6

Packet Delay and Sequence Number Space in the Radio Link Protocol Layer

by

Euree Y. Kim

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirement for the Degrees of
Bachelor of Science in Electrical Engineering and Computer Science
and Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology jointly with QUALCOMM Incorporated
May 1998
[June 1998]

Author_____
Department of Electrical Engineering and Computer Science
May 8, 1998

Certified
by_____
Dr. Steven Finn
Thesis Supervisor

Accepted
by_____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

Eng.

Packet Delay and Sequence Number Space in the Radio Link Protocol Layer

by

Euree Y. Kim

euree@alum.mit.edu

Submitted to the
Department of Electrical Engineering and Computer Science

May 8, 1998

In Partial Fulfillment of the Requirement for the Degrees of
Bachelor of Science in Electrical Engineering and Computer Science and
Master of Engineering in Electrical Engineering and Computer Science at
the Massachusetts Institute of Technology jointly with QUALCOMM Incorporated

Abstract

The problem of queuing delay through the two-queue model consisting of the M/D/m and the M/G/1 queues is relevant to the Radio Link Protocol (RLP) layer implementation of a high speed wireless packet data service system. The frame round trip delay is an important performance measure of such system, and it influences the size of the sequence number space. The analysis of the frame delay and ultimately the sequence number space size in the RLP layer for a wireless CDMA packet data system is the topic of study in this thesis project. The round trip delay is taken to be the time between the issuance of a NAK by the mobile station and the time that the retransmitted frame leaves the base station transmitter. The queuing analysis of the two-queue model quantifies the round trip delay, which ultimately leads to the estimate of the size of the sequence number space that would reduce the probability of the protocol failure to an acceptably low level. Simulations follow to verify the analytical results and validate assumptions and conjectures.

Thesis Supervisor: Steven G. Finn
Title: Principal Research Scientist, MIT Laboratory of Information and Decision Systems

Acknowledgements

I wish to express my gratitude to my advisors, Steven Finn at MIT and Nagabhushana Sindshuayana at Qualcomm, Inc. They gave me a tremendous amount of guidance and advice without which this thesis would not have been possible.

I would like to thank Qualcomm Incorporated and the MIT VI-A Program for sponsoring this thesis and also for providing such a wonderful opportunity to do research and be a part of this innovative project. This was an exciting and challenging problem to work on, and I will always value this experience.

I would like to thank everyone who has helped me through this past year. Li, thank you for all the help you gave me in preparing this thesis. I value your critical eye and your constructive criticisms you never ceased to provide. Also, thank you so much for your friendship, your sense of humor, and your help in finishing my thesis. You can bet on my help in finishing yours! Henry, thank you for your love and support. Your care and concern have been invaluable to me, and you helped me in ways you could not imagine. Thank you for being there when things got frustrating and tough and for always filling me up with hope.

Lastly, I would like to thank my parents and my brother, Edward, for their endless love and support through all these years. I am very grateful that I have you in my life, and I will always know that I can count on you for your advice and encouragement. You challenged me to never stop going forward and going after my dreams. *Thank you for your encouragement!*

This thesis is dedicated to all of you.

TABLE OF CONTENTS

TABLE OF FIGURES

Chapter 1 : Introduction

The purpose of this thesis project is to evaluate the performance of the Radio Link Protocol in a high speed wireless data packet networks. The Radio Link Protocol is a new link layer protocol, invented only a few years ago. It was designed to aid the implementation of wireless communication networks by reducing the frame error rate observed on the underlying physical layer in wireless CDMA[1] systems. The purpose of this thesis is to evaluate the feasibility of implementing the Radio Link Protocol in a high-speed wireless data service system, by examining two performance issues. Namely, they are frame delay characteristics and sequence number space size requirements. Probabilistic analysis and concepts from queuing theory as well as numerical methods are used to analyze these aspects of the Radio Link Protocol in the wireless CDMA system.

This document is organized as follows. First, the general overview of the environment in which the RLP protocol operates is described, and a brief description of the problem is provided as well as the scope of the thesis project and the general approach that was taken.

[1] Code Division Multiple Access.

Secondly, the general procedures of the RLP is described, especially its transmit and receive procedures and the retransmission strategy. Next, the problems of packet delay and sequence number space are introduced and explored in more depth as well as the approaches and methodology used in the analysis. Fourth, the results of the queuing analysis are presented as well as the simulation procedures and their results. Lastly, we discuss the implications of the results, conclusions and possible future work.

Background Information

The objective of the project is to quantitatively analyze the performance of a CDMA data service system in which the Radio Link Protocol operates; it operates underneath the TCP/IP layer and above the CDMA physical layer[2]. In particular, the issues of packet delay and sequence number space size are explored. Research was done to obtain the delay characteristics of the data units transmitted on the RLP link and the adequate size of the sequence number, especially in view of high data rates. The next subsection describes the overall objective of this thesis as well as the environment in which the RLP operates. It also lists the assumptions that are made about the CDMA system of interest. A brief description of the problem and the general methodology used are also presented here.

A High Speed CDMA System : High Level Overview

The analysis in this thesis is based on the fact that the RLP will be operating in a generic CDMA packet data service system, which is discussed here. The CDMA system of interest will be referred to as High Speed Wireless Packet Data Service (HSWPDS). HSWPDS is intended to operate over IS-95 wireless mobile networks. It can support data rates that exceed 1 Mb/s[3], and it can also support mobility. In a given coverage area, there are a fixed number of base stations and a variable number, depending on the system's capacity, of mobile stations. The

---

[2] The requirements of the CDMA physical channel are defined in the TIA/EIA/IS-95 standards.

[3] Megabits Per Second.

base station is able to service multiple mobile stations; however, it serves one mobile station at a time, unlike in the IS-95 voice systems. A scheduling algorithm dictates the method with which the base station serves multiple mobile stations. The packets are said to be in the forward link if they are transmitted by the base station and said to be in the reverse link if they are transmitted by the mobile station.

For the purpose of handoffs[4], the mobile station maintains a list of base stations that it can receive data from. This list is called the mobile station's Active Set. The packets in the forward link are selected by the Selection Function module and sent off to all the base stations in the mobile station's Active Set. Each base station in the Active Set possesses the forward link packets to be sent to the mobile station should it receive a request for data that the mobile stations wishes to receive. The mobile station selects the base station that it wishes to receive data from based on the signal strengths from the base stations in its Active Set. Figure 1 below illustrates the high level modules of HSWPDS.

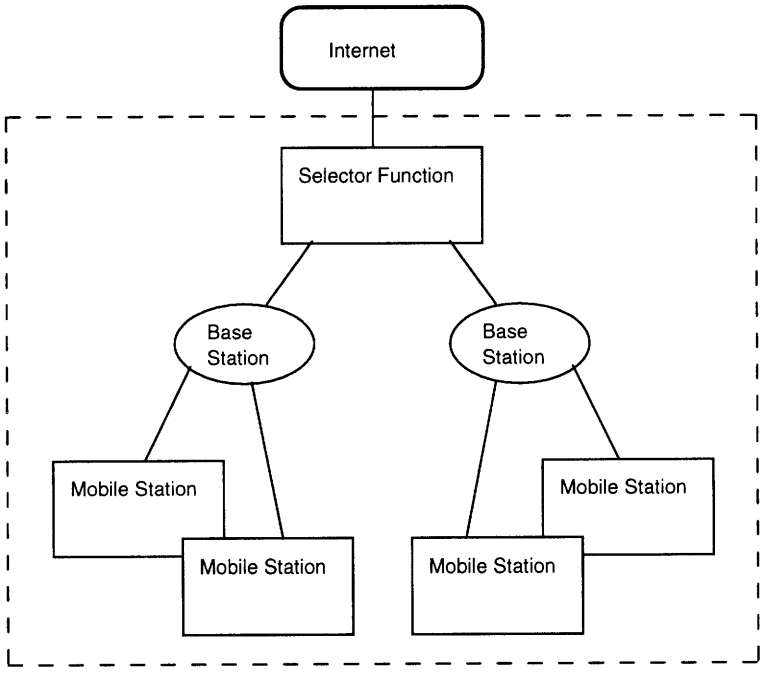


Figure 1: Schematic of High Speed Wireless Packet Data Service System

[4] Handoff aids the mobile station's mobility by providing a means of switching base stations from which it receives data.

The mobile station and the base station are the two ends of the air interface which is the primary interest for the research. The air link is defined by a protocol stack that consists of the CDMA physical layer, the Radio Link Protocol, PPP (Point-to-Point Protocol), and the TCP/IP (Transmission Control Protocol/ Internet Protocol). Figure 2 below illustrates the protocol stack that is assumed for HSWPDS.

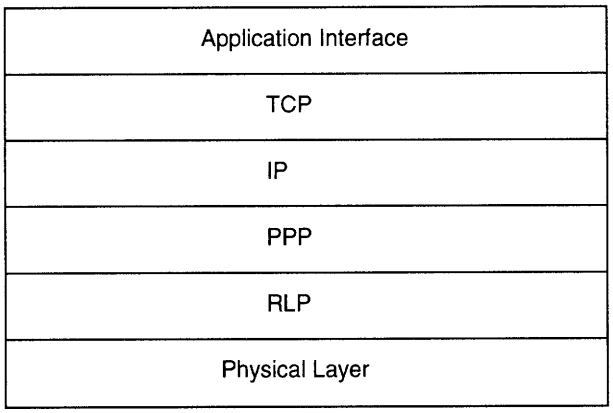


Figure 2: Protocol Stack for HSWPDS.

It is envisioned that a user will be able to perform the usual activities he/she executes on the Internet today using this wireless packet data system: connect through a dial-up application that runs over TCP to request a PPP connection and browse the web, for an example. The focus of this thesis is the Radio Link Protocol layer that operates on the wireless link between the two ends of the air interface.

Figure 3 below illustrates where the RLP operates in the CDMA system relative to the other protocols in the protocol stack.

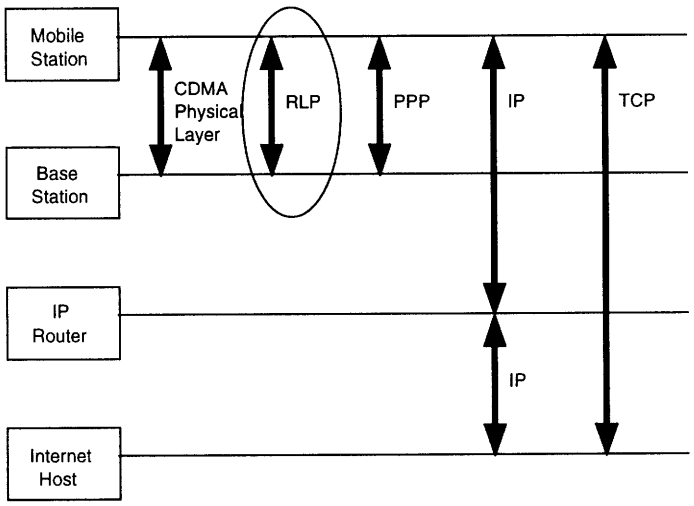


Figure 3: Endpoints of each protocol in the Protocol Stack.

As is evident from the diagram, the TCP/IP layer lies across the two end points (between a mobile station and an arbitrary Internet host) of the data communication link. It provides the end-to-end check on the data as well as flow control and acknowledgments of correctly received data. The PPP layer provides framing and a CRC[5]. The CRC at the PPP layer is very useful because the 32-bit checksum at the TCP layer (it being only a checksum, rather than a CRC) is considered weak. Therefore, the PPP layer operates beneath the TCP/IP layer to significantly reduce the undetected error rates of the protocol stack. The RLP layer operates on the wireless CDMA link, significantly reducing the frame error rate that is observed on the CDMA physical layer. The Radio Link Protocol is the central topic of this research effort.

[5] PPP provides a 16-bit cyclic redundancy check (CRC) on each frame.

Chapter 2 : System Components

The Channel

The wireless channel, the medium in which the RLP frames travel, has several important characteristics. The physical layer of the wireless CDMA system is much less than perfectly reliable; there exists a non-zero frame error rate that corrupts a percentage of the frames that encounter the channel. This is the unavoidable property of the wireless communications channel. The frame error rate of the physical layer, however, is improved by the Radio Link Protocol layer. Its retransmission strategy enables the transmitter to retransmit the frames that were in error, consequently reducing the frame error rate exhibited by the underlying physical layer. This was one of the design goals of the Radio Link Protocol. There also exists a propagation delay through the wireless channel. The distance and the speed of light mainly induce the propagation delay. The propagation delay is deemed negligible compared to the transmission delay and other processing delays throughout this thesis research, and it's ignored in this project. However, should the propagation delay become significant compared to the

other delay components, the results of our research can be modified by adding the propagation delay component.

Base Station

The base station is one end of the wireless link in HSWPDS, and it is the interface between the mobile station and the global network commonly referred to as the Internet. The base station is responsible for obtaining and possessing the information the mobile station wants, and transmitting the data units over the air interface to the mobile station according to some scheduling scheme. The method with which the base station obtains the data units for each mobile station is not within the scope of this thesis and not discussed here. A *frame* or a *packet* is a fixed-size data unit that is transmitted. We use the terms *frame* and *packet* interchangeably. For this thesis, we assume that each frame/packet is 1000 bits long.

The base station is able to serve more than one mobile station in a time-multiplexed scheme. The base station dedicates its forward link channel entirely to a single mobile station at any given time instant. A scheduling algorithm determines when a particular mobile station is to be served. As in Figure 4 below, the base station is modeled as an n-queue module where n is the number of mobile stations that it is currently serving.

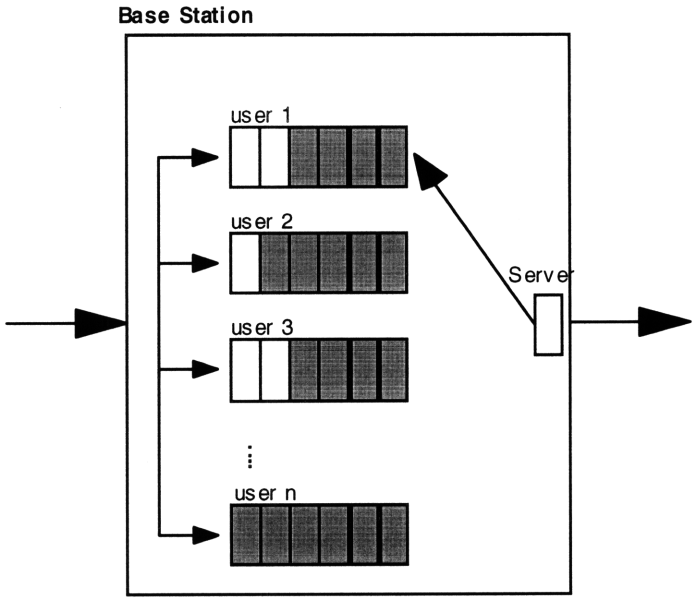


Figure 4: Model of the base station as n parallel transmission queues, each serving one mobile station. Note that the outgoing arrow represents the forward link channel, and one server uses it at a time.

The incoming arrow into the base station represents the flow of forward link data units into the base station that are meant for all the users. The source of the forward link traffic is either the Selector Function module which hold the fresh data units meant for the mobile stations or the data units whose retransmission is requested by the mobile station. Where the frames to be retransmitted come from is discussed later. We assume that there exists a mechanism that sorts the incoming data into the appropriate queue meant for a particular user without a significant delay. Also, nothing specific about the scheduling algorithm is assumed here. One queue exists for each user being served by the base station, and all the queues share one transmitter, which transmits one frame at a time. The transmitter visits each queue and transmits the frame that is at the head of the queue. This way, time-multiplexed transmission is achieved.

Another important issue is the data rate at which the transmitter transmits. The base station transmitter selects a data rate at the start of each frame transmission. This is done through a negotiation process, which requires communication with the mobile station. The data rate that is selected depends on the signal strength, available bandwidth, and the requested rate from the

mobile station. There are a fixed number of possible data rates from which the transmitter selects prior to each frame transmission after the data rate negotiation process.

It should be noted that Figure 4 above depicts a simple base station model. It does not attempt to represent the base station in full detail. The model only contains the components of the base station that are relevant to the questions of interest. What the figure above does not show which is relevant to the questions of interest is the detail of each of the n queues. The queue for each user is actually a prioritized queue; some members in the queue receive preference over others as defined by the specifications for the Radio Link Protocol. A more detailed model of the prioritized queue is developed in later chapters.

In conclusion, we are concerned with the way the base station transmits the RLP frames to all the users it serves. The ultimate goal is to quantify how fast the forward link RLP frames leave the base station and consequently what the delay is like across the air link from the base station to the mobile station.

Mobile Station

The mobile station receives an RLP frame from the base station, decodes it, and selects an action based on the content of the RLP frame. For the purposes of this project which is concerned with the forward link traffic, we model only one kind of RLP frame which the mobile station can receive on the forward link channel: RLP data frames. We ignore the RLP control frames because they typically travel on a separate channel.

The mobile station only receives RLP data frames, and its task is to attempt to decode them as they arrive. As long as the received frame is not an erasure, the mobile station processes the received frame and places it in a resequencing buffer. The schematic model of the mobile station is illustrated below.

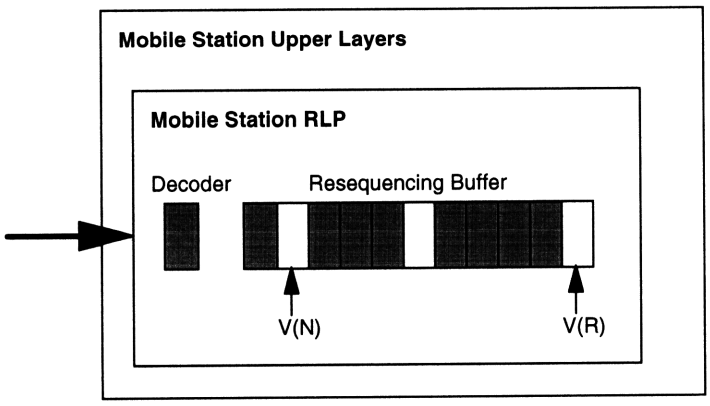


Figure 5: Model of the RLP of the mobile station as a decoder and a fvresequencing buffer.

The resequencing buffer stores the RLP data frames that are received out of order and orders them in sequence. If the sequence number of a valid incoming frame is equal to V(N)[6], the mobile station RLP layer passes up as many contiguous frames as possible to the upper layers, and the frames that are passed up are deleted from the resequencing buffer. If the sequence number of a valid incoming frame is V(R)[7], the received frame is a new expected frame, and the frame enters the resequencing buffer. The specifics about the receive procedure of RLP is described in Chapter 3.

The mobile station is also responsible for sending back to the base station NAKs for any missing frames. Since we require that the base station transmit its frames in order, any out of order frames indicate missing frames. The mobile station then transmits a NAK for every missing frame, requesting its retransmission. The detailed procedure, through which the mobile station does this, is outlined in the next chapter.

6 V(N) is an RLP receive state variable that stores the sequence number of the first frame that is missing.

7 V(R) is an RLP receive state variable that stores the sequence number of the next expected frame.

Problem Description

The focus of this thesis is to examine the issues of packet delay and sequence number space that result from implementing a protocol stack such as above in a high speed CDMA wireless packet data system. In particular, the Radio Link Protocol layer is analyzed in depth to quantitatively and statistically evaluate its contribution to the overall performance of the wireless communication system.

