{m}_1 - \lambda_2 \bar{m}_2)} \] (A.10)

which is (2.1).
Appendix B

B.1 Tables of the CSMA/CD Simulation Results

Table B.1: 95% Confidence intervals for the CSMA/CD simulation results shown in Figure 2.3 \((k = 2)\).

(a) \(\alpha = 0.001\)
(b) \(\alpha = 0.01\)
(c) \(\alpha = 0.05\)
(d) \(\alpha = 0.1\)
(e) \(\alpha = 0.2\)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.005</td>
<td>(1.004, 1.007)</td>
<td>1.006</td>
</tr>
<tr>
<td>0.101</td>
<td>1.059</td>
<td>(1.057, 1.061)</td>
<td>1.057</td>
</tr>
<tr>
<td>0.201</td>
<td>1.128</td>
<td>(1.124, 1.131)</td>
<td>1.127</td>
</tr>
<tr>
<td>0.300</td>
<td>1.214</td>
<td>(1.210, 1.218)</td>
<td>1.216</td>
</tr>
<tr>
<td>0.401</td>
<td>1.340</td>
<td>(1.331, 1.349)</td>
<td>1.337</td>
</tr>
<tr>
<td>0.500</td>
<td>1.496</td>
<td>(1.480, 1.512)</td>
<td>1.505</td>
</tr>
<tr>
<td>0.603</td>
<td>1.767</td>
<td>(1.746, 1.788)</td>
<td>1.768</td>
</tr>
<tr>
<td>0.702</td>
<td>2.195</td>
<td>(2.149, 2.241)</td>
<td>2.197</td>
</tr>
<tr>
<td>0.801</td>
<td>3.067</td>
<td>(3.021, 3.112)</td>
<td>3.062</td>
</tr>
<tr>
<td>0.851</td>
<td>3.896</td>
<td>(3.742, 4.049)</td>
<td>3.934</td>
</tr>
<tr>
<td>0.900</td>
<td>5.588</td>
<td>(5.388, 5.789)</td>
<td>5.719</td>
</tr>
<tr>
<td>0.911</td>
<td>6.314</td>
<td>(5.901, 6.727)</td>
<td>6.410</td>
</tr>
<tr>
<td>0.921</td>
<td>7.259</td>
<td>(6.868, 7.650)</td>
<td>7.206</td>
</tr>
<tr>
<td>0.928</td>
<td>7.872</td>
<td>(7.249, 8.495)</td>
<td>7.896</td>
</tr>
</tbody>
</table>

B.1 (a)
### APPENDIX B.

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.009</td>
<td>(1.009, 1.010)</td>
<td>1.010</td>
</tr>
<tr>
<td>0.100</td>
<td>1.064</td>
<td>(1.062, 1.066)</td>
<td>1.062</td>
</tr>
<tr>
<td>0.301</td>
<td>1.231</td>
<td>(1.226, 1.237)</td>
<td>1.231</td>
</tr>
<tr>
<td>0.501</td>
<td>1.553</td>
<td>(1.545, 1.561)</td>
<td>1.551</td>
</tr>
<tr>
<td>0.702</td>
<td>2.404</td>
<td>(2.367, 2.442)</td>
<td>2.381</td>
</tr>
<tr>
<td>0.801</td>
<td>3.620</td>
<td>(3.547, 3.692)</td>
<td>3.574</td>
</tr>
<tr>
<td>0.840</td>
<td>4.714</td>
<td>(4.392, 5.036)</td>
<td>4.588</td>
</tr>
<tr>
<td>0.862</td>
<td>5.575</td>
<td>(5.316, 5.834)</td>
<td>5.552</td>
</tr>
<tr>
<td>0.877</td>
<td>6.505</td>
<td>(6.172, 6.837)</td>
<td>6.512</td>
</tr>
<tr>
<td>0.899</td>
<td>9.158</td>
<td>(8.643, 9.673)</td>
<td>8.800</td>
</tr>
</tbody>
</table>

B.1 (b)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.030</td>
<td>(1.029, 1.030)</td>
<td>1.031</td>
</tr>
<tr>
<td>0.100</td>
<td>1.093</td>
<td>(1.091, 1.096)</td>
<td>1.090</td>
</tr>
<tr>
<td>0.300</td>
<td>1.313</td>
<td>(1.303, 1.322)</td>
<td>1.305</td>
</tr>
<tr>
<td>0.400</td>
<td>1.512</td>
<td>(1.507, 1.518)</td>
<td>1.495</td>
</tr>
<tr>
<td>0.501</td>
<td>1.838</td>
<td>(1.816, 1.860)</td>
<td>1.817</td>
</tr>
<tr>
<td>0.599</td>
<td>2.477</td>
<td>(2.428, 2.526)</td>
<td>2.420</td>
</tr>
<tr>
<td>0.701</td>
<td>4.214</td>
<td>(4.024, 4.403)</td>
<td>4.136</td>
</tr>
<tr>
<td>0.757</td>
<td>7.512</td>
<td>(7.155, 7.870)</td>
<td>7.493</td>
</tr>
</tbody>
</table>

B.1 (c)
### APPENDIX B.

