Wireless ATM Network Medium Access Control with

Adaptive Parallel Multiple Substream CDMA Air-interface

Tae-In Hyon

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Dr. N. J. Davis IV, Chairman
Dr. S. F. Midkiff
Dr. D. G. Sweeney
Dr. M. T. Jones
Dr. J. D. Arthur

June 19, 2001
Blacksburg, Virginia

Key Words: Wireless ATM, Adaptive MAC Protocol, DS-CDMA, Multi-Code CDMA

Wireless ATM Network Medium Access Control with

Adaptive Parallel Multiple Sub-stream CDMA Air-interface

Tae-In Hyon

Abstract

One of the most important components of any wireless network is the medium access control protocol. This research deals with wireless ATM (WATM) medium access control (MAC) protocol. Conventional studies concerning WATM have focused mainly on variations of the time-division-multiple-access (TDMA) method for the wireless aspect of WATM networks. However, there are many advantages that the direct-sequence code-division-multiple-access (DS-CDMA) air-interface method has, such as inherent robustness against multipath fading, better resilience against security infringement attempts, and greater overall capacity compared to the TDMA method as proven in the cellular telephone industry. The main reason behind the relatively broader support for the TDMA method is that the source bit rate is generally higher compared to the DS-CDMA method since the maximum data rate per mobile unit is limited by the processing gain of a traditional DS-CDMA method.

In this research, the problem of limited data rate often associated with a DS-CDMA air-interface is alleviated by employing the recently conceived multi-coded DS-CDMA as the primary air-interface, which is known to achieve maximum data rate per mobile unit comparable to applications employing TDMA. The focus of this research is on overcoming periods of significant deterioration of the wireless channel by adaptively employing bit combining. A MAC protocol called Adaptive Parallel Multiple Sub-stream CDMA (APMS-CDMA) is proposed to alternate between normal and “rake-in” mode to deal with the often hostile environment of a WATM network.

Although the context in which this research effort was conducted was a wireless ATM network environment, the protocol and techniques developed here can be applied to other infrastructure wireless systems using multi-code CDMA as their air-interface. Further,

independent of the air-interface technique employed, other wireless systems can benefit from the channel estimation and the traffic management techniques used in this research effort.

Acknowledgments

I would like to thank all the committee members – Dr. Nathaniel J. Davis IV, Dr. Scott F. Midkiff, Dr. Dennis G. Sweeney, Dr. Mark T. Jones, and Dr. James D. Arthur – for their helpful and insightful comments about my research. Especially, I would like to thank my advisor, Dr. Davis, for always keeping my research path in perspective and answering all my questions with clear and decisive suggestions that was instrumental in ensuring the completion of this dissertation. I am grateful to Dr. Midkiff for allowing me to be one of his GTAs for the various network classes, which enabled me to be exposed to various aspects of (wireless) computer networks.

There were also numerous visits made to Dr. Woerner concerning the general practices of CDMA usage, which helped shape the bit error rate analysis. Dr. Gray was always willing to listen and help me sort out the coding issues concerning the BCH code. For the use of the computer facilities, I am greatly indebted to the Virginia Tech Information Systems Center and to Dr. Dong S. Ha.

There were many occasions where the help of colleagues were indispensable in making progress in my research. Special thanks goes to Dr. Jae-Hong Park for his many productive comments about various aspects of statistics and numerical analysis methodology. Jahng S. Park helped me get acquainted with OPNET Modeler early on in research. Suk-Won Kim clarified various generation methods of PN codes in use today. There were constructive discussions concerning the usage of OPNET Modeler with Carl Fossa and Dr. Rusty Baldwin, who are and were with the Center for Wireless Telecommunications, respectively.

Finally, the full support of my wife, Young-Hee Hyon, was essential in making me see through the Ph.D. program, which inevitably involved some periods of turmoil. I also would like to thank my parents for instilling the importance of higher education from early on and giving helpful advice concerning my career path.

Any misstatements or errors contained within this dissertation are attributed solely to the author.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Background

The latest trend setting the pace in the modern communication world is the love for mobility. It is not difficult to witness people using their cellular phones in public places, despite the fact that there are usually cheaper public pay phones just feet away. This is a testament to the expanding need for people not to be tethered to wired terminals. This kind of attitude naturally carries into the computer networks arena. The demand for mobility in computer networks has created interest from the academic to the corporate community, and we are not far away from the general consumers showing interest in wireless multimedia services. The supply of networking is already being hard pressed to keep up with the burgeoning demand for better service, and the pursuance of mobility has made the issue even more challenging for engineers.