The subject of packet delay has been studied and researched for a long time; it is a critical issue that is the center of many research efforts. It is important because it affects the viability of a communication system. In a voice communication system, for example, delay is a critical constraint since voice communication is real time; it is not difficult to see that it would be very frustrating to speak to someone on the phone where the delay is say, one second. Although the delay constraint is not as critical in a data communication system, it is still a primary measure of the system performance. Packet delay is usually defined as the total time it takes to transfer a packet of data from a sender to a receiver; and obtaining the packet delay characteristic is one of the objectives of this research effort. In particular, the forward link traffic is the major concern since the traffic is heavier, higher data rates are involved, and also scheduling of the mobile stations is involved. For this thesis project, we focus our attention on the packet round trip delay characteristics in the RLP layer; the delay is the time from the departure of a NAK packet at the mobile station to the departure of the retransmitted packet at the base station as response to the original packet from the mobile station.

Packet delay is affected by a number of factors. First, it is influenced by the retransmission algorithm that the RLP employs; because there is always a nonzero probability that the transmitted packet will be in error or lost and therefore needs to be retransmitted, the method of retransmission must be taken into account in analyzing the packet delay. Secondly, packet delay is affected by the scheduling algorithm that the CDMA system employs. From the viewpoint of base station, the packets destined for a particular user do not leave the base station continuously, since the base station is often serving multiple mobile stations. The base station allocates its resources among the multiple users that it services according to a scheduling algorithm, which influences the packet delay from the base station to the mobile station. We take both factors into account in analyzing the delay statistics at the RLP layer.

Each frame on the RLP layer carries a field designated for the sequence number. Because the finite-bit sequence number would wrap around eventually, two packets that are apart by one wrap-around have the same sequence number. In order to prevent these two packets from arriving too close to each other and being recognized as the same packet, care must be taken in selecting the size of the sequence number. The size of the sequence number is influenced by the packet delay characteristic; namely, the peak data rate and the packet interarrival rate. In this research, we apply quantitative and probabilistic methods to compute an adequate sequence number space size, given its dependencies.

Outline of this Thesis

The rest of this thesis document is organized as follows. In chapter 3, we describe the general procedures and the retransmission strategy of the Radio Link Protocol. Since both packet delay and sequence number space depend heavily on the retransmission algorithm of the RLP, it is essential to understand how RLP operates.

In chapter 4, the high level analytical model is presented. The base station is modeled as an RLP transmitter and the data path of interest is captured in the two-queue model, consisting of an M/D/m[8] queue and an M/G/1 queue with vacations. The issues of packet delay and sequence number space are further explored in context of the analytical model.

In chapter 5, the low level queuing model is presented and analyzed. The receive queue, which receives reverse link data frames from all the mobile stations, is modeled with an M/D/m. The retransmission queue, which is responsible for queuing and transmitting the frames whose retransmissions are requested, is modeled with an M/G/1 with vacations.

In chapter 6, the results of the queuing analysis is outlined and explained further. In chapter 7, the simulation procedure is discussed. The simulation model was constructed using OPNET, a communications network simulation software. Some conclusions are drawn and possible

[8] The symbol M indicates Poisson distribution, D for constant distribution, E_k for Erlang distribution, and G for general distribution. In the notation, M/D/m, the first letter refers to the arrival process, the second to the service time distribution, and the third to the number of channels.

future work is suggested in chapter 8. Any possible improvements to the model and analysis presented in the thesis document are also suggested.

Chapter 3 : The Radio Link Protocol

Introduction

In this section, the general procedures as well as the design philosophy of the Radio Link Protocol (RLP) is discussed. The Radio Link Protocol (RLP) provides an octet stream transport service over forward and reverse CDMA traffic channels. RLP frames the data units from the higher layers for transportation on the physical layer; RLP is not aware of upper layer framing, and it operates on a featureless octet stream, delivering the octets in the order received. This chapter will describe the general procedures of RLP as well as the underlying design philosophy of the protocol.

Design Philosophy

Phil Karn of Qualcomm, Inc. first developed the radio link protocol[9] in 1993. It was intended to be a protocol that was compatible with the Transmission Control Protocol (TCP) and the Internet Protocol (IP) and also which reduced the error rate observed on the CDMA traffic channels. The primary motivation behind the design of RLP was to come up with a link layer protocol that will specifically carry TCP/IP segments. In addition, it was heavily inspired by the concept of end-to-end argument[10], developed at MIT LCS[11] by Saltzer, Reed and Clark. Since an end-to-end check and recovery is done at the higher level, the link layer does not need to be absolutely reliable. Increased reliability at the low levels may be redundant or insignificant compared to the cost of providing reliability at the higher levels. Following these principles, the RLP was designed to only provide adequate reliability, so that TCP/IP could easily recover failures without a big impact on performance. The RLP is a NAK-based protocol, unlike some other link layer protocols which are ACK-based.

Since RLP is a NAK-based protocol, the receiver of the data does not respond back to the transmitter unless the data it received is in error, whereas in an ACK-based protocol, the receiver sends a response for every correctly received data unit. Because an RLP receiver sends NAKs instead of ACKs, it is possible that the transmitter never hears from the receiver. This can occur for two reasons. The first possible reason is that the receiver accepts every data unit correctly; consequently, it has no need to send a NAK. The second reason is that the receiver may be malfunctioning, or not functioning at all, in which case it is unable to send the NAKs. The question is, "is it possible for the transmitter to be able to distinguish the two cases?" The protocol will have failed if the answer is no.

How is the RLP transmitter able to distinguish between the receiver not responding due to perfect transmission and the receiver not responding due to malfunctioning? The argument is two-fold. First, the RLP was purposely designed to have only adequate reliability (i.e. a non-zero error rate). Therefore, it is very unlikely that the receiver and the transmitter will observe

---

[9] Qualcomm, Inc. San Diego, C.A., USA, the site of author's MIT VI-A Internship, 1995-1997.

[10] "End-to-End Arguments in System Design" by Saltzer, Reed, and Clark, MIT Laboratory for Computer Science, November 1984, ACM Transactions on Computer Systems II, pp. 277-288.

[11] Massachusetts Institute of Technology, Laboratory for Computer Science.

perfect transmission of every data unit, and NAKs will be sent. Secondly, because the TCP/IP layer performs an end-to-end check, the error will eventually be detected at the higher layers. The TCP/IP layer is ACK-based; the TCP/IP transmitter will notice the absence of ACKs which would indicate that there is a problem. The TCP/IP layer, therefore, is indeed able to take care of erroneous transmissions in the lower layers and even malfunctions of the RLP receivers.

RLP is not absolutely reliable; it was not designed to be. However, it functions to lower the error rates observed in the CDMA physical layers. In a typical CDMA system, a bit error rate of 10^{-4} is not unusual. [12] For 1000 bit frames, a typical frame erasure rate on the physical layer is about 10 percent. The RLP layer lowers the effective frame erasure rate by as much as a thousand-fold. RLP does this with its retransmission mechanism, which will be described shortly. In addition, RLP does not attempt to recover all of its missing data units. After a certain number of retries, the RLP layer passes the data units it has received up to the higher layers for further recovery.

General Procedures

This section provides the general transmit and receive procedures of the Radio Link Protocol. Special attention is paid to the retransmission strategy of the RLP.

State Variables

The Radio Link Protocol is a pure NAK-based protocol. That is, the receiver requests the retransmission of RLP data frames that were not received instead of acknowledging correct data frames. The RLP layer accepts the data packet from a higher layer and formats them for transmission on the physical layer. Every RLP frame contains a sequence number field that identifies the frame. The sequence number of each new RLP data frame is set to V(S), the send state variable maintained by the RLP layer. The state variable V(S) counts octets, and the sequence number of an RLP frame is defined as the octet number, V(S), of the first octet in the RLP data frame. After sending the data frame, V(S) is incremented by the number of octets

[12] Error rate observed in actual test CDMA system as reported by Qualcomm.

that are contained in the frame. Even though an RLP frame can have a variable number of octets, all RLP data frames in our discussion are assumed to be of constant length. The send state variable, V(S), is illustrated in Figure 6 below.

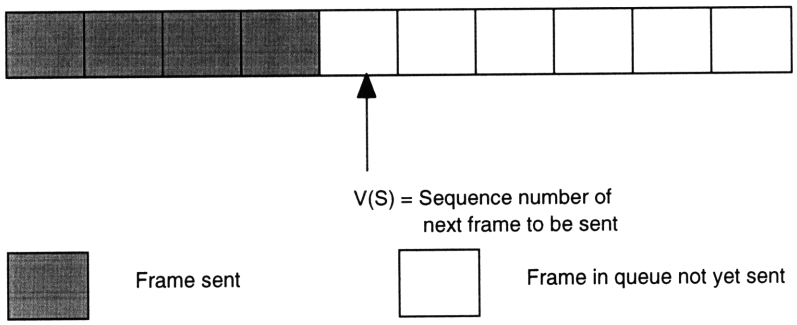


Figure 6: Transmit procedure of the RLP.

The two receive state variables, V(R) and V(N), contain the expected value of the RLP frame sequence number of the next new frame to be received and the sequence number of the next needed frame to make sequential delivery, respectively. This is illustrated in Figure 7 below.

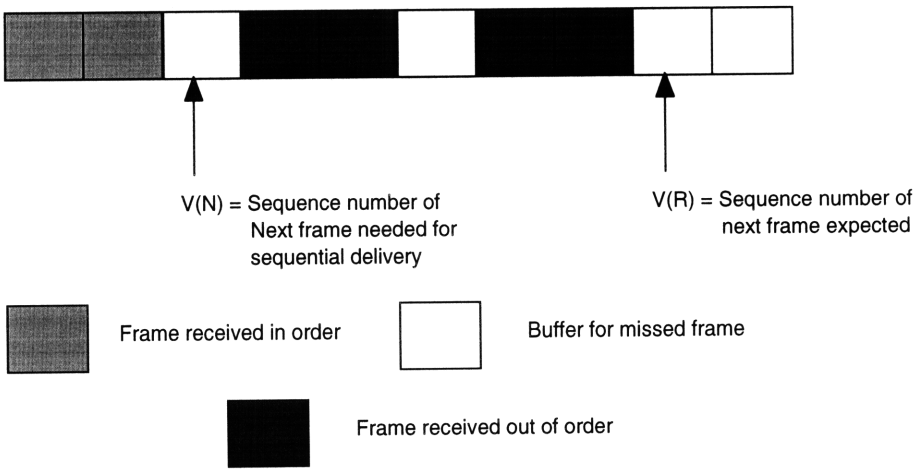


Figure 7: Receive procedure of the RLP.

As the above figure shows, the RLP layer maintains buffer space for resequencing data frames that are received out of order.

Transmit Procedures

The transmit procedure for the RLP layer is as follows. First, the sequence number of the new data frame is set to V(S), the send state variable. The frame is transmitted, and V(S) is then incremented by the number of octets that are in the new data frame.

Receive Procedures

Upon receiving a valid RLP data frame that contains a non-zero number of octets, the RLP layer processes the newly received frame by comparing its sequence number to the receive state variables, V(R) and V(N).

If the sequence number of less than V(N), then the data frame with the sequence number has been received already. The RLP data frame is discarded as a duplicate.

If the sequence number is greater than or equal to V(N) and less than V(R), then the data frame is new, and it is stored in the resequencing buffer. In particular, if the sequence number is equal to V(N), the RLP layer passes the data in all contiguous RLP data frames in the buffer, from V(N) upward, to the higher layer. V(N) is set to a new value. The octets that are passed to the upper layer are then removed from the resequencing buffer.

If the sequence number is equal to V(R), then the data frame is new, and it is stored in the resequencing buffer. In particular, if V(R) is equal to V(N), they are both incremented by the number of octets contained in the new data frame, and all the octets in the new data frame are passed to the higher layer. If V(R) is not equal to V(N), only V(R) is incremented by the number of octets in the new data frame.

If the sequence number is greater than V(R), then the data frame is new, and it is stored in the resequencing buffer. V(R) is set to the received sequence number. In addition, the RLP layer sends one NAK RLP control frame for each unreceived RLP data frame from V(N) to V(R)-1, inclusive, requesting their retransmission. V(R) is then incremented by the number of the octets in the data frame received.

Upon receiving a NAK, the transmitter's RLP layer places copies of the requested RLP data frames in its outgoing queue. For each data frame whose retransmission is requested, the

receiver's RLP layer maintains a NAK retransmission timer that counts frames. A retransmission timer is associated with every frame whose retransmission is requested, every time it is requested. The NAK retransmission timer is incremented for each valid new RLP data frame or an idle frame received. The value of the retransmission time out is dependent upon the parameter RLP_DELAYs, and it is said to be expired when it is incremented to RLP_DELAYs[13].

The transmitter's RLP layer is given three opportunities to successfully transmit a given RLP data frame. In other words, if any RLP data frame requested has not arrived when its NAK retransmission timer expires for the first time, the receiver's RLP layer receiver sends two identical NAK RLP control frames for each unreceived RLP data frame from V(N) upward. The NAK retransmission is then restarted for the requested data frame.

If any requested data frame has not arrived when the NAK retransmission timer expires for the second time, the receiver sends three identical NAK RLP control frames for each unreceived data frame from V(N) upward. The NAK abort timer is then started which is implemented in the same way as the retransmission timer.

If the data frame has not been received by the time the NAK abort timer expires, the receiver's RLP layer sets V(N) to the next missing frame and passes any RLP data frames with sequence numbers less than V(N) in order of sequence number to the higher layer. Further recovery is the responsibility of the higher layer protocols.

By allowing the RLP layer to be given retransmission opportunities, the Radio Link Protocol reduces the apparent error rate exhibited by the physical channel for more efficient operation of the upper layers. However, it intentionally is designed not to be perfectly reliable because the cost of providing the perfect reliability at the RLP layer proves to be marginal since errors may occur at a higher layer which RLP cannot detect, and therefore, as argued by Saltzer, Reed, and Clark, it is much more cost effective to provide end-to-end check at the higher layers.

[13] RLP_DELAYs is an implementation dependent parameter, obtained at protocol initialization.

Chapter 4 : The Analysis Model

In this chapter, the analytical model that is used to quantify the delay through the wireless channel on the Radio Link Protocol layer is presented and described in detail; in addition, the assumptions that are made to simplify the model are discussed here.

Base Station as RLP Transmitter : Model

As previously shown in Figure 4, the base station is modeled as a finite set of parallel queues, one for each user that the base station is serving. Each queue holds RLP frames intended for one mobile station, and the server for that queue transmits the next frame in the queue once permission to transmit is granted by the scheduling module. Let us now focus on one of those queues intended to serve mobile station, i. Figure 8 below takes a closer look at a transmission queue for mobile station, i.

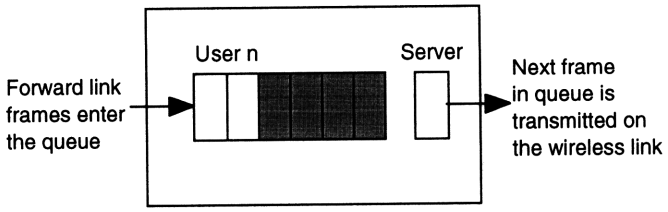


Figure 8: Model of the RLP transmission queue for mobile station n residing in the base station.

The transmission queue for mobile station i is more complicated than the diagram suggests, however. The forward link frames that are placed in the queue include the fresh data units that come from the Selector Function Module as well as the frames whose retransmissions are requested in the NAKs from the mobile station. In addition, the transmission queue is a prioritized queue. The RLP transmit procedure mandates that the frames being retransmitted have priority over the frames being sent for the first time. In other words, the next new frame is sent only when there is no frame to be retransmitted. Therefore, it is necessary to model this characteristic of the queue. It should also be mentioned that we assume the queue holds only data frames. The RLP control frames are assumed to be sent over a separate control channel. In summary, the model of the RLP transmission queue for mobile station i is a prioritized queue, which holds RLP data frames as well as the frames, requested to be retransmitted for that mobile station.

The base station needs to take all the incoming frames, decode them, and sort them. As mentioned previously, the data frames can enter the base station in two different ways. One is the flow of RLP frames from the Selector Function, which provides the base station with the new forward link frames to be sent to the appropriate mobile stations. The other is the reverse link traffic from the mobile stations; however, we are primarily interested in the round trip delay of an RLP frame that begins with the NAK from the mobile station. Therefore, the only part of the reverse link traffic we are interested in is the flow of NAKs into the base station. In order to quantify the round trip delay of an RLP frame, we model the following RLP data path; the mobile station detects a missing frame and it issues and sends a NAK. Upon receiving the NAK, the base station decodes it, places the requested frame in the retransmission queue, and transmits it to the mobile station. Furthermore, we model the portion of the base station

dedicated to a single user as follows with two queues: a receive queue and a retransmission queue. This two-queue model will enable us to quantitatively analyze the round-trip delay defined previously and ultimately arrive at an adequate sequence number space size, which is our desire.

The Two-Queue Model : What and Why

The first queue of our two-queue model, which we will call the *receive queue*, holds all the incoming reverse traffic RLP frames - from all the mobile stations - which wait to be decoded. After each frame is decoded, it is placed in an appropriate place for further processing. If it is decoded as a reverse link data frame, it is placed in a resequencing buffer and later passed up to the upper layer. If it is decoded as a NAK, which is the case we are concerned with, the appropriate frame to be retransmitted is put into the second queue in our model, which we will call the *retransmission queue.* Figure 9 below illustrates the model of these queues. Note that the two-queue system such as in Figure 9 works for a single user. In a base station serving multiple users, k number of such two-queue system would be found. The queue labeled forward outgoing queue in the diagram represents the queue that is filled with the fresh data frames from the Selector Function. We are not concerned with the way the data frames enter the forward outgoing queue or how the frames in this queue are processed; it is shown here for completeness. We are only interested in the fact that the transmitter selects the appropriate frame from either of these queues according to the priority scheme mentioned previously - forward outgoing or retransmission queues - for transmission onto the wireless link.

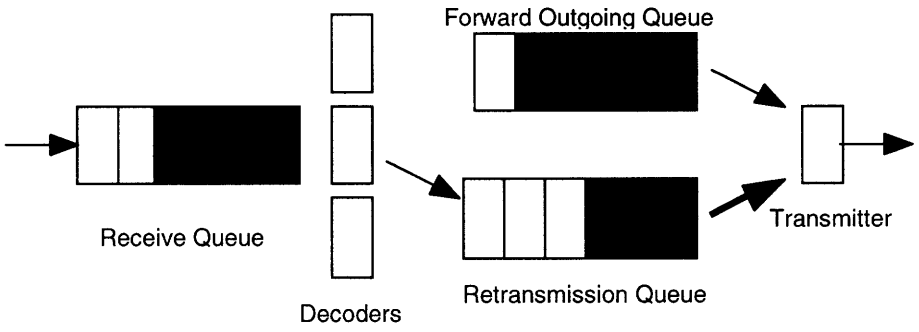


Figure 9: More detailed model of a base station transmission queue. Note that this model serves a single user.

When a transmitter visits a particular user, the transmitter selects the next frame from the forward outgoing queue only if the retransmission queue is empty. All the frames in the retransmission queue have priority before the frames being sent for the first time. Because of this priority system, we are able to ignore the forward outgoing queue in our queuing analysis. We also assume that there are multiple decoders to read the reverse link traffic frames that are coming in and waiting in the receive queue. This attribute is taken from an actual implementation of the CDMA system.

The transmitter, when it is given its turn, transmits the next frame in the retransmission queue or the forward outgoing queue (if the retransmission queue is empty). Therefore, since the base station only serves one mobile station at a time, the transmitter for user n does not work all the time. After sending a frame, it waits for the scheduling algorithm to indicate that user n is to transmit again. Therefore, the complete model for the queuing analysis consists of a receive queue and a retransmission queue.