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.055</td>
<td>(1.053, 1.057)</td>
<td>1.057</td>
</tr>
<tr>
<td>0.100</td>
<td>1.133</td>
<td>(1.130, 1.137)</td>
<td>1.128</td>
</tr>
<tr>
<td>0.300</td>
<td>1.445</td>
<td>(1.431, 1.459)</td>
<td>1.428</td>
</tr>
<tr>
<td>0.400</td>
<td>1.799</td>
<td>(1.776, 1.823)</td>
<td>1.753</td>
</tr>
<tr>
<td>0.502</td>
<td>2.503</td>
<td>(2.451, 2.556)</td>
<td>2.461</td>
</tr>
<tr>
<td>0.550</td>
<td>3.303</td>
<td>(3.196, 3.410)</td>
<td>3.150</td>
</tr>
<tr>
<td>0.570</td>
<td>3.682</td>
<td>(3.544, 3.821)</td>
<td>3.612</td>
</tr>
<tr>
<td>0.580</td>
<td>4.104</td>
<td>(3.937, 4.271)</td>
<td>3.908</td>
</tr>
<tr>
<td>0.601</td>
<td>4.828</td>
<td>(4.700, 4.956)</td>
<td>4.706</td>
</tr>
<tr>
<td>0.605</td>
<td>5.139</td>
<td>(4.804, 5.473)</td>
<td>4.904</td>
</tr>
<tr>
<td>0.614</td>
<td>5.589</td>
<td>(5.394, 5.785)</td>
<td>5.424</td>
</tr>
<tr>
<td>0.626</td>
<td>6.460</td>
<td>(6.012, 6.908)</td>
<td>6.358</td>
</tr>
</tbody>
</table>

#### B.1 (d)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.108</td>
<td>(1.105, 1.110)</td>
<td>1.109</td>
</tr>
<tr>
<td>0.100</td>
<td>1.227</td>
<td>(1.221, 1.233)</td>
<td>1.214</td>
</tr>
<tr>
<td>0.198</td>
<td>1.442</td>
<td>(1.430, 1.455)</td>
<td>1.415</td>
</tr>
<tr>
<td>0.300</td>
<td>1.877</td>
<td>(1.855, 1.900)</td>
<td>1.850</td>
</tr>
<tr>
<td>0.400</td>
<td>3.061</td>
<td>(3.000, 3.123)</td>
<td>3.030</td>
</tr>
<tr>
<td>0.421</td>
<td>3.592</td>
<td>(3.491, 3.694)</td>
<td>3.574</td>
</tr>
<tr>
<td>0.440</td>
<td>4.369</td>
<td>(4.185, 4.553)</td>
<td>4.300</td>
</tr>
<tr>
<td>0.461</td>
<td>5.800</td>
<td>(5.346, 6.253)</td>
<td>5.531</td>
</tr>
<tr>
<td>0.470</td>
<td>6.577</td>
<td>(6.250, 6.903)</td>
<td>6.361</td>
</tr>
<tr>
<td>0.480</td>
<td>7.735</td>
<td>(7.347, 8.123)</td>
<td>7.620</td>
</tr>
</tbody>
</table>

#### B.1 (e)
### B.2 Tables of the MSTDM Simulation Results

Table B.2: 95% Confidence intervals for the MSTDM simulation results shown in Figure 2.10 ($\alpha = 0.0125$, $k = 3$).

(a) $\rho_v = 0$
(b) $\rho_v = 0.044$
(c) $\rho_v = 0.147$
(d) $\rho_v = 0.293$
(e) $\rho_v = 0.437$
(f) $\rho_v = 0.585$
(g) $\rho_v = 0.738$