A further trend being noticed is that the information being conveyed over the network, whether it is wired or wireless, is now converging toward digital technology. In recent years, the number of subscribers to analog mobile radio systems has started to decrease. On the other hand, the number of subscribers to digital mobile radio systems is increasing steadily [HaM00]. The obvious reason being that on top of the many advantages that digital technologies have over analog technologies in terms of bit error rate and reliability, it enables voice and video, as well as the conventional data information to be integrated on the same network. For example, we can witness the cellular telephone network moving toward digital technology at every chance [Siu94]. The U.S. Advanced Mobile Phone System (AMPS), which is the first major standard to

be deployed in a large-scale commercial cellular phone system, is an analog system that uses frequency modulation (FM) for radio transmission in a frequency division multiple access (FDMA) setting. Careful studies that included frequency reuse plans, signal-to-interference ratio, and, most importantly, the efficient use of the spectrum, all led to the conclusion that digital transmission technology was inevitable in keeping pace with the increasing user demand. In late 1991, the U.S. Digital Cellular (USDC) system interim standard, IS-54, was introduced, which incorporated a Time Division Multiple Access (TDMA) technology. IS-54 enabled three users in the same 30 kHz bandwidth where the analog AMPS system was only able to fit in one user, providing practical proof that digital technology indeed had merits that could be used to advantage [EIA90]. Continuous advancement in digital signal processing technology enabled Qualcomm, Inc. to implement the Direct-Sequence Code Division Multiple Access (DS-CDMA) method for transmission in a commercial setting for the first time. DS-CDMA is essentially an extension to the basic Spread Spectrum (SS) technology that was used by the military to combat potential unauthorized forces eavesdropping on transmissions or jamming signals. DS-CDMA proved effective in working under signal-to-interference levels that were much harsher than requirements for analog or even digital TDMA. DS-CDMA systems use the same set of frequencies in every cell, including neighboring cells, which directly converts to providing higher capacity.

Today's computer networks are fast becoming a medium for people to send their video, audio, and traditional data, which are labeled as "multimedia" for this research. The once vivid line that distinguished between computer data and video/audio is no longer obvious. A basic building block that is proposed to accommodate the need for multimedia is Asynchronous Transfer Mode (ATM), which has received growing attention as the standard for the Broadband Integrated Services Digital Network (B-ISDN). ATM is well suited for multimedia applications due to its intrinsic characteristics of supporting multiple streamlines with varying data rates and quality-of-service (QoS) features that can be adjusted to users' needs.

Inspired by the potential of ATM, tremendous efforts have been exerted by a growing number of researchers to generate the wireless extension to the B-ISDN networks, which will be able to support multimedia with guaranteed QoS. (The standardization attempt is currently in progress.) There are many discussions and controversies over what wireless protocol to employ

in the wireless aspect of the wireless ATM (WATM) network. The two competing protocols for the air-interface are DS-CDMA and TDMA. Opponents to employing the DS-CDMA system as the standard point out the fact that DS-CDMA limits the peak data rate of a connection to a relatively low value, which may pose a problem for broadband applications [PaP97]. However, by using a higher frequency bandwidth enabling broadband CDMA (B-CDMA) and implementing a method such as multi-code CDMA, the data rate limitation can be overcome [IGi95]. Moreover, the intrinsically statistical nature of the DS-CDMA air-interface method provides a natural integration in the WATM network of various types of traffic [MHK94].

1.2 Research Goals

The primary objectives of this research are two-fold. The first is to extend the knowledge of WATM network MAC protocols by examining the existing works that have been directed toward achieving the goal of better WATM networks using various medium access control (MAC) protocols. Particular emphasis is given to the two most sought-after air-interface technologies in WATM, TDMA and DS-CDMA. Second, a new way of adapting to the network situation that affects the actual performance of the network as a whole is investigated, and a method that will improve the overall network performance is proposed. The method employed will adapt to the environment in a dynamic manner to achieve better overall throughput in the face of an intermittently degraded wireless link that is expected to occur. To achieve these goals, a communication network simulator OPNET Modeler®[1] version 6.0.L is used to model the WATM network.