It is important to note that the two-queue model is not a cascaded queue model. That is, the input to the retransmission queue is not taken entirely from the output of the receive queue. This is because only a fraction (a small fraction, actually) of the frames decoded in the receive queue results in a retransmission request, and only a retransmission request causes a frame to enter the retransmission queue. This allows the two queues to be decoupled and analyzed separately. The delays through the queues only need to be added at the end.

With the model above, we want to quantify the round trip delay of an RLP frame. We model the receive queue as a M/D/m queue, and we model the retransmission queue as a M/G/1 queue. The arrival characteristic of the frames into the receive queue is assumed to be Poisson, and the multiple servers function as the decoders for the frames in the queue with a constant service rate. A later section presents the parameters assumed in the analysis to put things in perspective. A closed form solution to a general M/D/m is not available in the literature. However, an approach was taken to break down the queue into multiple analytically more tractable queues to obtain an upper bound on the queuing delay.

The second queue, the retransmission queue, also is assumed to have an exponential packet arrival rate. The service time has a general distribution; in particular, the service time represents the different data rates that the transmitter is permitted. Namely, there are seven discrete levels of data rates that are permitted by the transmitter. The choice of data rate is made each time the transmitter sends a frame, and it depends on the available bandwidth and signal strength among other things, and a complicated negotiation process occurs between the base station and the mobile station to agree on the data rate. Thus, the characteristic of the service time takes the form of a pmf – *probability mass function.* The probability mass function of the data rate is taken from simulations from an actual implementation of a CDMA system. We assume that the scheduling is done in a simple round-robin fashion. Therefore, a particular mobile is served after every other mobile has been served, and so on. In summary, the retransmission queue is modeled as an M/G/1 queue. The service rate is decided for each frame according to the service time probability mass function, and the transmitter employs a simple round robin scheduling algorithm in visiting the user queues. Simulations are used to validate the two-queue model. See Chapter 7.

Figure 10 below pictorially summarizes the queuing model – the receive queue as an M/D/m and the retransmission queue as an M/G/1.

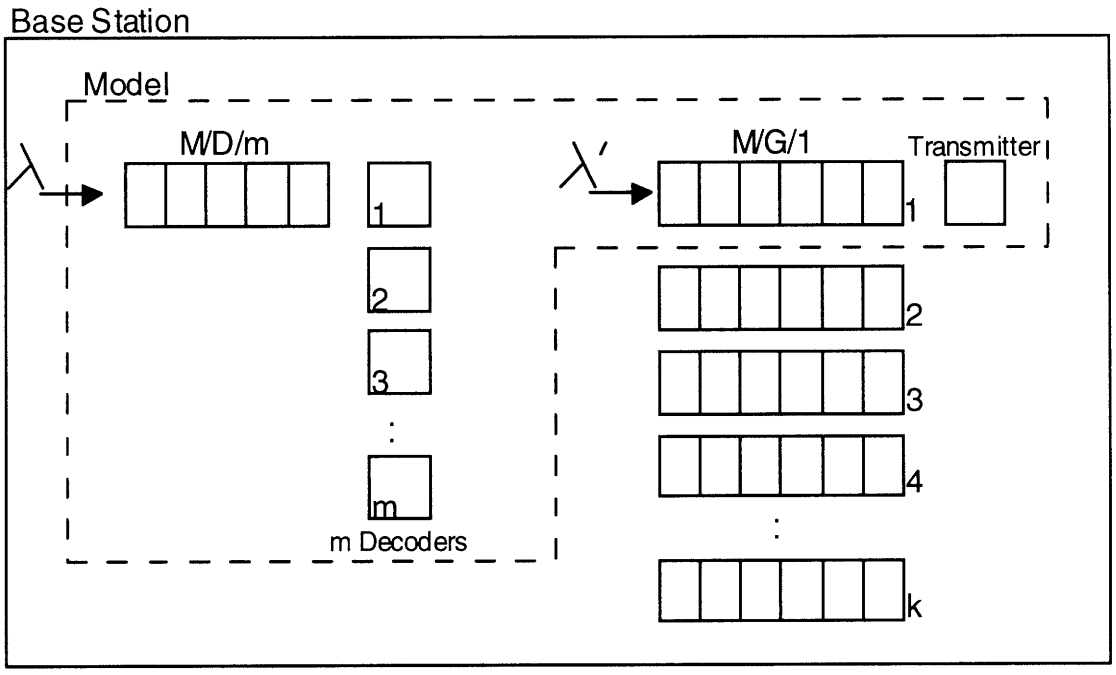


Figure 10 : The solid box represents the model of the base station with the receive queue and each of the k M/G/1 queues intended for each user. The dashed box represents the portion of the base station modeled here; it includes the portion of the base station that is devoted to serving one user.

Packet Delay : A Word About the Analysis Approach

As previously mentioned in the discussion for the motivation of the two-queue model, we define the round trip of an RLP frame to be the time between the transmission of a NAK by the RLP receiver, the mobile station, to the reception of the requested octet at the mobile station. The round trip delay is illustrated below.

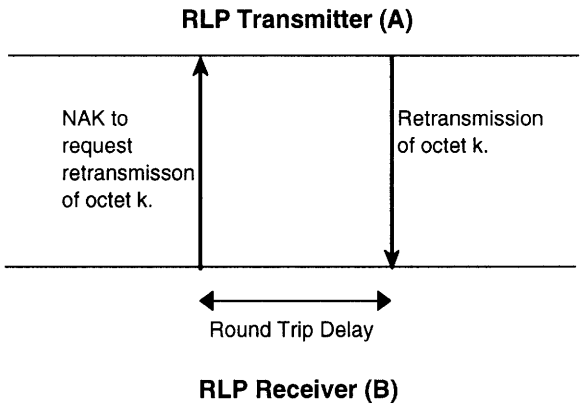


Figure 11: Definition of the round trip delay in the RLP. From the receiver's point of view, it is the time between the sending of a NAK and the reception of the retransmitted frame.

Choosing to neglect the propagation delay through the wireless channel, the round trip frame delay is simply the queuing delay through the two-queue model presented in the previous section. The goal is to find the queuing delay through the M/D/m queue and the M/G/1 queue with vacations.

The round trip frame delay also depends on other factors. It is influenced by the retransmission strategy of the RLP. This is because it may take more than one round trip delay to deliver an RLP frame correctly. If q [14] is the probability that a given frame is received at the mobile station with no errors, it is easy to see that there will be no retransmission of the frame with probability q (i.e. the original transmission is successful). Note that we are assuming independence of events such as frame errors. The probability that a packet is received at the first retransmission is $(1-q)q$. The probability that a packet is received at the second retransmission is $(1-q)^2 q$, and so on. This result is summarized as the probability that the m^{th} retransmission is successful.

$$P(m) = (1-q)^m q$$

[14] The value of q for the RLP is typically about 0.9.

Since the RLP allows up to three retransmissions, given $q = 0.9$, the expected number of transmissions of a frame is 1.111. The probability that a frame is not successful, at which time the RLP layer "gives up", is $(1-q)^4 = 10^{-4}$. We illustrate the RLP retransmission strategy with a feedback loop system. Figure 12 below represents the feedback loop with the two-queue system.

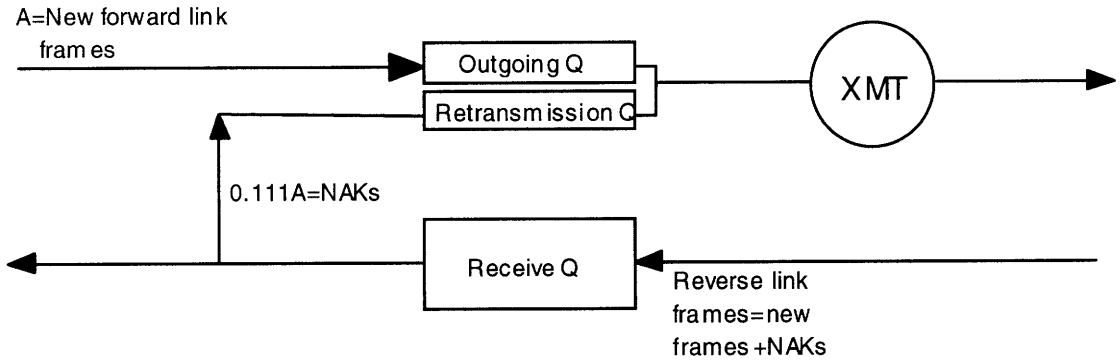


Figure 12: The feedback loop system that illustrates the interaction of the two-queue model and the RLP retransmission algorithm. A has a unit of frames per second. This model applies to either a single user (in which case the transmitter is not always attached to the queuing system, of course) or a composite system for multiple users.

Based upon a typical CDMA physical layer bit error rate of 10^{-4} and a frame length of 1000 bits, approximately 10 percent of the transmitted frames will be received in error. Given the maximum of three retransmissions under RLP, the reverse link traffic contains retransmission requests amounting to 11.1 percent of the forward link new packet traffic. Retransmissions join the stream of new outgoing forward link frames in the forward link queue. Another way to look at it is to think of 1000 packets initially sent on the forward link. Based on the CDMA physical layer frame erasure rate, about 100 of them come back, requesting retransmission. Of the 100 first time retransmissions, about 10 of them come back again for a second retransmission. Of these 10 retransmitted frames, approximately 1 comes back requesting a third retransmission. This last packet has 10 percent chance that it will fail, after which RLP no longer tries to recover. Therefore, operating over a physical channel with 10 percent frame erasure rate, the RLP lowers the erasure rate by a factor of 1000, making the effective frame

erasure rate 10^{-4}. The upper protocol layers have the responsibility to perform the end-to-end check.

The frame delay also depends on the scheduling algorithm involved. Because the scheduling algorithm determines how often frames are transmitted to a particular mobile station, it cannot be ignore in analyzing the packet delay. It is not the intention of this thesis to discuss the details of different scheduling algorithms or to evaluate their merits. However, it is the goal of this project to recognize that the scheduling algorithm makes a vital contribution to the packet delay in this particular kind of CDMA communication system, and assume a simple but reasonable scheduling algorithm in quantifying the packet delay. Consequently, a simple round-robin scheduling scheme is incorporated into our two-queue model.

Sequence Number Space : A Word About the Analysis Approach

The fundamental problem of the sequence number space is that the sequence number is represented with a finite number of bits. As a consequence, two frames labeled with the sequence number, say k, must be far enough away from each other so that they are not mistaken to be the same frame. That is, the wrap around of the sequence number cannot occur too often or at the wrong time. This problem is especially of importance because of the retransmission strategy of the RLP; the retransmission mechanism can potentially result in a sequence number that is valid for a long time, although with a small probability. The problematic situation is depicted in the figure below. Ideally, the sequence number space should be big enough so that the wrap around does not happen too often, thus decreasing the chance of two distinct frames with the same sequence number arriving at the destination at around the same time. However, it should not add unnecessarily many overhead bits to the data frames. This is the inevitable trade off of this type of a communication system. The purpose of the analysis in this thesis is to recommend a sequence number space size that would ensure that the probability of two distinct frames labeled k are mistaken to be identical is very low.

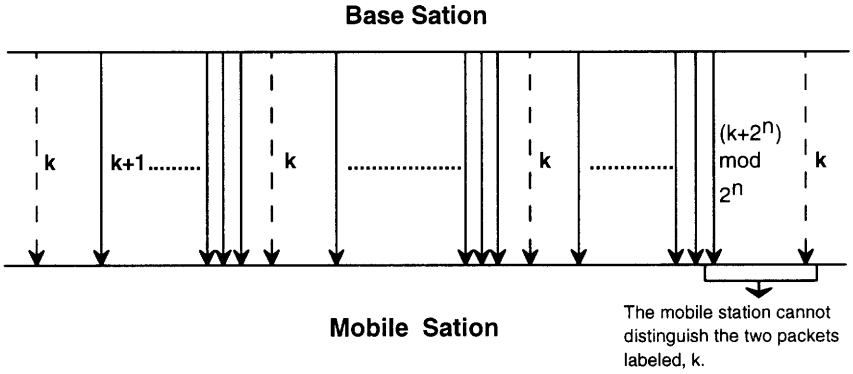


Figure 13: The fundamental problem of the finite length sequence number.

The situation illustrated above can arise because the transmitter is allowed to retransmit data frames that the receiver requests to be sent again. Assuming the sequence number counts octets (or equivalently, any transmittable data unit), on the chance the octet k is retransmitted several times, the sequence number k that refers to the octet being retransmitted would be valid for quite some time. In the mean time, the sequence number could increment to a point where it wraps around to reach k again. Some retransmission strategies such as ARQ (*a*utomatic *r*epeat re*q*uest) avoid this event with use of a *sliding window*. Windowing is a mechanism that is used to ensure that the transmitter does not send too many packets too fast without making sure that they were received correctly. For example, in the stop-and-wait ARQ, the window size is 1; the correct reception of a packet is ensured with an ACK from the receiver before transmitting the next packet. After sending a packet, the transmitter waits. Upon receiving a NAK, the transmitter retransmits the packet, and upon receiving an ACK, it proceeds to send the next packet. If either the frame or the return ACK or NAK is lost, the transmitter eventually times out and resends the old packet. The event depicted in the figure above is avoided with absolute certainty in the stop-and-wait ARQ; the sequence number space in stop-and-wait ARQ does not need to be larger than 1. However, delay is prolonged because correct reception is confirmed one packet at a time. The packet delay is improved in two other retransmission strategies, namely go back N and selective go back N ARQ. The size of the window is increased to allow the transmitter to send more than one packet without having to wait. The sequence number

space requirements of these two retransmission strategies are discussed in [2]. We now extend a similar approach to discuss the RLP retransmission strategy.

The RLP Retransmission Method and The Sequence Number Space

We now consider the bounds on the sequence space on an RLP link. The RLP retransmission method was discussed in a previous section. In short, the transmitter is given up to three opportunities to retransmit an RLP frame. The receiver sends NAKs for frames that it detects to be missing, and the transmitter sends off the requested frame upon receiving a NAK. However, since there is no windowing mechanism similar to the one used in the go back protocols, the transmitter has no upper limit on the sequence number of the packet it can send at any given time. Because of the absence of a window, there is a non-zero possibility that the sequence number will wrap around to reach k again while the frame k is still being retransmitted; thus, we can only put a bound on the sequence number space with less than complete certainty. It is the purpose of this thesis to analyze the requirements on the sequence number space of RLP.

Let A be the RLP transmitter and B the RLP receiver. Let N_b be the last octet successfully received by B and N_n the next octet successfully received. We wish to place an upper and lower bounds on the value of N_n. Because the RLP transmitter A has a window of infinite size, it is impossible to place a bound with absolute certainty. The best we can do is recommend a sequence number space that would fail with an acceptably low probability.

We first consider the upper bound on N_n. Note that since we are assuming frames of fixed size, the sequence number refers to a particular frame, rather than an octet. The largest sequence number that N_n can possibly be depends on several factors. One is the length of the burst error. Since erred frames beyond N_b cannot be decoded correctly, the next successfully decoded frame can have the sequence number of $N_b + k$, where k-1 is the number of consecutive frames that were received in error. k is a positive integer greater than 0, called the length of the burst error. The figure below illustrates the scenario in which the length of the burst error is k-1. k can possibly get quite large, and we can quantify the

probability that N_n is $N_b + k$ to be $(1-q)^{k-1}q$, where q is the probably of receiving and decoding a successful octet. In addition, the probability that N_n is greater than or equal to $N_b + k$ is $(1-q)^{k-1}$. We make the assumption that the NAKs are always error-free. The upper bound on N_n also depends on the RLP retransmission timer and the nature of the scheduling algorithm. The RLP timer determines the length of the time the receiver waits to explicitly request a frame that it detects to be missing. If the retransmission succeeds, the sequence number of the retransmitted frame becomes invalid, and the sequence number can go up further. In a sense, as the retransmissions succeed, the *sliding window* of RLP moves up, allowing higher sequence numbers without causing trouble. The scheduling algorithm determines the approximate frequency at which the receiver receives frames, and it is easy to see this, along with the loading of the system, will have a significant effect on the upper bound of the sequence number, N_n.

In this thesis project, we wish to find the upper bound of N_n, with an acceptably small probability that it will exceed this upper bound.

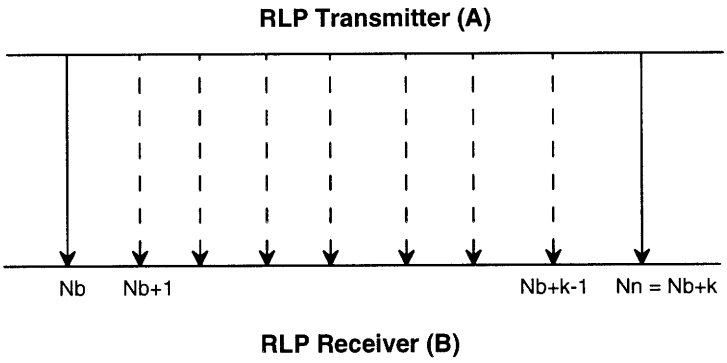


Figure 14: An RLP scenario in which the burst error is k -1 frames long.

Next we consider the lower bound on the value of N_n. The lower bound is much more difficult to analyze because the retransmission algorithm of RLP plays a more significant role. The critical question is as follows. Assuming that the value of N_n is N_b -d, how big can d be? In other words, an octet from how much far in the past could still be valid and must be

accepted as a retransmission of a previously failed frame? We wish to quantify the probability of events so that we can put a bound on the value of d.

It is desirable that the value of N_n is between $N_b + k$ and $N_b - d$ with a probability very close to 1.

$$P\big((N_n < N_b + k) and (N_n > N_b - d)\big) \cong 1,$$

which can be rewritten as,

$$\begin{aligned} 1 - P\big((N_n > N_b + k) or (N_n < N_b - d)\big) &= 1 - \big[P(N_n > N_b + k) + P(N_n < N_b - d)\big] \\ &= 1 - P(N_n > N_b + k) - P(N_n < N_b - d) \\ &= 1 - \alpha_k - \alpha_d \\ &= 1 - (\alpha_k + \alpha_d) \end{aligned}$$

In order for the left-hand side to be close to 1, α_k and α_d must be very small. Let them be the upper limit on the corresponding probabilities. We have already considered $\alpha_k \geq P(N_n > N_b + k)$ along with the issues of the length of the burst error, the RLP retransmission timer, and the scheduling algorithm. We now consider the RLP retransmission strategy more carefully to study the lower bound, $\alpha_d \geq P(N_n < N_b - d)$.

In computing this quantity, we wish to compute the probability that the newly arrived frame (N_n) has the sequence number d or more less than the previously successful octet (N_b). In other words, we are concerned with the number of fresh frames between the original transmission and the final and successful transmission of the frame, $N_b - d$. Let N_w be the number of fresh frames that were received by B since the original transmission of $N_b - d$. The figure below illustrates the scenario in which the quantity, N_w, is equal to 7. The dashed lines represent erred transmissions while the solid lines represent successful transmissions and receptions. The bold line signifies the final and successful transmission of $N_b - d$. NAKs and other transmission failures are omitted from the diagram.