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.011</td>
<td>(1.010, 1.011)</td>
<td>1.011</td>
</tr>
<tr>
<td>0.100</td>
<td>1.066</td>
<td>(1.064, 1.068)</td>
<td>1.064</td>
</tr>
<tr>
<td>0.301</td>
<td>1.238</td>
<td>(1.233, 1.244)</td>
<td>1.239</td>
</tr>
<tr>
<td>0.501</td>
<td>1.586</td>
<td>(1.579, 1.593)</td>
<td>1.584</td>
</tr>
<tr>
<td>0.698</td>
<td>2.534</td>
<td>(2.480, 2.588)</td>
<td>2.518</td>
</tr>
<tr>
<td>0.801</td>
<td>4.160</td>
<td>(4.017, 4.302)</td>
<td>4.157</td>
</tr>
<tr>
<td>0.850</td>
<td>6.034</td>
<td>(5.799, 6.270)</td>
<td>6.439</td>
</tr>
<tr>
<td>0.867</td>
<td>7.578</td>
<td>(7.334, 7.822)</td>
<td>8.135</td>
</tr>
<tr>
<td>0.878</td>
<td>9.300</td>
<td>(8.367, 10.234)</td>
<td>9.855</td>
</tr>
</tbody>
</table>

B.2 (a)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.011</td>
<td>1.039</td>
<td>(1.036, 1.041)</td>
<td>1.038</td>
</tr>
<tr>
<td>0.101</td>
<td>1.101</td>
<td>(1.099, 1.102)</td>
<td>1.100</td>
</tr>
<tr>
<td>0.301</td>
<td>1.308</td>
<td>(1.303, 1.314)</td>
<td>1.306</td>
</tr>
<tr>
<td>0.501</td>
<td>1.738</td>
<td>(1.730, 1.746)</td>
<td>1.739</td>
</tr>
<tr>
<td>0.701</td>
<td>3.055</td>
<td>(2.999, 3.112)</td>
<td>3.100</td>
</tr>
<tr>
<td>0.801</td>
<td>5.785</td>
<td>(5.676, 5.893)</td>
<td>6.049</td>
</tr>
<tr>
<td>0.821</td>
<td>7.617</td>
<td>(7.376, 7.859)</td>
<td>8.045</td>
</tr>
<tr>
<td>0.831</td>
<td>8.861</td>
<td>(8.345, 9.377)</td>
<td>9.279</td>
</tr>
</tbody>
</table>

B.2 (b)
### B.2 (c)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.014</td>
<td>1.121</td>
<td>(1.116, 1.126)</td>
<td>1.119</td>
</tr>
<tr>
<td>0.104</td>
<td>1.218</td>
<td>(1.214, 1.222)</td>
<td>1.210</td>
</tr>
<tr>
<td>0.205</td>
<td>1.353</td>
<td>(1.349, 1.357)</td>
<td>1.343</td>
</tr>
<tr>
<td>0.305</td>
<td>1.537</td>
<td>(1.531, 1.542)</td>
<td>1.529</td>
</tr>
<tr>
<td>0.405</td>
<td>1.829</td>
<td>(1.813, 1.844)</td>
<td>1.827</td>
</tr>
<tr>
<td>0.505</td>
<td>2.292</td>
<td>(2.267, 2.316)</td>
<td>2.309</td>
</tr>
<tr>
<td>0.603</td>
<td>3.236</td>
<td>(3.200, 3.271)</td>
<td>3.313</td>
</tr>
<tr>
<td>0.705</td>
<td>6.395</td>
<td>(6.079, 6.711)</td>
<td>6.742</td>
</tr>
<tr>
<td>0.724</td>
<td>8.574</td>
<td>(7.906, 9.243)</td>
<td>8.760</td>
</tr>
<tr>
<td>0.733</td>
<td>9.184</td>
<td>(8.734, 9.634)</td>
<td>9.781</td>
</tr>
</tbody>
</table>

### B.2 (d)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.019</td>
<td>1.309</td>
<td>(1.301, 1.317)</td>
<td>1.316</td>
</tr>
<tr>
<td>0.108</td>
<td>1.486</td>
<td>(1.470, 1.502)</td>
<td>1.467</td>
</tr>
<tr>
<td>0.209</td>
<td>1.762</td>
<td>(1.740, 1.784)</td>
<td>1.740</td>
</tr>
<tr>
<td>0.309</td>
<td>2.156</td>
<td>(2.101, 2.211)</td>
<td>2.143</td>
</tr>
<tr>
<td>0.409</td>
<td>2.772</td>
<td>(2.750, 2.794)</td>
<td>2.829</td>
</tr>
<tr>
<td>0.507</td>
<td>4.445</td>
<td>(4.294, 4.595)</td>
<td>4.662</td>
</tr>
<tr>
<td>0.567</td>
<td>7.160</td>
<td>(6.811, 7.510)</td>
<td>7.372</td>
</tr>
<tr>
<td>0.579</td>
<td>8.693</td>
<td>(7.653, 9.733)</td>
<td>8.538</td>
</tr>
<tr>
<td>Data throughput</td>
<td>Mean data delay</td>
<td>95% Confidence interval</td>
<td>Analytical mean</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------</td>
<td>---------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>0.023</td>
<td>1.681</td>
<td>(1.641, 1.721)</td>
<td>1.658</td>
</tr>
<tr>
<td>0.112</td>
<td>2.039</td>
<td>(2.013, 2.065)</td>
<td>1.985</td>
</tr>
<tr>
<td>0.213</td>
<td>2.524</td>
<td>(2.486, 2.562)</td>
<td>2.528</td>
</tr>
<tr>
<td>0.313</td>
<td>3.637</td>
<td>(3.546, 3.727)</td>
<td>3.713</td>
</tr>
<tr>
<td>0.412</td>
<td>6.988</td>
<td>(6.379, 7.597)</td>
<td>6.956</td>
</tr>
<tr>
<td>0.422</td>
<td>8.414</td>
<td>(7.214, 9.614)</td>
<td>7.931</td>
</tr>
</tbody>
</table>