1.3 Document Overview

Chapter 2 presents the MAC layer protocols of some of the already standardized wireless LANs, specifically IEEE 802.11 and HIPERLAN. ATM itself is briefly introduced, as well as well as some of the proposed MAC layer protocols for WATM. Some comparison between the two most popular forms of the air-interface to be used as part of the MAC protocol, namely, the

[1] OPNET Modeler® is a registered trade mark of OPNET Technologies.

TDMA and DS-CDMA, is made to provide insight into the merits and demerits concerning their suitability as wireless MAC protocols.

Chapter 3 addresses some of the problems that face any wireless application and considers the importance of robustness and the ability of the protocol to handle a harsh wireless link environment. The research objective is given based on the problems mentioned. Some assumptions about the system are also given.

Chapter 4 provides a detailed explanation of the concept and methodology of the newly proposed APMS-CDMA MAC protocol. A detailed analysis is made of the concept of a "rake-in" mode, which is used adaptively.

Chapter 5 sets forth some of the parameters, assumptions, and methods employed in doing simulation for this research. Also, a validation of the simulation model is done against the DQRUMA MAC protocol in order to confirm proper working of the new design.

Chapter 6 gives the simulation results obtained through this research. Also the results are examined to show that substantial improvements in overall throughput were achieved using the APMS MAC protocol, compared to the conventionally proposed WATM system using the DQRUMA MAC protocol.

Chapter 7 concludes the dissertation with a summary of the research, including the significance of the research results. Finally, recommendations for future research directions are given.

Chapter 2: Background and Literature Survey

There are still many implications that suggest marked differences between existing wireless local area networks (WLANs) and the wireless asynchronous transfer mode (WATM). Section 2.1 describes these differences. In Section 2.2.1 and Section 2.2.2, the two already established standards – the IEEE standard (802.11) and ETSI's HIPERLAN standard – are introduced, respectively. In Section 2.3, the background information on the WATM network is presented. The ATM itself is described in Section 2.3.1, and Section 2.3.2 surveys various proposed wireless ATM (WATM) MAC protocols in the literature.

2.1 Wideband Wireless Local Access

Traditional WLANs and the WATM both provide wideband wireless local access (WWLA) [PaZ97]. However, there are still differences that prevail. WLAN is a mature technology with off-the-shelf products available. WATM is an emerging technology that has caught the attention of wireless network researchers and developers relatively recently. In fact, the standards for WATM have yet to be ratified.

WLANs are considered mainly as products sold by the manufacturer for private use while WATM is largely considered to be a service provided by the operating company. WATM is expected to provide end-to-end ATM connectivity and quality of service (QoS) [PaZ97]. For the moment, WLANs and WATM offer two alternatives for wideband wireless access to the network backbone, and there is no indication that these two will be united any time soon, although significant overlaps between these two suggest that it is possible in the long run.

2.2 Wireless Local Area Network (IEEE802.11, HIPERLAN)

WLANs provide access to conventional local area networks (LANs) that have already been proven and deployed. The majority of the conventional LANs in the U.S. conform to the IEEE 802.3 standard [IEE85], the most prominent of which is Ethernet. The IEEE 802.11 standard [IEE99] carries on and extends the basic idea of the IEEE 802.3 standard to handle data networks in a wireless environment. A more in-depth look into the IEEE 802.11 is presented in Section 2.2.1.

In Europe, the European Telecommunications Standards Institute (ETSI) Committee RES-10 was founded in 1991 to develop a standard called HIgh PERformance radio LAN (HIPERLAN). HIPERLAN is designed to provide high data rate up to 20 Mbps in short distance radio links between computers using frequency bands at 5.2 and 17.1 GHz [Fla94, PaP95].