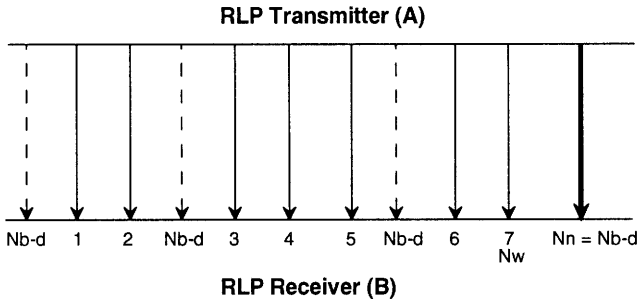


Figure 15: An RLP scenario in which the number of fresh octets since the original transmission of $N_b - d$ is N_w=7.

N_w represents the number of frames we have to trace back to find the frame's first transmission. With the definition of the new quantity, N_w, it is easy to see that the following equality holds.

$$P(N_n < N_b - d) = P(N_w > d) \leq \alpha_d$$

Therefore, the problem is now modified to find the probability that N_w is greater than d. In particular, we want to select d so that the probability is bounded by α_d. The proposed approach is to consider the new quantity, N_w, and find out its upper bound with a small probability that it will exceed this upper bound.

We acknowledge that the upper bound heavily depends on the RLP retransmission scheme and the round trip delay of an RLP frame. Because we are tracing back in time, the manner in which the RLP retransmits a particular frame is very critical. Also, because the scheduling algorithm directly affects how fast the receiver receives data frames, the number of RLP data frames that could have been sent since the original transmission of N_b-d depends on the scheduling. Therefore, the loading of the system since the number of mobiles in the system greatly influences how often a particular mobile is served. Once the upper and lower bounds of N_n are obtained, we are able to estimate the size of the sequence number space that will fail with a small probability.

The sequence number space size is closely related to the RLP round trip frame delay. Another approach to obtain the adequate sequence number space size is to consider the RLP round trip delay and how many packets can be sent during that delay. Given our definition of the round trip delay, the sender of the RLP frame can potentially keep transmitting upward in the sequence number space while a particular sequence number stays valid because the particular packet is awaiting retransmission. In attempting to avoid the wrap around of the sequence number space right around when a packet is being retransmitted, we limit the sequence number space size according to the RLP round trip delay. In the worst case, this would correspond to the number of frames in transit assuming the highest data rate. This quantity, the number of data frames in transit at any given time, is a reasonable estimate of the sequence number space size. As we will see later, this will likely to be a conservative estimate, because (1) the transmitter does not transmit all the time and (2) the transmitter does not always transmit at the highest rate possible.

Chapter 5 : Queuing Analysis

In the previous chapters, we established the notion that the RLP frame round trip delay is simply the sum of the queuing delay through the two queues in the proposed analytical model. In this chapter, we present the analysis of the two-queue model. The two-queue model consists of an M/D/m queue, representing the receive queue, and an M/G/1 queue with vacations, representing the retransmission queue. The objective is to arrive at a quantitative measure of the frame delay through the queues. We explore the M/D/m queue in the next section and the M/G/1 queue in the following section.

The Receive Queue as an M/D/m Queue

The receive queue accepts all RLP data frames from all the mobile stations in the sector that it is covering. The servers decode the frames in order and channel them in the appropriate direction. It is modeled as an M/D/m queue. The arrival process of the receive queue

includes the arriving frames from the multiple mobile stations being served, and it is assumed to have an exponentially distributed interarrival time. The service time for each of the servers has a constant distribution, since the frame sizes are constant.

No analytical closed form solution of the queue - the probability distribution of the queuing delay - of this type exists in the literature. A closed form solution for the occupancy probabilities does exist; however, heavy numerical computation is necessary to arrive at the occupancy probabilities [8], which we choose not to utilize here. Instead of attempting to arrive at the exact probability density function of the queuing delay, we introduce a transformation of the queue into a single-server queue whose solution completely describes the behavior of the original M/D/m system. The transformation is done as follows. The M/D/m queue is broken down into m E/D/1 queues. This way, analyzing one of the m queues enables us to completely understand the behavior of the collective system. The justification for this is very straightforward to see.

Transformation of the M/D/m into m E/D/1 Queues

We claim that the delay characteristics of our M/D/m queue is equivalent to that of an E/D/1 queue; in other words, understanding one will give a complete description of the other. Consider an M/D/m system in which the servers are ordered. The frames that are removed from the queue enter the servers in order (i.e. the first frame enters the first server, the second to the second server, and so on). With this scheme, every m^{th} frame will always enter the same server because the constant service rate ensures that the server is idle by the time m^{th} frame is removed from the queue. What results is the decoupling of the system into m separate queues, namely m E/D/1 queues. The input to each of the E/D/1 is Erlang of order m. Queue for user i receives every m^{th} frame of the Poisson input into the M/D/m queue. Because the queues are completely decoupled, the behavior of the original M/D/m system is preserved. The figure below illustrates the M/D/m to E/D/1 transformation.

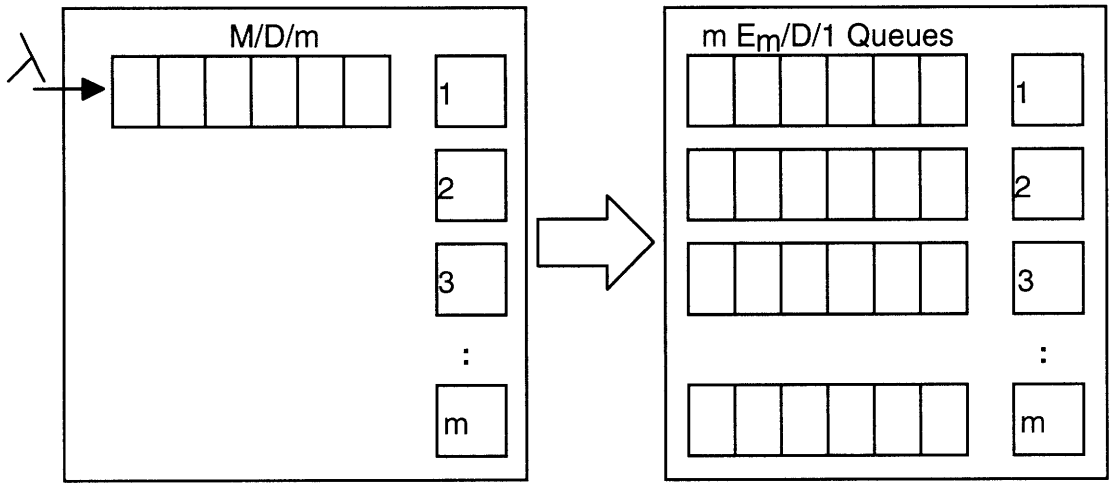


Figure 16 : Problem formulation in which the M/D/m queue is broken down into m separate and independent E/D/1 queues which have Erlang m arrival characteristics.

The resulting system consists of m independent queues that have Erlang-m arrival characteristics and constant service rate. Now, we analyze a single E/D/1 queue.

The E/D/1 Queue

No known analytical solution exists for the E/D/1 queue either. The analysis of an E/D/1 queue is very difficult because of the complexity of the residual service time analysis as well as the combination of random and deterministic characteristics of the system. There is no clean and simple way to express neither the states of the system nor the transitions between them. An attempt was made to obtain the occupancy probabilities for the E/D/1 queue; however, we arrived at a mathematical relation, between the joint probability of number in the system and residual service time at one instant in time and the corresponding probability at another instant in time, which seems impossible to solve. A rigorous and complete theoretical description of the E/D/1 queue is yet to be developed.

So, we take a different approach. For the moment, we assume that the occupancy probabilities for the E/D/1 are known. Isolating a single E/D/1 queue, we examine the upper bound on

the queuing delay, given that there are n frames already in the queue at the time of a new arrival. Since the service rate is constant, computing the upper bound delay is trivial if we know the number of frames already awaiting service. It is simply,

$$(\text{Queuing Delay} \mid n \text{ frames waiting}) \leq nD + D = D(n+1),$$

where D is the deterministic service time. The additional D is the maximum amount of residual service time for the frame currently being serviced at the time of the new arrival. Also the above expression is an inequality because we are assuming the maximum residual time. The residual time spans from 0 to D, and it is uniformly distributed; the probability of entering at a time while a frame is being serviced is equal to the probability of entering it at any other time. Since we know the distribution of the residual time, we can do better than the above bound.

$$p(w = (n_o + 1)D + \alpha \mid n_o) = \text{constant} = \frac{1}{D},$$

where α is the residual time, and w the delay through the system. With the quantities, $p_n(n)$, we can obtain the marginal probability distribution for the waiting time. It is not hard to see that the probability distribution for the system time will look like a staircase, with the width of the step D and heights dictated by $p_n(n)$. However, as described already, the probabilities, $p_n(n)$, are yet to be found. Thus, we explore the possibility of utilizing the occupancy probabilities for the E/M/1 queue (i.e. a more general case) to approximate the delay distribution for the E/D/1.

E/D/1 vs. E/M/1

The analytical solution of the occupancy probabilities for the E/M/1 queue exists. We denote it by $p_n^{E/M/1}(n)$. We conjecture that $p_n^{E/M/1}(n)$ will provide a upper bound approximation of the delay through the E/D/1 queue. Comparison of the M/M/1 and the M/D/1 queues shows that the average waiting time for the M/D/1 is half the waiting time for the M/M/1

queue; the M/D/1 performs better. We conjecture that the relative performance of the E/D/1 and the E/M/1 will parallel that of the M/D/1 and the M/M/1. This conjecture was verified by simulation for the range of parameters considered in this work. See example in Figure 17.

Therefore, we used the analytical results for the steady state occupancy probabilities of the E/M/1 queue to bound the delay through the E/D/1 queue. Figure 17 below plots the function, $p_n^{E/M/1}(n)$, versus n. The system parameters that are used are shown following the plot. For our analysis, an Erlang input process is of order 10 with the mean interarrival rate of 100 frames per second. The service time is 200 frames per second. Thus, the system utilization factor is 0.5.

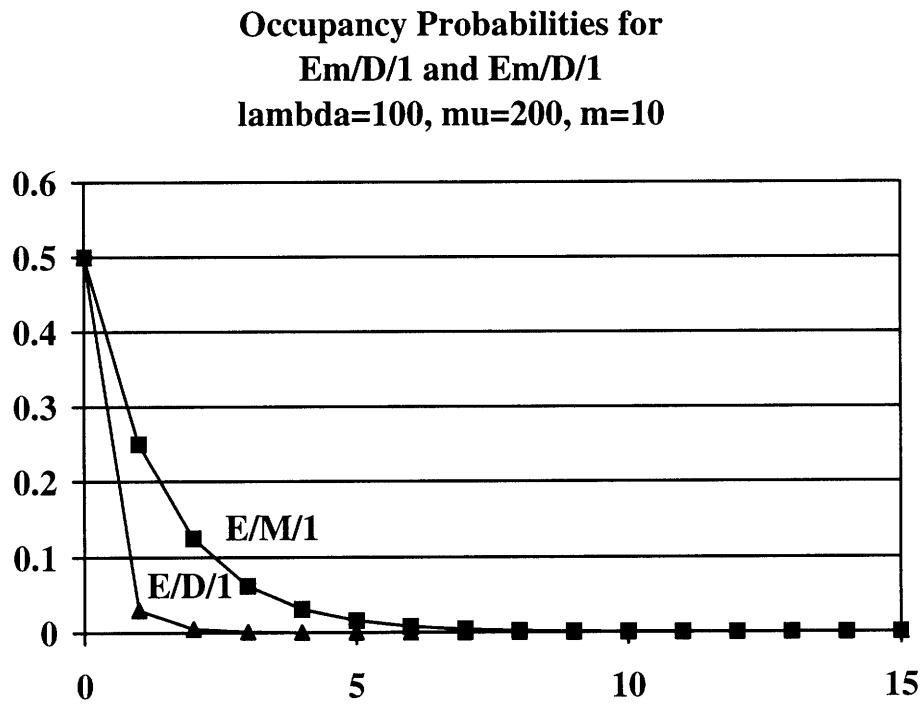


Figure 17 : Occupancy probabilities for the E/M/1 queue, analytically obtained.

From the analytical result, the queue length of the E/M/1 is found to be greater than 14 (N_{max}) with probability of about 10^{-6}. The probability of an arrival finding an empty queue is 0.5, which is as expected. Since the E/D/1 is expected to have a smaller average queue length,

we expect[15] $p_n(n)$ for the E/D/1 to be more "scrunched" up towards the lower states than the E/M/1. This allows us to use $p_n^{E/M/1}(n)$ to approximate the delay probabilities for the E/D/1. In particular, $p_n^{E/M/1}(n)$ will provide the upper bound delay. To compute the upper bound value of the delay, w_0, for which the probability of exceeding it is, say, 10^{-6}, we would have to find the maximum $N_{\max}$ such that $\sum_{n=0}^{N_{\max}} p_n^{E/M/1} < 1-10^{-6}$. Then, w_0 is greater than $(N_{\max}+1)D$[16] with probability less than 10^{-6}. This results in an overestimate of the delay characteristic, using $p_n^{E/M/1}(n)$ instead of $p_n^{E/D/1}(n)$. In an ideal world, we would have the exact distribution, $p_n^{E/D/1}(n)$; however, using the distribution, $p_n^{E/M/1}(n)$, provides a conservative estimate. The expression below illustrates this bound.

$$p(w=(n_o+1)D+\alpha)=\sum_n p(w|n)p_n(n) \le \frac{1}{D}\sum_{n=n_0} p_n^{E/M/1}(n)=\frac{p_n^{E/M/1}(n_0)}{D},$$

α is a positive number between 0 and D, signifying any residual service time. The waiting time distributions looks like a staircase, with the interval lengths D. Or, expressed in terms of a CDF,

$$p(w<(n_o+1)D+\alpha) \le \sum_{n\le n_0} p_n^{E/M/1}(n)$$

The above bound is not a very tight bound. Given the value from the analytically obtained queue size probabilities, the estimate for the delay for which $p(w \ge w_0) \approx 10^{-6}$ is bounded by $(N_{\max}+1)D$, or 75 milliseconds. As the simulation results will show, this is an overestimate. The E/M/1 result is a loose upper bound. Simulation will show, however, that the E/M/1 bound is useful for predicting the required sequence number space size for our CDMA system.

---

[15] Verified by simulation (e.g. see Figure 17).

[16] D=5 milliseconds.

The Retransmission Queue as an M/G/1 Queue

The retransmission queue is modeled as an M/G/1 queue. This section discusses the system characteristics surrounding the retransmission queue, introduces the model we use to describe the behavior of the retransmission queue, and finally presents the analytical results found. For this queuing subsystem, we consider the model in which all the packets whose retransmission is requested enter a single queue and all the other packets enter the appropriate queue depending which user they are destined for. Thus, the queue containing the retransmitted packets is modeled as a preemptive M/G/1 queue.

The Radio Link Protocol specifies that the packets being retransmitted receive priority over packets being sent for the first time. In order to incorporate this property of the retransmission queue, we model it in the following way.

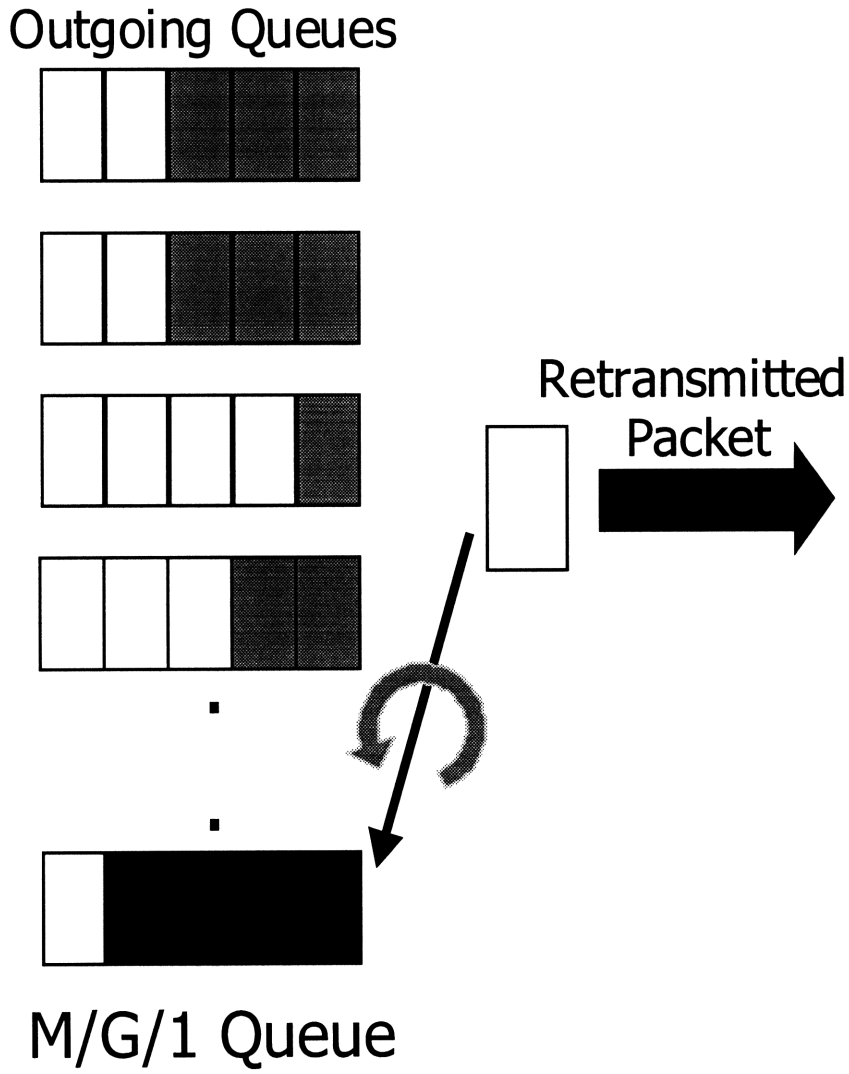


Figure 18 : The model of the retransmission queue. All the retransmitted packets enter a single queue, and all the other data packets enter the data queue corresponding to the appropriate user.

We have gathered all the packets being retransmitted in a single queue, which we call the retransmission queue. The other queues contain fresh RLP data packets destined for the appropriate users. The retransmission queue is modeled with a preemptive M/G/1 system. This means that the packets in the retransmission queue receive *absolute priority*. If an arrival was to occur in the retransmission queue while the transmitter is serving one of the other queues, it immediately leaves the current job and serves the packet that arrived. Because of the priority of the packets in the retransmission queue, the transmitter serves the retransmission

queue until it is empty. Only when there is no more packets waiting in the retransmission queue, the transmitter serves the queues containing the fresh RLP data packets to the users in a round robin fashion.

Therefore, we only need to find the delay through the retransmission queue without worrying about the other queues. Because of the preemptive nature of the retransmission queue, the queues containing the packets being sent for the first time do not enter the analysis.

For the analysis, we make use of the Pollaczek-Khintchine transform equation for the M/G/1 queue. The Pollaczek-Khintchine result provides the moment generating function, $F_w(s)$, for the system delay time (i.e. waiting time plus service time, denoted by the random variable, w). It is given below.

$$F_w(s) = \frac{(1-\rho)sF_x(s)}{s-\lambda[1-F_x(s)]}$$

$F_x(s)$ is the moment generating function for the service time, λ is the interarrival rate of the retransmitted packets entering the queue, and ρ is the system utilization factor. The values of the parameters that are relevant to our analysis are shown below.

$$\lambda = 50 \text{ pkts/sec}$$
$$\overline{X} = 3.94 \text{ ms (see } p\text{mf)}$$
$$\rho = \lambda\overline{X} = 0.197$$

From this moment generating function, we can compute the probability distribution of the system time. Numerical methods were used to obtain the probability distribution of the delay for the retransmission queue.