B.2 (e)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.027</td>
<td>2.530</td>
<td>(2.489, 2.571)</td>
<td>2.404</td>
</tr>
<tr>
<td>0.117</td>
<td>3.211</td>
<td>(3.135, 3.285)</td>
<td>3.126</td>
</tr>
<tr>
<td>0.217</td>
<td>5.116</td>
<td>(4.882, 5.349)</td>
<td>5.229</td>
</tr>
<tr>
<td>0.266</td>
<td>7.357</td>
<td>(6.503, 8.210)</td>
<td>7.151</td>
</tr>
<tr>
<td>0.277</td>
<td>8.553</td>
<td>(7.601, 9.506)</td>
<td>8.102</td>
</tr>
</tbody>
</table>

B.2 (f)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.031</td>
<td>4.937</td>
<td>(4.845, 5.029)</td>
<td>4.503</td>
</tr>
<tr>
<td>0.071</td>
<td>5.647</td>
<td>(5.601, 6.023)</td>
<td>5.367</td>
</tr>
<tr>
<td>0.091</td>
<td>7.209</td>
<td>(6.474, 7.944)</td>
<td>6.303</td>
</tr>
</tbody>
</table>

B.2 (g)
Table B.3: 95% Confidence intervals for the MSTDM simulation results shown in Figure 2.11 ($\alpha = 0.05$, $k = 2$).

(a) $\rho_v = 0$
(b) $\rho_v = 0.045$
(c) $\rho_v = 0.146$
(d) $\rho_v = 0.292$
(e) $\rho_v = 0.439$
(f) $\rho_v = 0.586$

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>1.030</td>
<td>(1.029, 1.030)</td>
<td>1.031</td>
</tr>
<tr>
<td>0.100</td>
<td>1.093</td>
<td>(1.091, 1.096)</td>
<td>1.090</td>
</tr>
<tr>
<td>0.300</td>
<td>1.313</td>
<td>(1.303, 1.322)</td>
<td>1.305</td>
</tr>
<tr>
<td>0.400</td>
<td>1.512</td>
<td>(1.507, 1.518)</td>
<td>1.495</td>
</tr>
<tr>
<td>0.501</td>
<td>1.838</td>
<td>(1.816, 1.860)</td>
<td>1.817</td>
</tr>
<tr>
<td>0.599</td>
<td>2.477</td>
<td>(2.428, 2.526)</td>
<td>2.420</td>
</tr>
<tr>
<td>0.701</td>
<td>4.214</td>
<td>(4.024, 4.403)</td>
<td>4.136</td>
</tr>
<tr>
<td>0.757</td>
<td>7.512</td>
<td>(7.155, 7.870)</td>
<td>7.493</td>
</tr>
</tbody>
</table>

B.3 (a)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.011</td>
<td>1.062</td>
<td>(1.058, 1.066)</td>
<td>1.059</td>
</tr>
<tr>
<td>0.102</td>
<td>1.132</td>
<td>(1.131, 1.132)</td>
<td>1.129</td>
</tr>
<tr>
<td>0.302</td>
<td>1.403</td>
<td>(1.399, 1.407)</td>
<td>1.391</td>
</tr>
<tr>
<td>0.402</td>
<td>1.652</td>
<td>(1.645, 1.659)</td>
<td>1.633</td>
</tr>
<tr>
<td>0.501</td>
<td>2.083</td>
<td>(2.069, 2.097)</td>
<td>2.053</td>
</tr>
<tr>
<td>0.600</td>
<td>2.977</td>
<td>(2.967, 2.988)</td>
<td>2.930</td>
</tr>
<tr>
<td>0.702</td>
<td>6.576</td>
<td>(6.390, 6.761)</td>
<td>6.317</td>
</tr>
<tr>
<td>0.720</td>
<td>8.589</td>
<td>(8.197, 8.982)</td>
<td>8.219</td>
</tr>
</tbody>
</table>