At this time, we introduce the service time characteristics for the retransmission queue. The retransmission of a packet is requested in a NAK, and the retransmission queue is responsible for queuing and transmitting the requested packet. In transmitting the packet, the service time of the transmitter represents the time during which the transmission of a frame in the queue takes place. In our application, we consider a particular case where the service time takes a

value from seven discrete values. In particular, the service time has the following probability mass function, $p_x(x)$ [17].

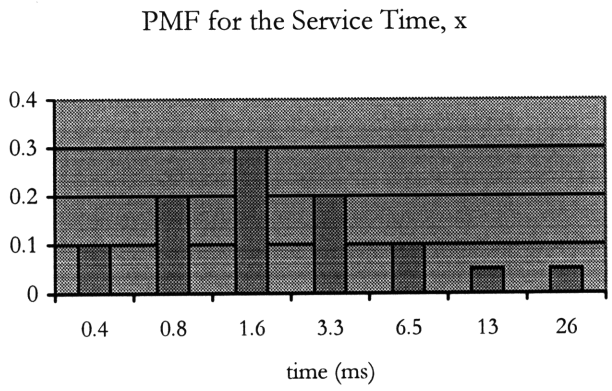


Figure 19 : The service time for the retransmission queue takes its value from one of seven possible values.

The above *pmf* results in the following moment generating function for the service time, x.

$$F_x(s) = Ae^{-X_1 s} + Be^{-X_2 s} + Ce^{-X_3 s} + De^{-X_4 s} + Ee^{-X_5 s} + Fe^{-X_6 s} + Ge^{-X_7 s}$$

The constants, A through G and X_1 through X_7, have been left as variables; their values are obtained from the probability mass function of the service time (see above). The resulting s-transform of the total system time in the retransmission queue is therefore as follows.

$$F_w(s) = \frac{(1-\rho)s[Ae^{-X_1 s} + Be^{-X_2 s} + \ldots + Ge^{-X_7 s}]}{s + \lambda(1-[As^{-X_1 s} + Bs^{-X_2 s} + \ldots + Gs^{-X_7 s}])}$$

Analytically obtained mean delay through the retransmission queue is about 5.5 milliseconds. Numerical methods are used to obtain the probability distribution of the delay for the retransmission queue. MATLAB functions are written to take as input the service time characteristics and arrival rates, and output the delay characteristics in the form of a CDF. PLOT 1 in Appendix D illustrates the CDF for our system. For a probability level of 10^{-6}, the

[17] See Appendix B for the source of $p_x(x)$.

delay is bounded by about 85 milliseconds. For $\lambda = 100$, which is illustrated in PLOT 2, the delay is observed to be about 130 milliseconds for probability level 10^{-6}.

What happens if the retransmission queue were non-preemptive? Let us assume that an arrival into the retransmission queue occurs while the transmitter is sending a data packet at the slowest rate possible. Therefore, the system delay through the non-preemptive retransmission queue will be increased by this longest possible service time of a data packet. For the particular system and system parameters that we are interested in, the longest possible service time is 26 milliseconds (see Figure 19). Therefore, the upper bound for the M/G/1 delay is 111 milliseconds with probability 10^{-6}. However, this does not increase the overall system delay to the point where more sequence number bits are needed to account for the increased delay due to the non-preemptive nature of the queue. For a different service time distribution or a different queue utilization factor, the added delay due to the non-preemptive nature of the system mat be significant, and this should be included in the analysis.

In conclusion, this chapter explored the analytical methods to obtain the upper bound delay through the queuing subsystems in the two-queue model. For the receive queue, the M/D/m model was decoupled into multiple E/D/1 queues, which allowed the analysis of a single E/D/1 queue to fully understand the behavior of the M/D/m system. In order to approximate delay through the E/D/1 queue, the queue size probabilities of the E/M/1 queue were utilized. This resulted in a somewhat loose bound, and simulations in Chapter 7 will quantify how loose a bound it is. For the retransmission queue, a pre-emptive M/G/1 queue was used to model the queue that received packets that are to be retransmitted. The Pollaczek-Khintchine transform equation was used to represent the behavior of the system time. In addition, the complete delay distribution for the retransmission queue was obtained numerically. In Chapter 7, these results will be examined in view of the simulation models and their results.

Chapter 6 : Summary and Discussion of Analysis

This chapter briefly presents the summary of results that were obtained in the queuing analysis of Chapter 5. The RLP transmitting module within the CDMA base station that is responsible for transmitting a packet whose retransmission is requested in an incoming NAK is modeled in the two-queue model. The two-queue model consists of a queuing module responsible for receiving and processing of the NAK and a queuing module responsible for actually transmitting the packet being retransmitted. The data path that the model depicted is the RLP frame round trip delay – the delay from the receipt of the NAK at the base station to the retransmission of the packet that the NAK requested. The quantitative analysis of this round trip delay is important because it directly influences the size of the sequence number space in the RLP layer. Because of the nature of the RLP's retransmission strategy and absence of a windowing scheme, a sequence number may remain valid for quite some time and possibly until the sequence number wraps around. Therefore, care must be taken to insure that the wrap around occurs after the round trip delay of a NAK frame.

The two-queue model is developed to model the RLP round trip delay data path. Within the two-queue model, the receive queue is modeled with an M/D/m queue, and the retransmission queue with an M/G/1 queue. The system delay through the two queues is analyzed and the upper bound delay is obtained for an error probability level of 10^{-6}. The upper bound delay through receive queue is approximated to be about 75 milliseconds, which is thought to be conservative. As will be seen later from simulations, the delay is greater than about 9 milliseconds with probability 10^{-6}, and the delay through the retransmission queue is greater than 82 milliseconds with the same probability. The sum of the two delays constitutes the upper bound of the frame round trip delay. Given this delay value and the highest transmission rate that the base station transmitter can tolerate – about 2500 packets/second, we obtain the number of bits in the sequence number that would lower the probability of the RLP failure, due to the sequence number failure, to 10^{-6}. Our study shows that nine sequence number bits will do the job. Consider the chart below.

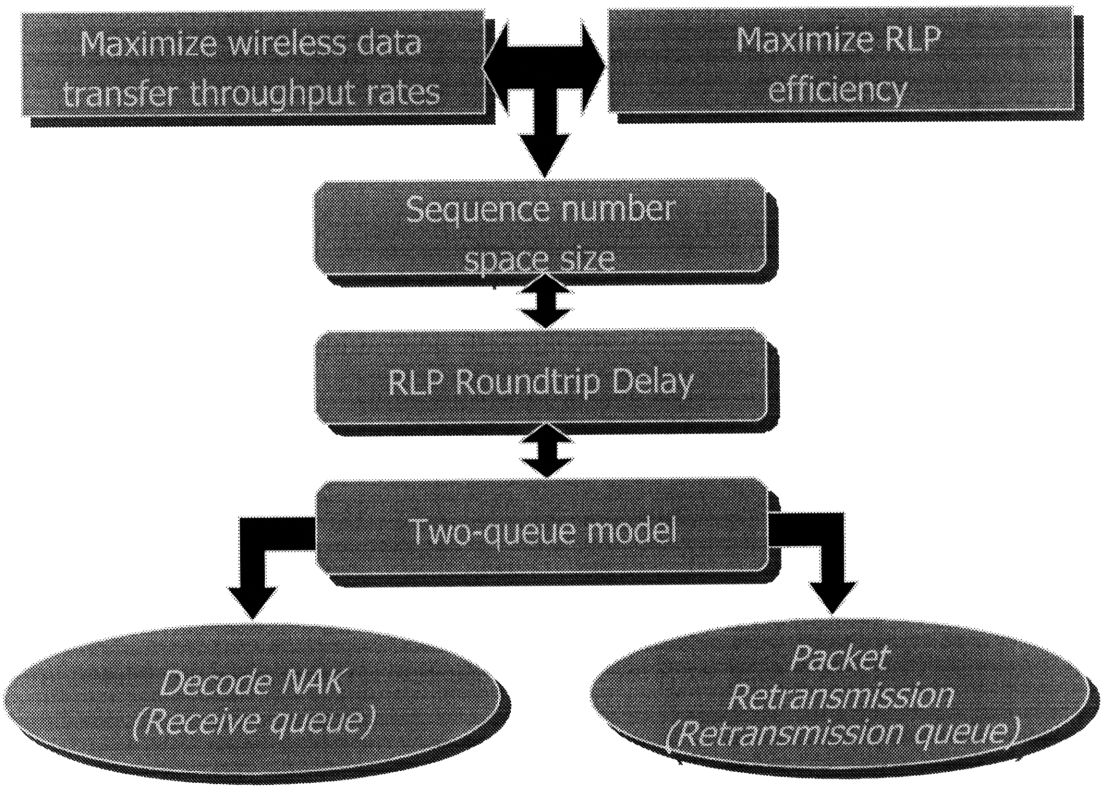


Figure 20 : This flowchart provides the context in which this study originates and where the two-queue model comes from and illustrates the two components of the model.

The chart above lays out the relationships between the different performance issues at hand as well as how they are linked to the central topics of this study. Initially, the number of sequence number bits suggested for this system was 24 bits. The result that only nine sequence number bits will be sufficient indicates that the amount of overhead can be safely reduced by more than 50 percent. This also has other implications. The smaller ratio of the overhead bits to the data bits means higher effective data throughput in the RLP layer, and it indicates higher performance of the overall CDMA system. Figure 20 shows how the performance issues are related to each other and to the primary goals of this thesis. It also illustrates the origin of the two-queue model and its components.

So, what have we done? We have taken the relevant parts of the CDMA base station RLP transmitter and modeled it to quantify the frame round trip delay. In doing so,

1. we created a way of determining the number of the sequence number bits to bring the probability of protocol failure to an arbitrarily low level. We chose 10^{-6} for our analysis.

2. in understanding the behavior of the M/D/m queue, we found a functionally equivalent system, namely the $E_m/D/1$ queue, that we can analyze to completely describe the behavior of the former queuing system.

3. obtaining the complete probability distribution of the delay the M/G/1 retransmission queue. This was done numerically.

These are the major contributions of this thesis project. We came up with a methodology to quantify the upper bound round trip delay for an RLP NAK for an arbitrarily chosen probability level. In doing so, we also found some interesting points about the behavior of a queuing system with exponential interarrival time, constant service rate, and multiple servers.

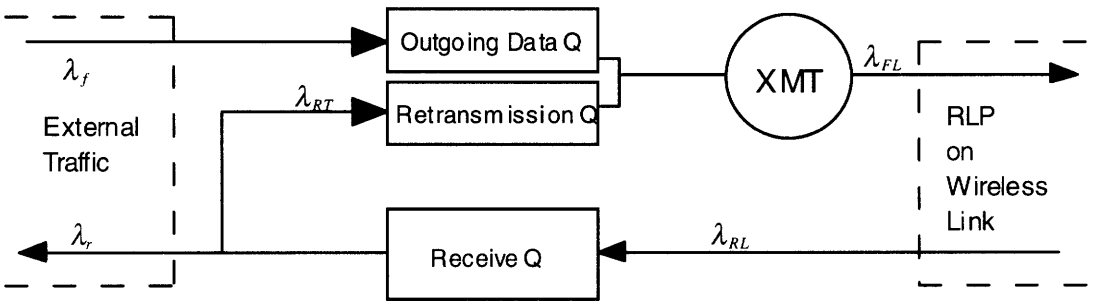
We must not forget about the assumptions that we made that led us to these results. For example, the arrival rates into our two-queue model were assumed to be Poisson. Whether or not the arrivals are independent of each other should be looked into to assess the validity of our model. We also did not consider the possible bursty nature of the RLP erasures. We assumed that the errors that occur in the RLP layer happen independently. In addition, we assumed that the NAKs that travel from the mobile station to the base station are error-free.

We state that the results from our analysis were obtained with these assumptions about the system and the RLP layer.

Chapter 7 : Simulations

The previous chapters have discussed the background and motivation behind the study, the modeling procedure, and the analysis as well as its results of the two-queue model of the RLP transmitter. The motivation was to quantitatively analyze the round trip delay of an RLP frame, ultimately arriving at a reasonable sequence number space size that would prevent its wrap around at inopportune instances and keep the probability of protocol failure acceptably low. These topics are inevitably related to the performance and efficiency of the RLP layer.

The analytical model was simulated using OPNET, a communications networks simulation software. This chapter outlines and discusses the work done with OPNET and the results of the simulations as well as their meaning and contributions to this thesis project. Figure 21 below illustrates the system that was simulated. Following the figure, we include the system parameters that were used in our work.



λ_f(forward data traffic) =450 pkts/sec

λ_{RL}(reverse link traffic) =1000 pkts/sec $=0.111\lambda_f + \lambda_r$

λ_{RT} (retransmission traffic) $=0.111\,\lambda_f$

λ_r = reverse data traffic

λ_{FL}(forward link traffic) $=\lambda_f + \lambda_{RT} = 1.111\,\lambda_f$

forward link erasure rate = 0.1

reverse link erasure rate = 0

The M/D/m Queue – The Receive Queue Parameters[18]

$$\lambda_{RL} = 1000 \text{ pkt/s}$$
$$\mu = 200 \text{ pkt/s}$$
$$D = 0.005 \text{ s/pkt}$$
$$m = 10$$
$$m\mu = 2000 \text{ pkt/s}$$
$$\rho = \frac{\lambda}{m\mu} = 0.5$$

The M/G/1 Queue – The Retransmission Queue Parameters

$$0.111\lambda_f = \lambda_{RT} = 50 \text{ pkt/s}$$
$$\overline{X} = 0.00394 \text{ s/pkt}$$
$$\rho = \lambda\overline{X} = 0.197$$

Figure 21 : Diagram of the simulated system including the list of parameter values for the individual queuing systems.

[18] Typical utilization for Qualcomm's test system.

E/D/1 vs. E/M/1

OPNET was used to observe how the E/D/1 performed compared to the E/M/1 queue. Models of the E/D/1 and the E/M/1 were constructed using the OPNET modeling components and the delay statistic was chosen and plotted. PLOT 3 and PLOT 4 in Appendix D of this document illustrate the CDF's of the delay through the E/D/1 and the E/M/1 queues with comparative system parameters, respectively. For the E/M/1 queue, the upper bound delay is about 90 milliseconds with probability of exceeding it 10^{-6}, whereas, for the E/D/1, the upper bound delay is about 9 milliseconds with the same probability. According to the CDF's, the E/D/1 queue does better in delivering the shorter queuing delay. The zero variance of the service time is the main reason for this. For the E/D/1 queue, the smallest delay you expect is the service time, which would occur if there were no contributions from the queuing delay. For the E/M/1 queue, the exponential distribution of the service time pushes the curve to the right, causing the delay at which the quantity, $p_w(w > w_0) \approx 10^{-6}$, at a much later point. A G/G/1 upper bound for the waiting time states that the upper bound waiting time for the E/D/1 queue is less than that for the E/D/1 queue, due to the zero variance of the service time [2]. This is confirmed by the simulation results.

The behavior of the queue size for the E/D/1 and E/M/1 were also obtained and plotted. This was done in order to confirm the validity of obtaining the bound on the delay characteristics of the E/D/1 queue with the occupancy probabilities of the E/M/1 queue. The simulation was run for 10,000 seconds (about 10^7 packet transmissions) and the cumulative probability functions were obtained from the simulation data. PLOT 5 (E/D/1) and PLOT 6 (E/M/1) illustrate the queue size probabilities CDF's. As we expected, the queue size of the E/D/1 queue is more concentrated around the lower states than the E/M/1 queue; the average queue length (in number of packets) is smaller for the E/D/1 queue. The behavior of the analytically obtained occupancy probabilities for the E/M/1 matched the one obtained from the simulation quite well. The queue size for which the probability of having a larger queue is 10^{-6}, was found to be 11 from the simulation data. The corresponding number was found to be 14 from the analytically found result. Comparing these two results, the estimate for the upper bound delay though the receive queue is 60 milliseconds vs. 75 milliseconds. With either result, we find that the delay estimate based on the occupancy

probability of the E/M/1, $p_n^{E/D/1}(n)$, is an overestimate. From the simulation data, we find that the queue size of the E/D/1 is larger than 2 with probability 10^{-6}. Based on this result, the upper delay is found to be 15 milliseconds. Comparing the two delays, 15 and 60 milliseconds, we again find that the estimate is very conservative. We also note that for the both queues, the analytical results gave more conservative results than the simulation findings.

Finally, we go back to the original M/D/m model of the receive queue and comment on the validity of our simplification of the problem. Recall that we had simplified the M/D/m queue into m E/D/1 queues. Each E/D/1 took every m^{th} frame of the arrival that would have entered the M/D/m queue. PLOT 7 plots the delay CDF of the M/D/m queue with the corresponding parameter values as those used for the E/D/1 queue model. The delay characteristics are the same. Comparing it with PLOT 3, we observe that the CDF curves start at the same place and reach their limit at around the same value of delay. This confirms the validity of our simplifying transform of the M/D/m queue into many E/D/1 queues in modeling the receive queue. The mean delay as well as the upper bound delay were found to be the same.

Sensitivity to the Number of Servers

We observe that the behavior of the M/D/m queue (or, the E/D/1 queue) is sensitive to the value of m, the number of servers (or, the order of Erlang arrivals). We first qualitatively reason through how the system would behave for different m, and we show that the simulation results verify the reasoning. We can equally deal with either the M/D/m or the E/D/1 because they are functionally equivalent. We choose the queue that provides the easier way to understand the particular behavior.

For the E/D/1 queue, the parameter, m, is the order of the Erlang arrivals. As m gets large, the interarrival time also becomes large, and a frame is more likely to see an empty queue upon arrival. If the service rate is kept constant, the frames are going to experience no waiting time as m gets very large. This suggests the queuing system approaches a D/D/1 queue for very large m, which would result in a constant delay.

Simulations were performed, keeping the utilization factor the same, $\rho = 0.5$. The service rate was altered to keep ρ constant for the changing values of m. The delay CDFs for values of m =[2, 5, 10, 20] are plotted in PLOT 8.

As m is increased from 2 to 20, the CDF approaches that of a deterministic distribution. Each case was run for 3000 seconds which is equivalent to about $10^5 - 10^7$ events. The number of servers and the service rate were altered, and the total arrival rate into the M/D/m and ρ were kept the same. It appears that even if we keep the utilization factor the same, it is more desirable to increase the number of servers rather than to increase the rate of each server. From a designer's point of view, more servers may mean higher cost and more physical space, but the relationship between number and speed may not be linear. Lastly, it is important to be able to go back and forth between the M/D/m and E/D/1 queues to understand any particular queue behavior because of their functional equivalence.

Sensitivity to the Arrival Rate

We shall denote the arrival rate into the M/D/m queue as $\lambda_{RL} (= 0.111\lambda_f + \lambda_r)$. Therefore, the arrival rate into the E/D/1 queue would be λ_{RL}/m. We briefly consider the possibility of an increase in λ_{RL} due to an increase in reverse data traffic, λ_r. First, we consider the M/D/m queue. As the average arrival rate becomes large, the average interarrival time, $1/\lambda_{RL}$, the standard deviation of the interarrival time, $1/\lambda_{RL}$, decreases.

For the E/D/1 queue, let us consider the increase in the incoming traffic as well as the order of Erlang, so that the parameter, λ_{RL}/m, does not change (i.e. $m = K\lambda_{RL}$). The standard deviation of the E/D/1 queue's interarrival time, $\frac{\sqrt{m}}{\lambda_{RL}} = \frac{\sqrt{K}}{\sqrt{\lambda_{RL}}}$, decreases as λ_{RL} increases.

Therefore, while the average interarrival time remains constant, the standard deviation as well as the variance decreases as λ_{RL} and m increase. The arrivals look more and more deterministic. This is the result of the law of large numbers. Thus, with a deterministic service

rate, the queue approaches a D/D/1 queue. The simulation results in PLOT 8 verify this conclusion.

The M/G/1 Queue

The simulation results provide a way to evaluate the analytic results and to spot any issues that were overlooked in the analysis. OPNET provided a mechanism to compare the performance of the system model with that of different types of queues, which enhanced the understanding of the system of interest.

For the M/G/1 queue, which modeled the retransmission queue, simulations were also performed to validate our analytic model. We built the M/G/1 simulation model with the system parameters presented previously. In order to simulate the different possible packet service times with corresponding probabilities, we assumed a constant service rate and varied the packet size according to the service time probability distribution. Specifically, a service rate of 1,000,000 bps was chosen, and the packet size of the following distribution was used.

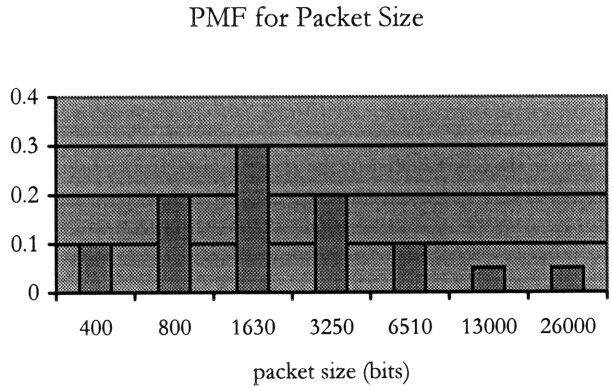


Figure 22 : The probability distribution for the packet size. For the simulation of the retransmission queue, a constant service rate was assumed and the service time probability distribution was converted a probability distribution for the packet size.

Simulations were run to obtain the mean delay and the upper bound delay through the retransmission queue system for the parameters in Figure 21. The average delay through the

system was found to be about 5 milliseconds. PLOT 9 shows the cumulative density function for the delay. Looking at the trace data, the delay for which the probability of exceeding it is less than 10^{-6}, is about 80 milliseconds. By doubling λ_f, an arrival rate into the retransmission queue of 100 *packets/second* was simulated (see PLOT 10). The mean delay was found to be 8 milliseconds and the upper bound delay 125 milliseconds with probability of 10^{-6}. By doubling the arrival rate, the mean delay and the upper bound delay only increased by 50 percent. Because of the presence of the feedback loop in the system, changing the rate into the retransmission queue would ultimately change the overall delay through the system depicted in Figure 21.

The delay (both mean and upper bound) for the retransmission queue is found to be significantly greater than the delay through receive queue even though the arrival rate is a lot greater for the receive queue and the average service time is a smaller for the retransmission queue. The reason behind this can be traced to the number of servers and the service characteristics. In the sensitivity analysis, described in sections **Sensitivity to the Number of Servers** and **Sensitivity to the Arrival Rate**, we have determined that the system with more decoders perform better than the system with less decoders, assuming everything else is the same. We can see this in effect in comparing the two components in the two-queue model. Even with the different arrival rates, the queue with multiple servers performs better. Another difference between the two queues is the service characteristic. One has a constant service time while the other has a non-constant service time distribution. We can see that the system with a constant service rate performs better. This can easily be seen from examples of M/M/1 *vs.* M/D/1.

The simulation runs of the model greatly enhanced the understanding of the system behavior. We extracted the mean delays as well as the upper bound delays through the two-queue model, and we were able to vary a few system parameters to see the sensitivity of the system to these parameters.

Chapter 8 : Conclusions and Future Work

For this thesis project, the Radio Link Protocol layer of a wireless CDMA channel was examined. Specifically, the objective was to quantify the RLP frame delay characteristic as well as the sequence number space size requirements. In order to do so, a model of the base station RLP transmitter was constructed to characterize the data path of interest. The data path of interest was defined by the frame round trip delay; the path starts at the mobile station when it sends a NAK for a missing frame. The NAK is received at the base station, enters the receive queue, and eventually decoded. Once the NAK is decoded, the frame whose retransmission is requested is placed on the retransmission queue and finally transmitted. This data path is depicted in the two-queue model; it consists of the receive queue and the retransmission queue. The two-model utilizes the M/D/m and the M/G/1 queues to depict the round trip data path.

The queuing analysis of the two-queue model allowed us to quantify the delay. To simplify the M/D/m queue, it was broken down into m E/D/1 queues. The analysis of a single E/D/1 queue was capable of capturing the behavior of the M/D/m system. However, E/D/1 was very analytically difficult to solve. Therefore, only a bound was found on the E/D/1 delay. The M/G/1 queue was analyzed to obtain the mean delay and the upper bound delay for an

arbitrarily low probability that the delay will exceed it. With an error probability of 10^{-6}, the delay through the two-queue model system was found to be 90 milliseconds.

With the results involving the delay characteristics of the two-queue model, the issue of sequence number space was explored. An adequate size of the sequence number space of nine bits was chosen based on the delay and the RLP retransmission strategy that would reduce the probability of the protocol failure to about 10^{-6}. In case of a failure, the protocol would be reset and the recovery of the missing data frames would be left for the upper protocol layers to take care of. Nine sequence number bits is much less than the 24 bits which was initially suggested for such a packet data system. This decrease in the sequence number bits indicates savings in overhead and cost as well as higher effective data throughput. Higher effective data throughput in the RLP layer ultimately affects the overall efficiency of the system.

This project looked at some specific issues of the RLP layer. It examined the frame delay characteristics based on a queuing model and eventually made suggestions on the size of the sequence number space with which the wrap around would not occur at inopportune times.

A great deal was learned about system modeling, queuing analysis, and simulation data interpretation. There are a number of things during the analysis and simulation phases of the research effort that we could have done better; some of these issues were mentioned in the previous chapters, and the others are included in the list of possible future works.

Finally, we make a few suggestions about possible future work. The list below describes the suggestions.

- Model the priority-based non-preemptive M/G/1 queue to determine how different the delay characteristics would be with a non-preemptive scheme.

- Analytically find a tighter bound on the delay through an E/D/1 queue and perhaps arrive at analytical solution for the delay or the state probabilities.

- Explore and quantify the sensitivity of the analysis presented here to the change in system parameters other than number of servers and arrival rates.

- Model the retransmission queue to be insensitive to the scheduling algorithm and study the merits of different scheduling algorithms to optimize performance.

- Reexamine and challenge the validity of some of the assumptions made in our analysis. Such are the independent arrival characteristics into the queuing systems and also the independent frame erasure rate in the RLP layer.

Appendix A : Values of Parameters of the Two-Queue Model

The M/D/m Queue – NAK Decoding

$\lambda_{RL} = 1000$ pkt/s

$\mu = 200$ pkt/s

$D(= \overline{X}) = 0.005$ s/pkt

$m = 10$

$m\mu = 2000$ pkt/s

$\rho = \dfrac{\lambda}{m\mu} = 0.5$

$packet = 1000$ bits

The M/G/1 Queue – Retransmission of the Packet

$\lambda_f = 450$ pkt/s

$\lambda_{RT} = 50$ pkt/s

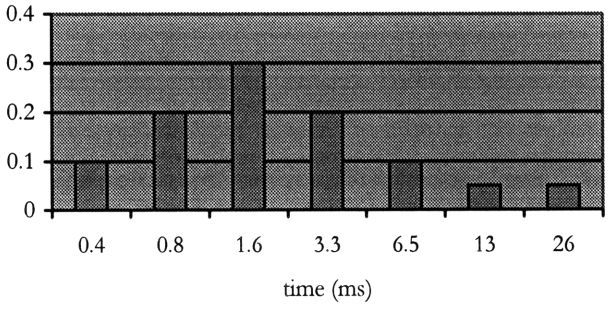
$\overline{X} = 0.00394$ s/pkt

$X =$ see chart below

$\rho = \lambda\overline{X} = 0.197$

$packet = 1000$ bits

PMF for the Service Time, x



Appendix B : Service Time Characteristics

The probability distribution for the retransmission queue's (modeled with a M/G/1 queue) variable data rates was obtained from Qualcomm, Inc. The information is for an actual wireless CDMA system, although in this project, we ignore the details of the scheduling algorithm. The figure below illustrates the probability distribution function.

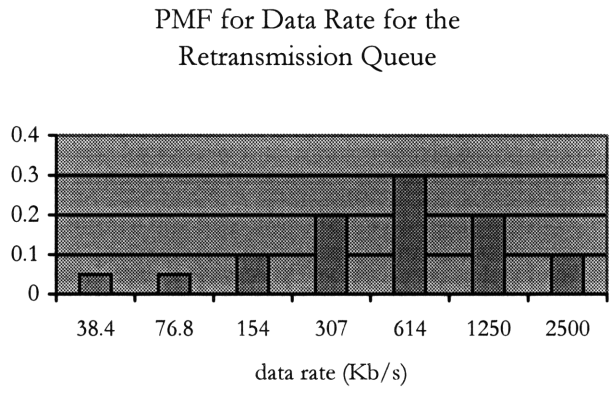


Figure 23 : Probability distribution for the variable rate tolerated by the retransmission queue transmitter.

The base station consults this *pmf* upon transmitting each frame. Therefore, the above probability distribution shows the percentage of frames that are sent at a particular data rate. Upon sending a frame, the base station performs a rate negotiation process with the mobile station of interest to determine the data rate for the particular frame.

Assuming 1000-bit frames, units of the abscissa axis is also *frames/second.* From this, we obtain the *pmf* for the frame service time by inverting each of the service rates. We obtain the mean service time by computing the expected value of the below *pmf.*

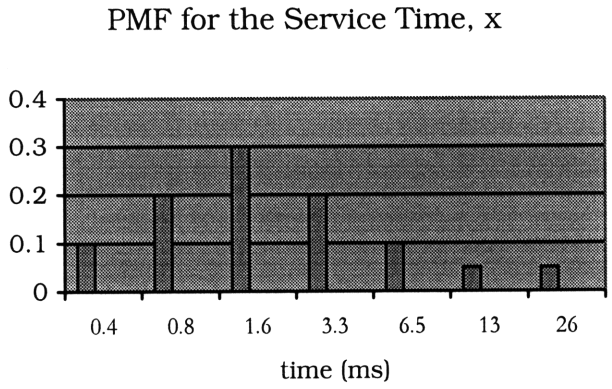


Figure 24 : The probability distribution for the service time of the retransmission queue.

Appendix C : Matlab Code

invert.m

```
function prob=invert(p,lambda,mag,n)
% obtain pdf excluding the impulse at w=0, weight (1-rho)
% assuming the stepsize is sufficiently small
% p: prob. vector for service time (discrete), prob;time
% lambda: poisson average arrival rate
% mag: magnification factor
% n: maximum queue length

ll=lambda/mag;
t=1;
n1=floor(p(2,1)*mag+.5);
tt(1:n1)=ll;

for i=2:size(p,2),
t=t-p(1,i-1);
n1=ceil(p(2,i-1)*mag+.5);
n2=floor(p(2,i)*mag+.5);
tt(n1:n2)=t*ll;
end;

tp=1;
pp=zeros(1,n*size(tt,2)-n+1);
for j=2:n,
tp=conv(tp,tt);
a=size(tp,2);
pp(1:a)=pp(1:a)+tp;
end;

r=(1-lambda*p(1,:)*p(2,:)');
prob=r*pp;
```

cdf.m

```
function prob=cdf(pmf)
prob(1)=pmf(1);
for i=2:size(pmf,2)
```

```
prob(i)=pmf(i)+prob(i-1);
end
```

chuck.m

```
function res=chuck(p,lambda,n)
% p: prob. vector for service time (discrete), prob;time
% lambda: poisson average arrival rate
% n: maximum queue length

tmp(1,1)=1-lambda*p(1,:)*p(2,:)';
E=p(1,:).*exp(-lambda*p(2,:));

P(1,1)=sum(E);
ip=1/P(1,1);
tmp(2,1)=(1-P(1,1))*ip*tmp(1,1);

J=1.0;
a=p(2,:);

for i=1:n-2,
J=J*lambda/i;
P(i+1,1)=J*E*a';
tmp(i+2,1)=ip*(tmp(i+1,1)-P(i+1,1)*tmp(1,1)-
fliplr(tmp(2:i+1,1)')*P(2:i+1,1) );
a=a.*p(2,:);
end

PP=P;
res=tmp;
```

chuck1.m

```
function [res,n]=chuck1(p,lambda,thresh)
% p: prob. vector for service time (discrete), prob;time
% lambda: poisson average arrival rate
% thresh: tail probability threshold to stop computation

tmp(1)=1-lambda*p(1,:)*p(2,:)';
E=p(1,:).*exp(-lambda*p(2,:));

P(1)=sum(E);
ip=1/P(1);
tmp(2)=(1-P(1))*ip*tmp(1);

J=1.0;
a=p(2,:);
i=0;
check=1-tmp(1)-tmp(2);

while (check>thresh)
J=J*lambda/++i;
```

```
P(i+1)=J*E*a';
tmp(i+2)=ip*(tmp(i+1)-P(i+1)*tmp(1)-
fliplr(tmp(2:i+1)')*P(2:i+1) );
check=check-tmp(i+2);
a=a.*p(2,:);
endwhile;

res=tmp;
n=i+2;
```

Appendix D : Numerical and Simulation Results

See the following pages for the plots.

PLOT 1 : Analytically obtained CDF of system delay for the retransmission (M/G/1) queue for $\lambda = 50$.

PLOT 2 : Analytically obtained CDF of system delay for the retransmission (M/G/1) queue for $\lambda = 100$.

PLOT 3 : CDF of system delay for the receive queue (E/D/1) from simulation.

PLOT 4 : CDF of system delay for the E/M/1 queue from simulation.

PLOT 5 : CDF of queue size for the receive queue (E/D/1) from simulation.

PLOT 6 : CDF of queue size for the E/M/1 queue from simulation.

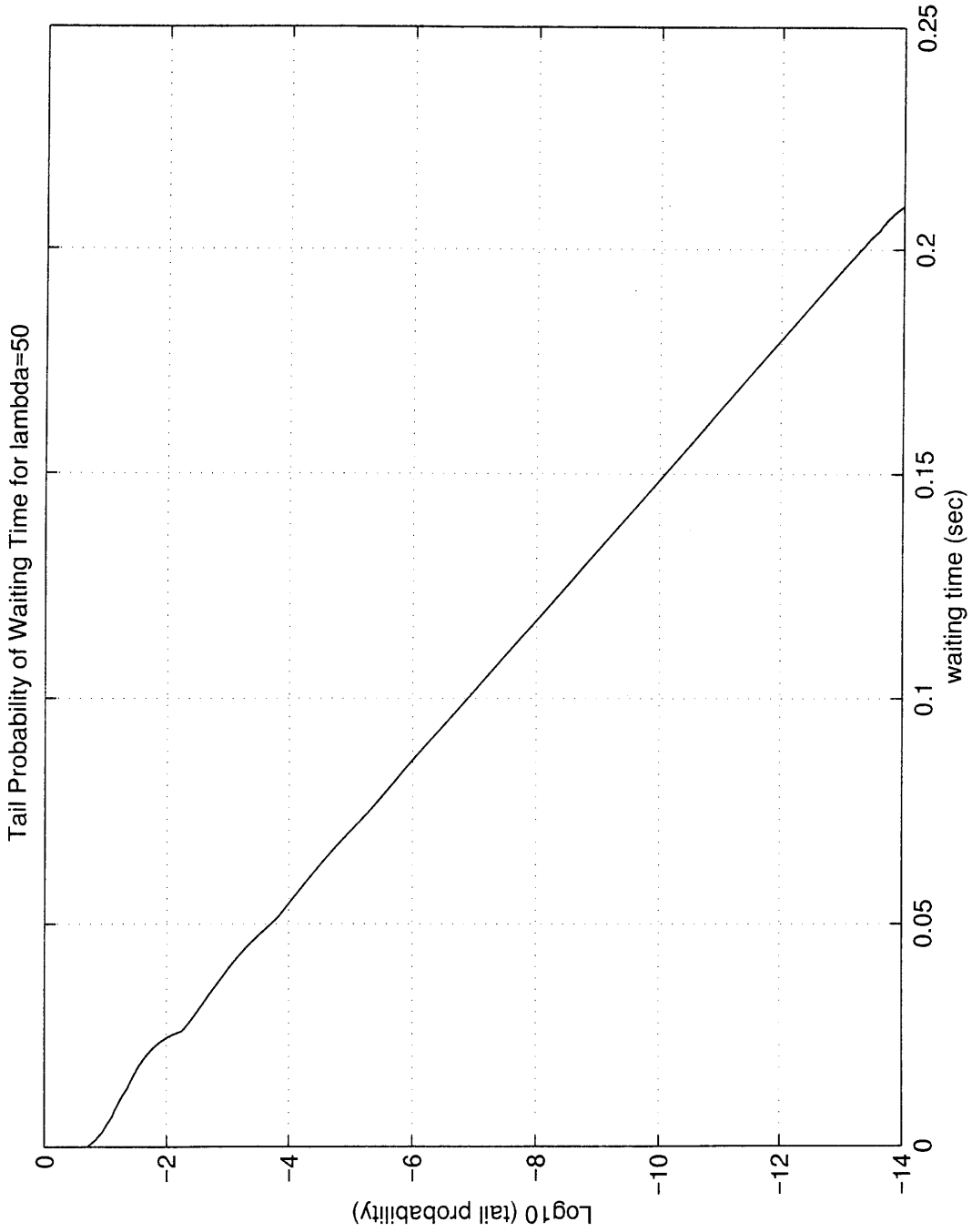
PLOT 7 : CDF of system time for the receive queue (M/D/m) from simulation.

PLOT 8 : CDF of system delay for the receive queue (E/D/1) for m =2, 5, 10, 20 from simulation.