B.3 (b)
<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.014</td>
<td>1.155</td>
<td>(1.147, 1.162)</td>
<td>1.149</td>
</tr>
<tr>
<td>0.104</td>
<td>1.269</td>
<td>(1.260, 1.278)</td>
<td>1.252</td>
</tr>
<tr>
<td>0.204</td>
<td>1.433</td>
<td>(1.426, 1.439)</td>
<td>1.415</td>
</tr>
<tr>
<td>0.305</td>
<td>1.699</td>
<td>(1.681, 1.717)</td>
<td>1.673</td>
</tr>
<tr>
<td>0.404</td>
<td>2.152</td>
<td>(2.131, 2.174)</td>
<td>2.137</td>
</tr>
<tr>
<td>0.505</td>
<td>3.054</td>
<td>(3.007, 3.101)</td>
<td>3.030</td>
</tr>
<tr>
<td>0.603</td>
<td>6.206</td>
<td>(5.979, 6.432)</td>
<td>6.234</td>
</tr>
<tr>
<td>0.624</td>
<td>8.171</td>
<td>(7.965, 8.378)</td>
<td>8.030</td>
</tr>
</tbody>
</table>

B.3 (c)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.018</td>
<td>1.395</td>
<td>(1.371, 1.420)</td>
<td>1.373</td>
</tr>
<tr>
<td>0.108</td>
<td>1.602</td>
<td>(1.574, 1.631)</td>
<td>1.559</td>
</tr>
<tr>
<td>0.208</td>
<td>1.956</td>
<td>(1.930, 1.982)</td>
<td>1.912</td>
</tr>
<tr>
<td>0.308</td>
<td>2.513</td>
<td>(2.479, 2.546)</td>
<td>2.529</td>
</tr>
<tr>
<td>0.408</td>
<td>3.863</td>
<td>(3.748, 3.978)</td>
<td>3.912</td>
</tr>
<tr>
<td>0.458</td>
<td>5.836</td>
<td>(5.435, 6.237)</td>
<td>5.738</td>
</tr>
<tr>
<td>0.478</td>
<td>6.640</td>
<td>(6.198, 7.081)</td>
<td>6.884</td>
</tr>
<tr>
<td>0.499</td>
<td>8.952</td>
<td>(8.233, 9.670)</td>
<td>9.064</td>
</tr>
</tbody>
</table>

B.3 (d)
<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.022</td>
<td>1.824</td>
<td>(1.786, 1.863)</td>
<td>1.747</td>
</tr>
<tr>
<td>0.113</td>
<td>2.304</td>
<td>(2.263, 2.346)</td>
<td>2.193</td>
</tr>
<tr>
<td>0.213</td>
<td>3.123</td>
<td>(3.027, 3.218)</td>
<td>3.085</td>
</tr>
<tr>
<td>0.312</td>
<td>5.336</td>
<td>(5.195, 5.478)</td>
<td>5.571</td>
</tr>
<tr>
<td>0.332</td>
<td>6.447</td>
<td>(6.182, 6.712)</td>
<td>6.589</td>
</tr>
<tr>
<td>0.353</td>
<td>7.957</td>
<td>(7.458, 8.456)</td>
<td>7.922</td>
</tr>
</tbody>
</table>

B.3 (e)

<table>
<thead>
<tr>
<th>Data throughput</th>
<th>Mean data delay</th>
<th>95% Confidence interval</th>
<th>Analytical mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.027</td>
<td>2.897</td>
<td>(2.848, 2.945)</td>
<td>2.678</td>
</tr>
<tr>
<td>0.067</td>
<td>3.353</td>
<td>(3.242, 3.464)</td>
<td>3.133</td>
</tr>
<tr>
<td>0.117</td>
<td>4.102</td>
<td>(3.957, 4.248)</td>
<td>3.922</td>
</tr>
<tr>
<td>0.166</td>
<td>5.231</td>
<td>(4.897, 5.565)</td>
<td>5.118</td>
</tr>
<tr>
<td>0.187</td>
<td>6.368</td>
<td>(6.110, 6.625)</td>
<td>6.072</td>
</tr>
<tr>
<td>0.207</td>
<td>7.247</td>
<td>(6.962, 7.533)</td>
<td>7.155</td>
</tr>
</tbody>
</table>

B.3 (f)
References


REFERENCES


REFERENCES


REFERENCES