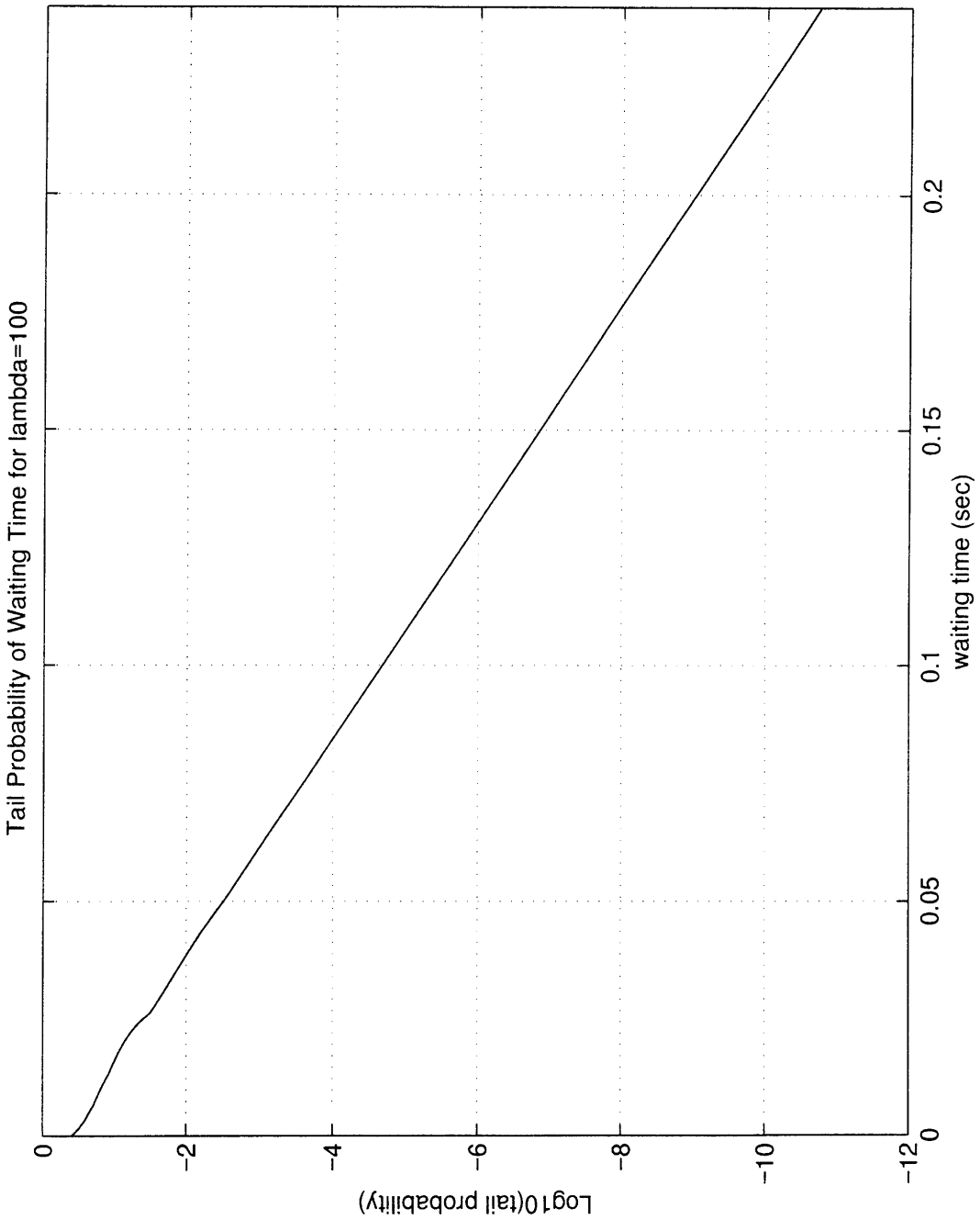
PLOT 9 : CDF of system delay for the retransmission queue (M/G/1) for $\lambda = 50$ from simulation.

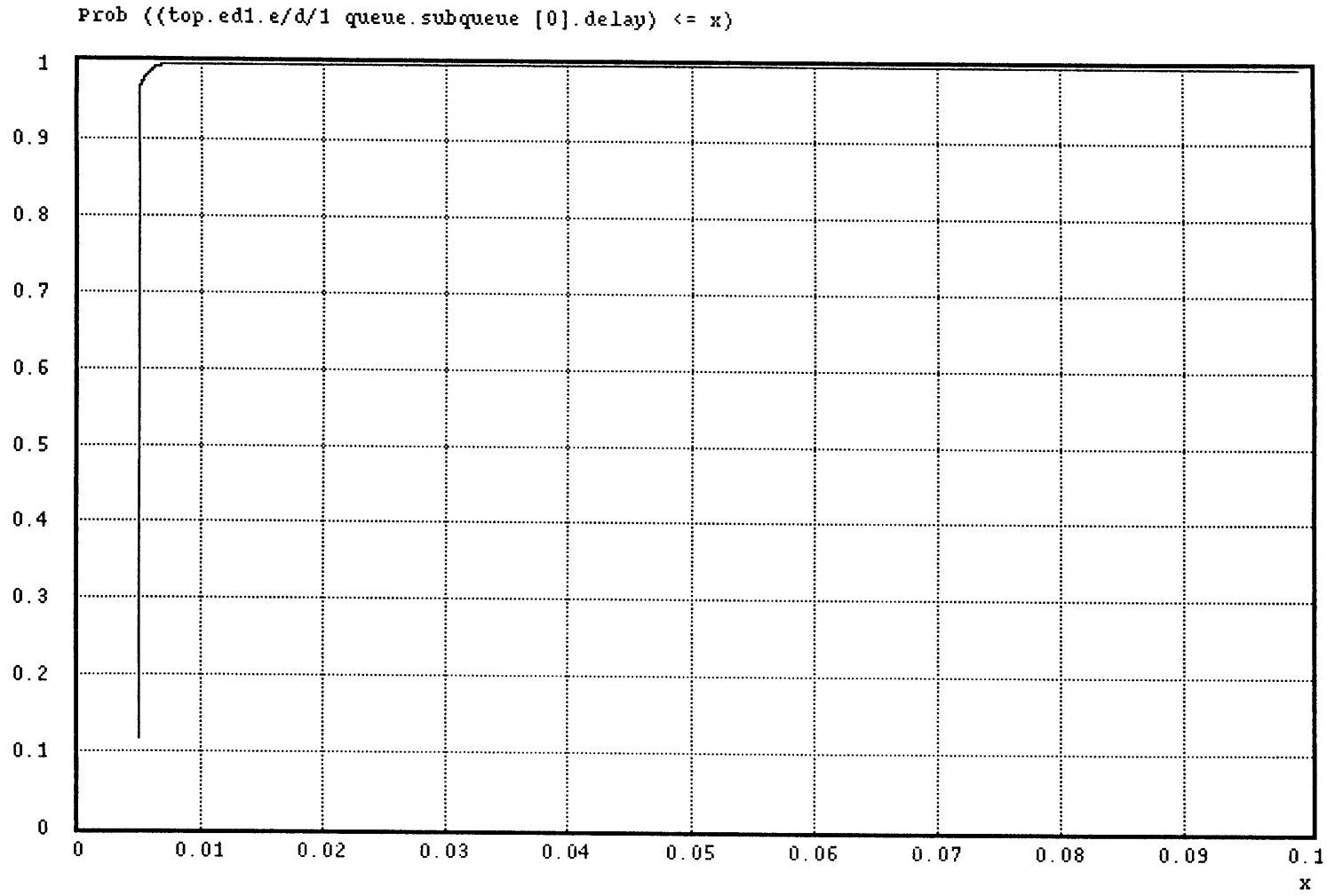
PLOT 10 : CDF of system delay for the retransmission queue (M/G/1) for $\lambda = 100$ from simulation.

PLOT 1

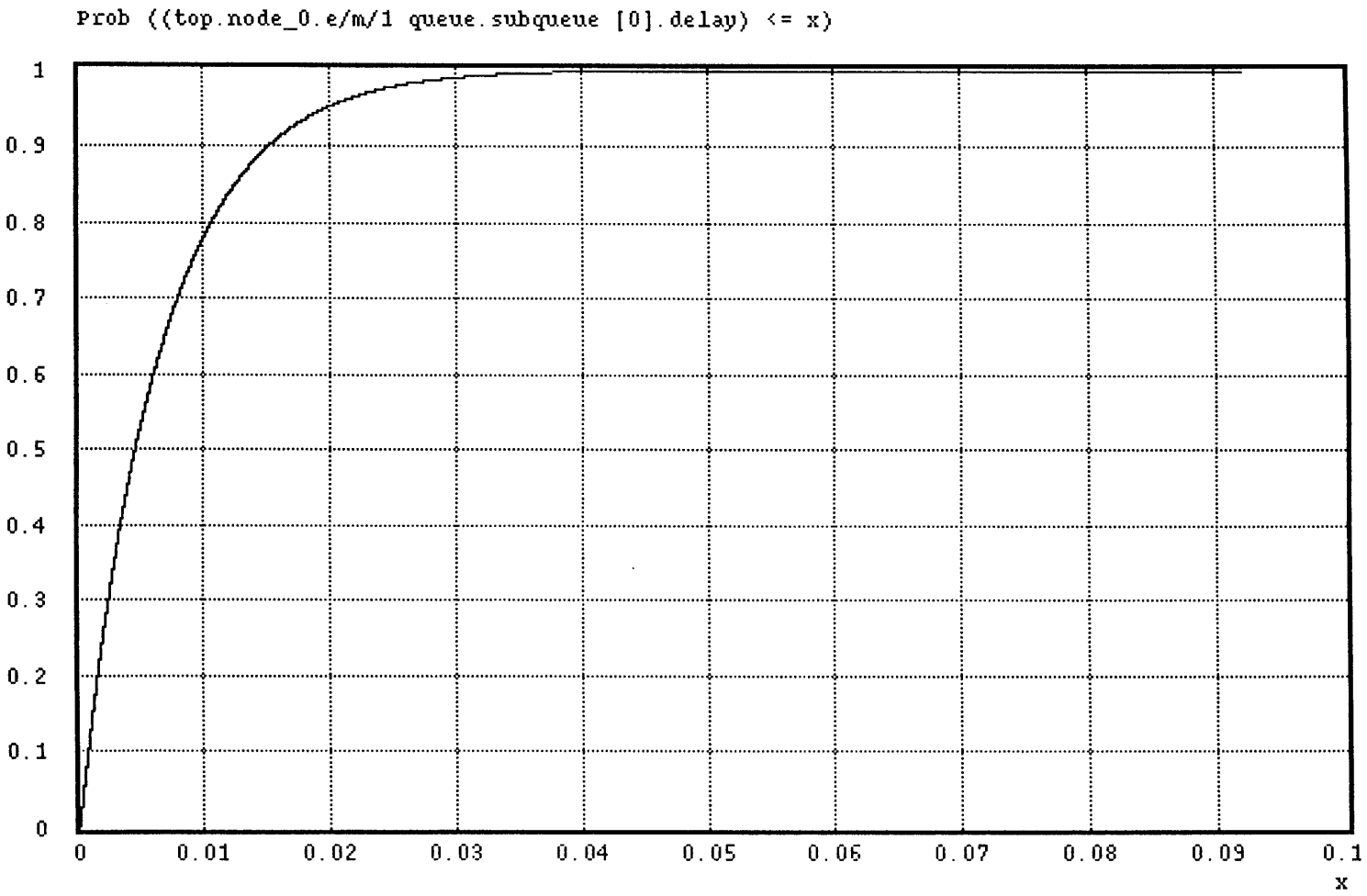

Tail Probability of Waiting Time for lambda=50
Log10 (tail probability)
waiting time (sec)
0
-2
-4
-6
-8
-10
-12
-14
0
0.05
0.1
0.15
0.2
0.25

PLOT 2

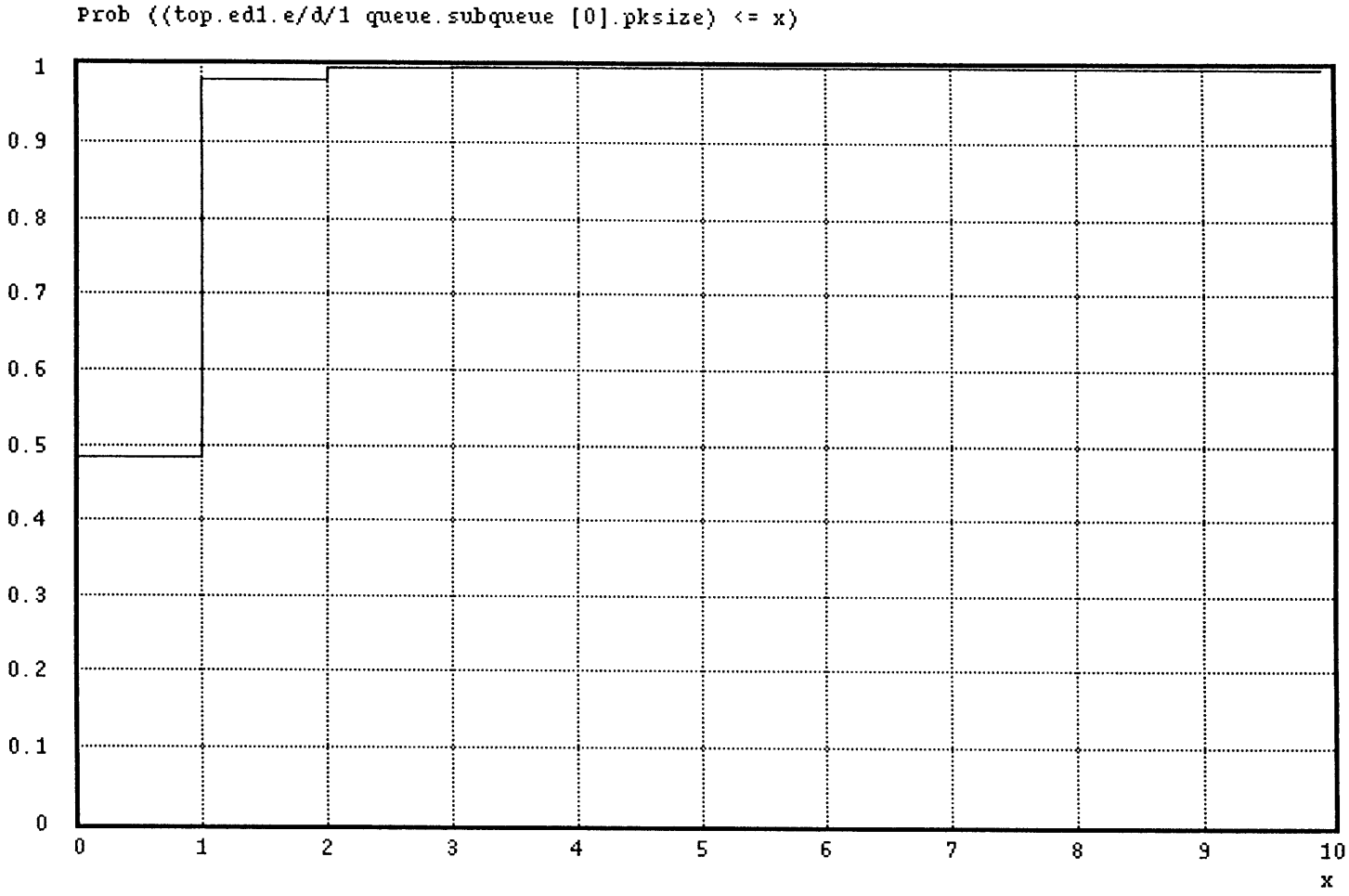

Tail Probability of Waiting Time for lambda=100
Log10(tail probability)
waiting time (sec)
0
-2
-4
-6
-8
-10
-12
0
0.05
0.1
0.15
0.2


Prob ((top.ed1.e/d/1 queue.subqueue [0].delay) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
x

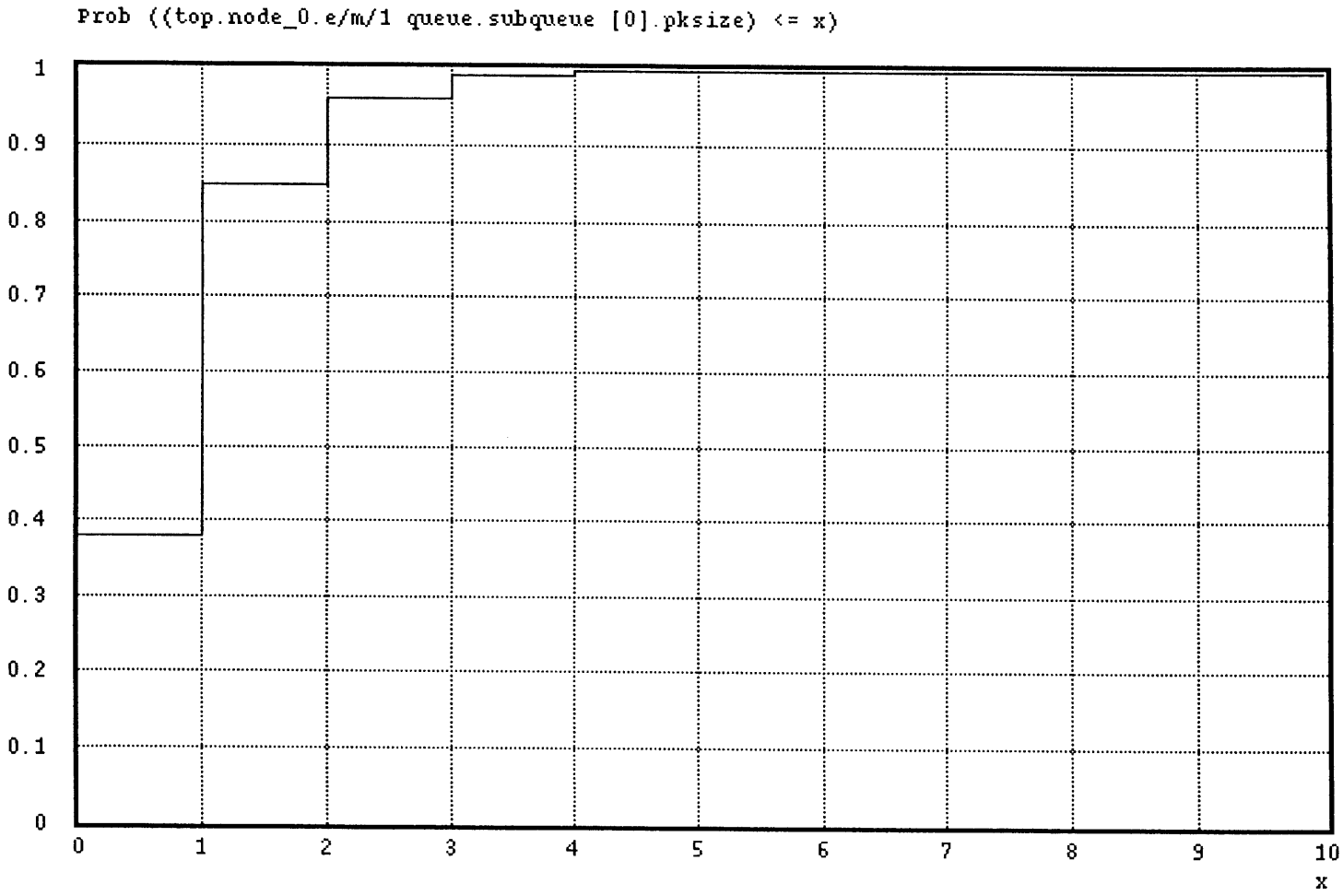
PLOT 3


Prob ((top.node_0.e/m/1 queue.subqueue [0].delay) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
x

PLOT 4

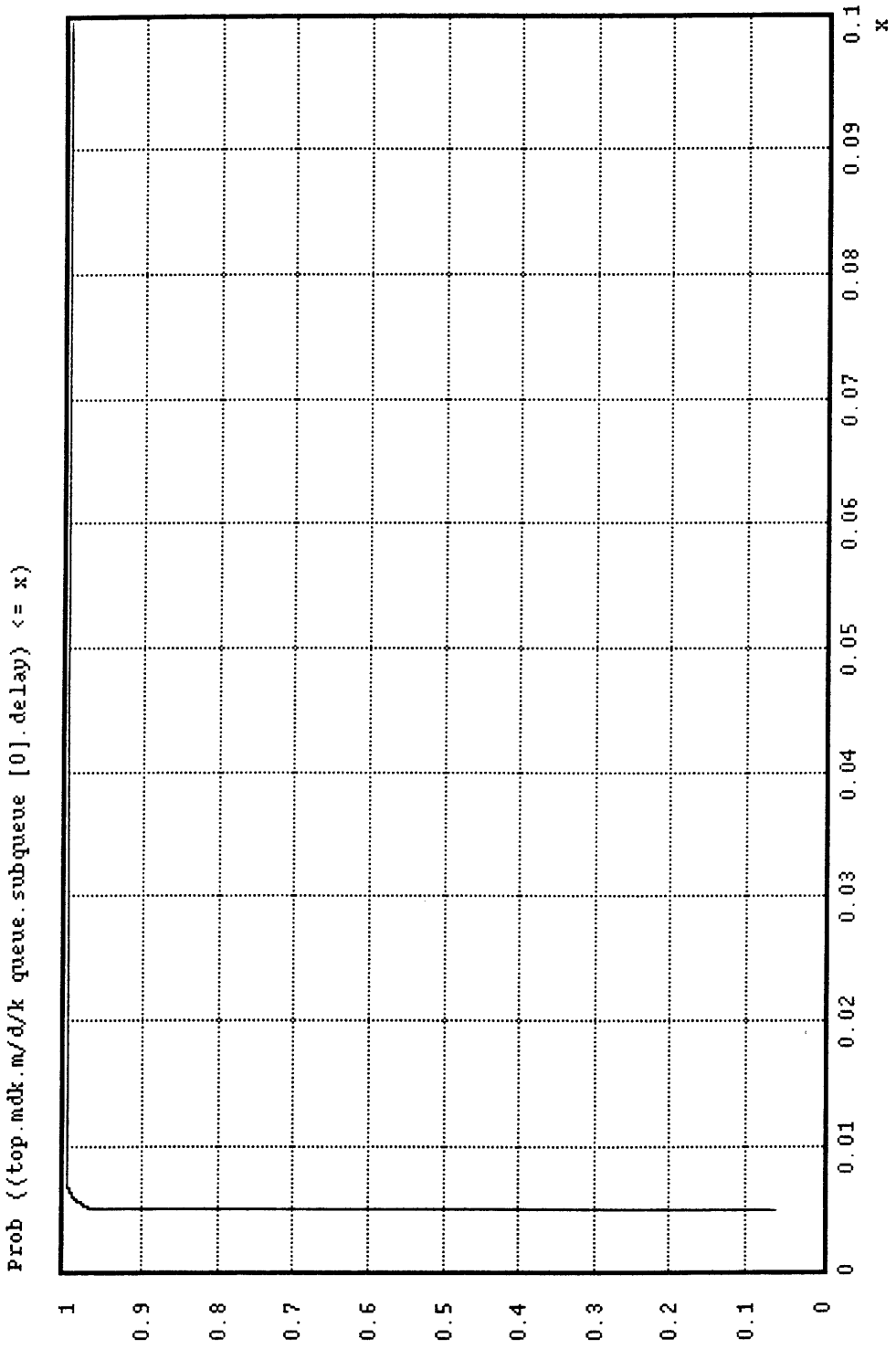

Prob ((top.ed1.e/d/1 queue.subqueue [0].pksize) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
1
2
3
4
5
6
7
8
9
10
x

PLOT 5

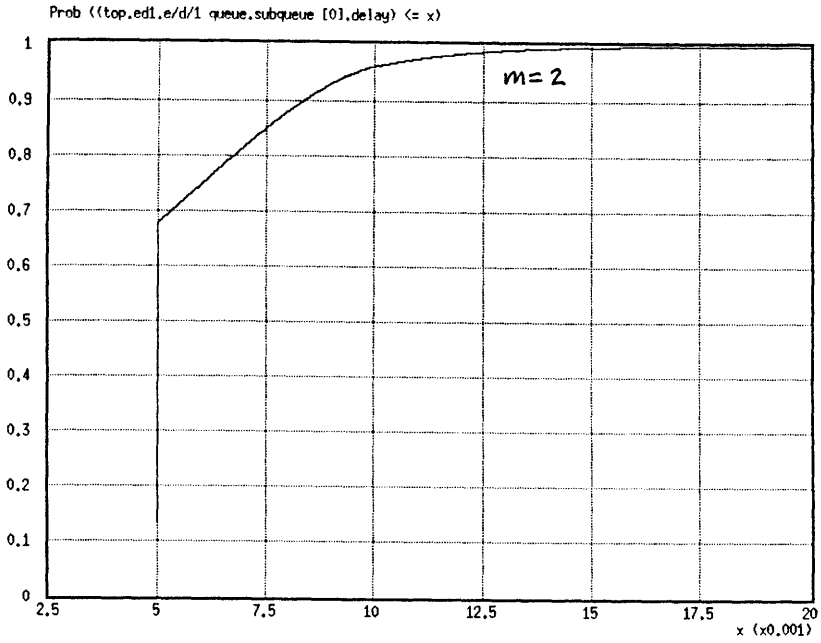

Prob ((top.node_0.e/m/1 queue.subqueue [0].pksize) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
1
2
3
4
5
6
7
8
9
10
x

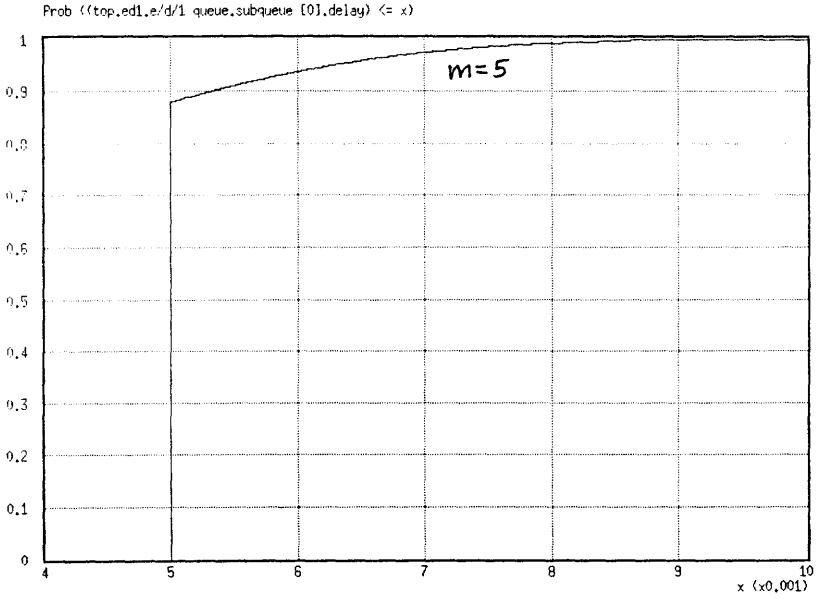
PLOT 6

PLOT 7

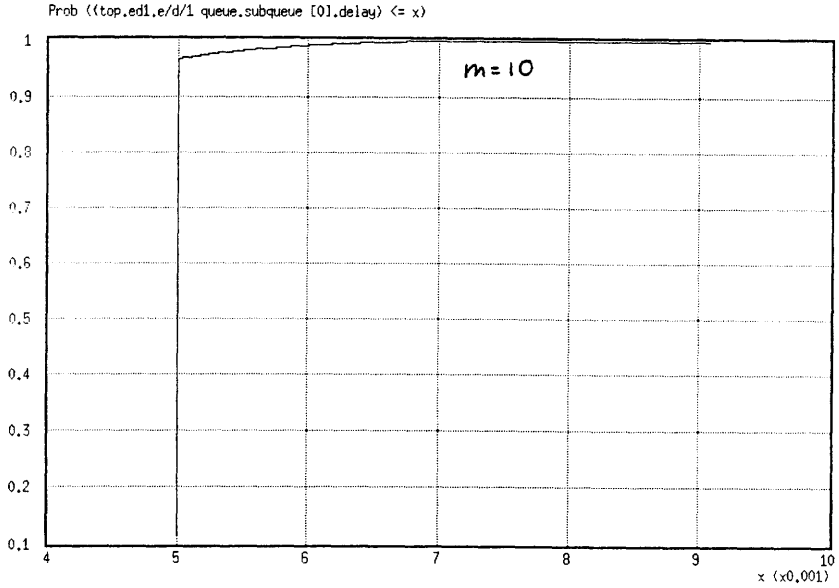

Prob ((top.mdk.m/d/k queue.subqueue [0].delay) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
x

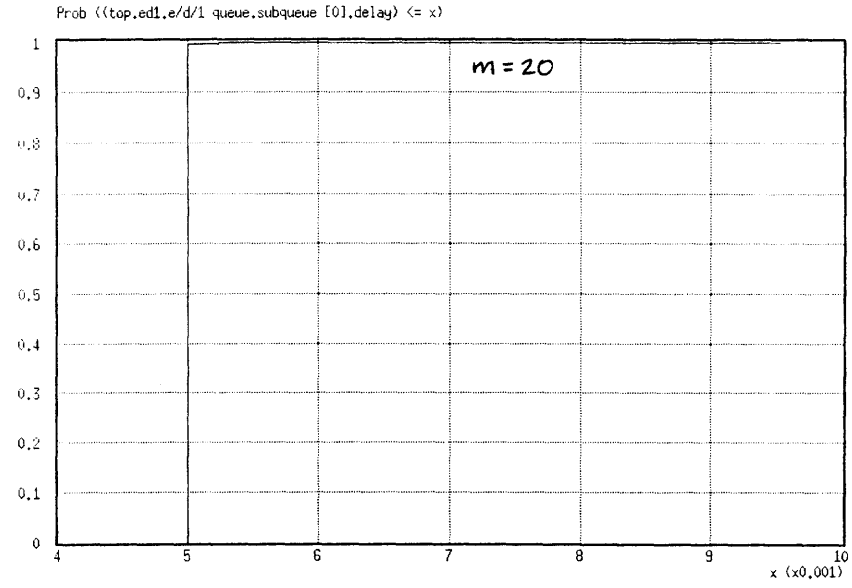
PLOT 8

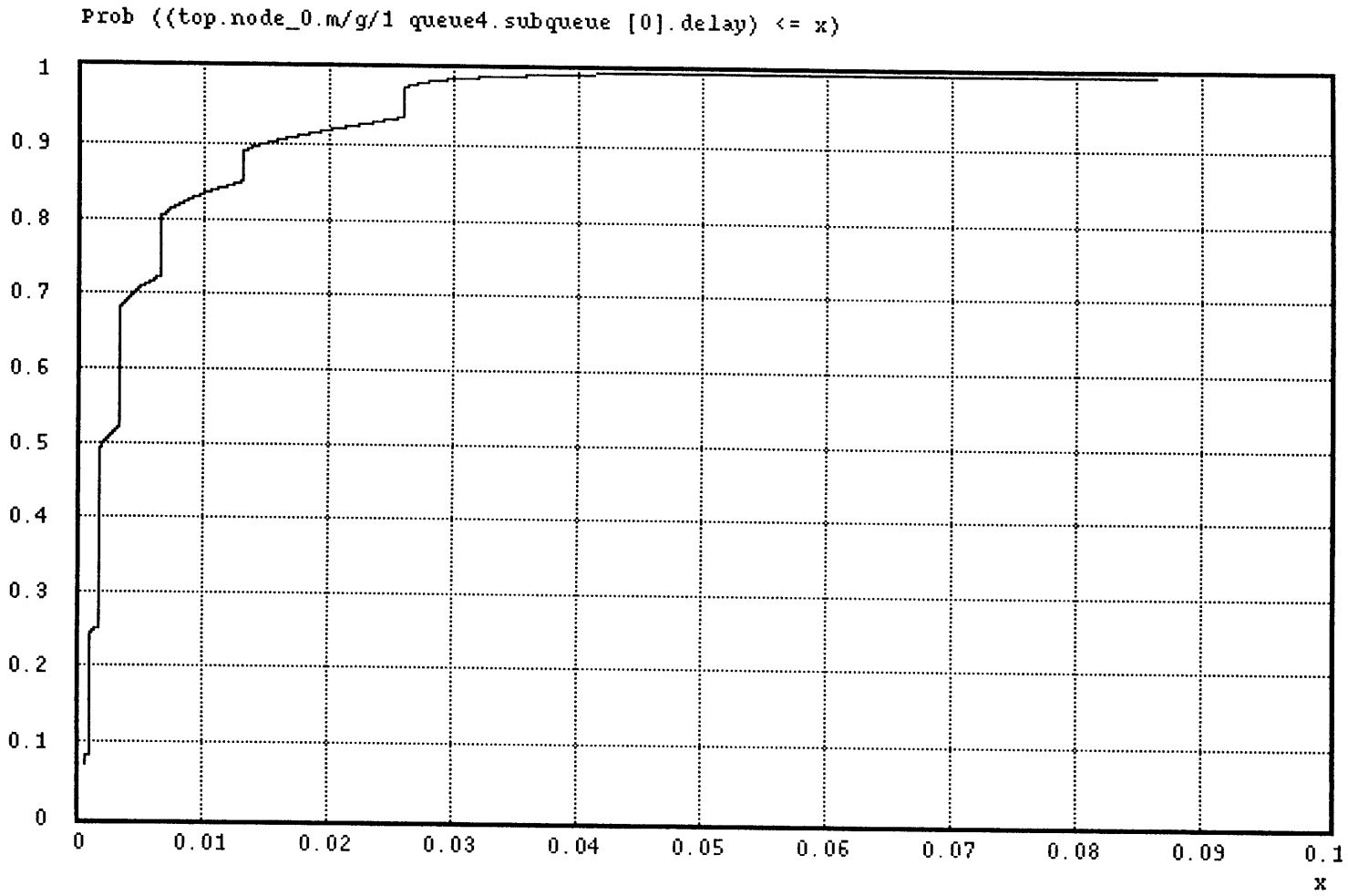

Prob ((top.ed1.e/d/1 queue.subqueue [0].delay) <= x)
m=2
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
2.5
5
7.5
10
12.5
15
17.5
20
x (x0.001)


Prob ((top.ed1.e/d/1 queue.subqueue [0].delay) <= x)
m=5
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
4
5
6
7
8
9
10
x (x0.001)

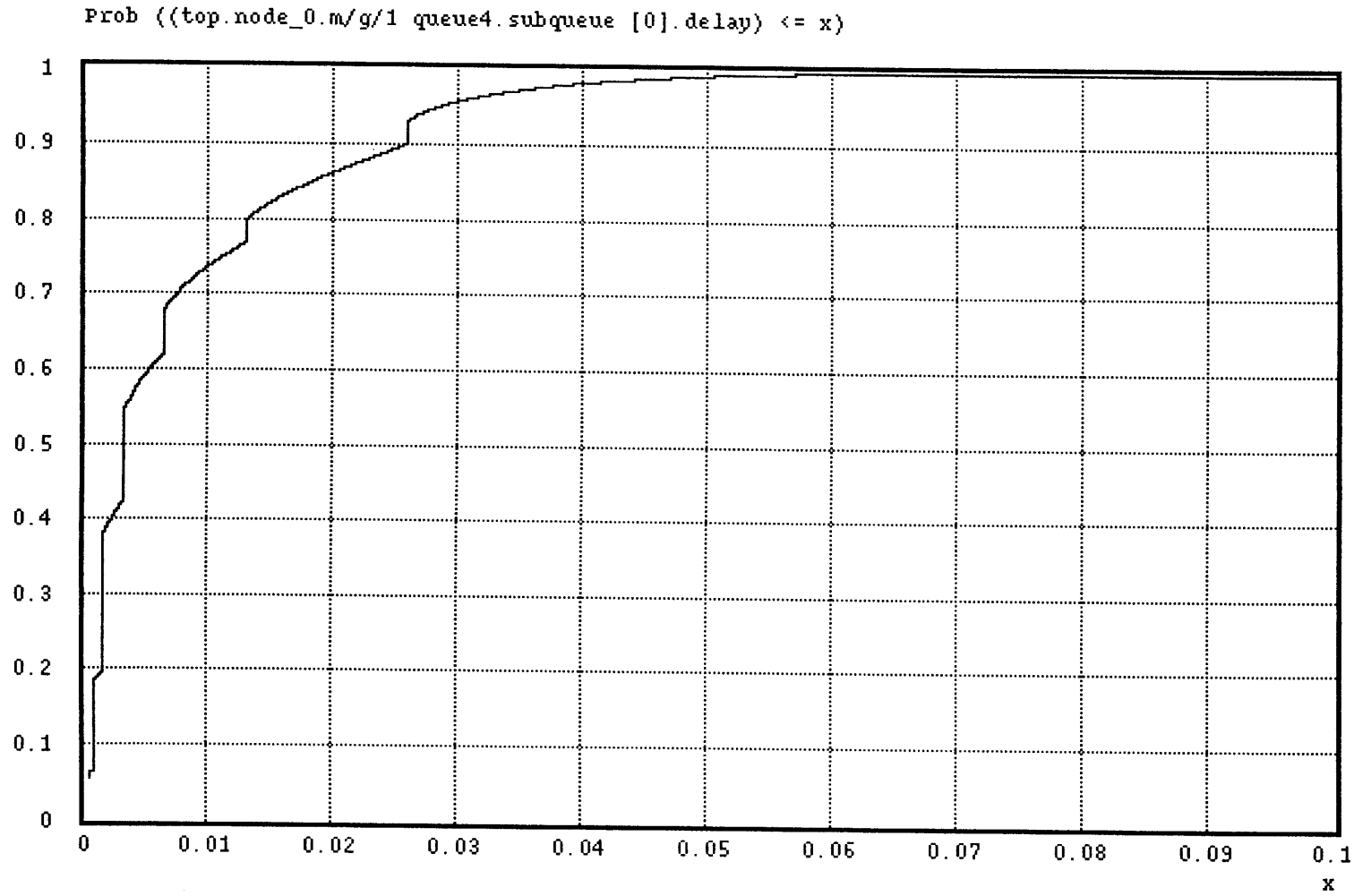
PLOT 8 (continued)






Prob ((top.node_0.m/g/1 queue4.subqueue [0].delay) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
x

PLOT 9


Prob ((top.node_0.m/g/1 queue4.subqueue [0].delay) <= x)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
x

PLOT 10

References

1. Bao, Gang. *"Performance Evaluation of TCP/IP Protocol Stack over CDMA Wireless Link."* J.C. Baltzer AG, Science Publishers.

2. Bertsekas, Dimitri and Gallager, Robert. ***Data Networks.*** Second Edition. Prentice-Hall, Inc., 1992.

3. Black, Uyless. ***TCP/IP & Related Protocols.*** Second Edition. McGraw-Hill, Inc., 1995.

4. Comer, Douglas E. and Stevens, David L. ***Internetworking with TCP/IP.*** Volume II. Prentice-Hall, Inc., 1991.

5. Daigle, John N. ***Queueing Theory for Telecommunications.*** Addison Wesley, 1992.

6. Drake, Alvin W. ***Fundamentals of Applied Probability Theory.*** McGraw-Hill Publishing Company, 1988.

7. Flannery, Brian P.; Press, William H.; Teukolsky, Saul A.; Vetterling, William T. ***Numerical Recipes in C.*** Second Edition. Cambridge University Press, 1996.

8. Gross, Donald and Harris, Carl M. ***Fundamentals of Queueing Theory.*** Second Edition. John Wiley & Sons, Inc., 1985.

9. RFC 793, ***Transmission Control Protocol.*** DRAPA Internet Program, Protocol Specification, 1981.

10. TIA/EIA/IS-99, ***Data Services Option Standard for Wideband Spread Spectrum Digital Cellular System.*** Interim Standard, 1995.

11. TIA/EIA/IS-657, ***Packet Data Service Option Standard for Wideband Spread Spectrum System.*** Interim Standard, 1996.

12. TIA/EIA/IS-707.2, ***Data Service Options for Wideband Spread Spectrum Systems: Radio Link Protocol.*** Interim Standard, Ballot Version, 1997.