2.2.1 IEEE 802.11

The IEEE Project 802.11 Committee was formed in 1990 "to develop a medium access control (MAC) and physical layer (PHY) specification for wireless connectivity for fixed, portable and moving stations within a local area" [IEE99]. The IEEE 802.11 employs the carrier sense multiple access with a collision avoidance (CSMA/CA) technique for medium access control. The existing products conforming to the IEEE 802.11 standard support one of the following three operating modes in the PHY layer. A direct-sequence spread spectrum (DSSS) or frequency hopping spread spectrum (FHSS) mode is used in the industrial, scientific, and medical (ISM) bands for its applications. For shorter distances, and often times indoor applications, a diffused infrared (IR) technology is used. In the IEEE 802.11, the medium access control (MAC) layer supports two types of network architectures, which are the infrastructure network and the ad hoc network.

Readers interested in in-depth coverage of the IEEE 802.11 standard are referred to [IEE99].

2.2.1.1 Infrastructure Network

An infrastructure network is an architecture for providing communication between wireless stations and a wired network. The communication is established from the wireless stations to the wired network through an access point (AP). This arrangement allows for point coordination of all of the stations in the basic service area and ensures proper handling of data traffic that is essentially contention-free. A coverage area is defined by its AP and its associated wireless stations. These devices constitute the basic service set (BSS). In an infrastructure based IEEE 802.11, there are usually more than one BSSs that are connected by a wired network such as Ethernet (IEEE 802.3) via the APs.

2.2.1.2 Ad hoc Network

An ad hoc network is an architecture that is used to support mutual communication among wireless clients that are usually created spontaneously. An ad hoc network does not support access to a wired network, and an AP is usually not included as part of the network. The IEEE 802.11 standard specifies CSMA/CA as its access method in the Distributed Coordination Function (DCF), which is the fundamental access protocol adjoining the PHY layer.

2.2.1.3 CSMA/CA Protocol

The CSMA/CA protocol avoids collisions instead of detecting a collision like the algorithm used in IEEE 802.3. This is due to the fact that it is difficult to detect a collision in an RF transmission network. The MAC layer operates together with the PHY layer by sampling the radio frequency (RF) energy over the medium where the data are being transfered. The PHY layer uses a clear channel assessment (CCA) algorithm to determine if the channel is clear. This is accomplished by measuring the RF energy at the antenna and determining the strength of the received signal. If the received signal strength is below a specified threshold, the channel is assumed clear and the MAC layer is given permission for data transmission. Another CCA option specified by the standard is that the carrier sensing can be used to determine if the channel is available. Carrier sensing is done by checking whether the incoming signal is of the same carrier type as the IEEE 802.11 transmitters.

The CSMA/CA protocol allows for options that can minimize collisions by using request-to-send (RTS), clear-to-send (CTS), data, and acknowledge (ACK) transmission frames, in a sequential manner. Communication is commenced when one of the wireless stations sends a short RTS frame. The RTS frame includes information about the destination and the length of message. The message duration is known as the "network allocation vector" (NAV). The NAV alerts all others in the medium to back off for the duration of the transmission. The receiving station issues a CTS frame, which echoes the sender's address and the NAV. The use of the RTS, CTS, Data and ACK sequence is especially helpful when dealing with what is known as the "hidden node" problem.

Although DCF is intrinsically a contention-based asynchronous access method, the optional Point Coordination Function (PCF) makes it possible to have contention-free access in the case of an infrastructure network configuration.

2.2.1.4 PHY Layer

The PHY layer of the IEEE 802.11 protocol is at the bottom of the Open System Interconnection (OSI) stack, and it interfaces between the MAC and wireless media. The PHY layer consists of two sublayers, the physical layer convergence procedure (PLCP) and the physical medium dependent (PMD).

There are three kinds of PHY layers – the DSSS PHY, FHSS PHY, and the IR PHY. Each of them has their own exclusive PLCP and PMD sublayers. For all three kinds of PHY layers, the preamble and header portion of the PLCP protocol data unit (PPDU) is always sent at 1 Mbps, while the PLCP service data unit (PSDU) portion can be sent at either 1 Mbps or 2 Mbps.

The DSSS PHY uses the 2.4 GHz frequency band. The DSSS PMD sublayer converts the binary bits of information from the PLCP protocol data unit (PPDU) into RF signals by using carrier modulation and DSSS techniques. Each DSSS PHY channel occupies 22 MHz of bandwidth. Although channels are spaced 5 MHz apart, the channels that are used simultaneously are spaced 25 MHz apart in the 2.4 GHz frequency band to allow noninterference among used channels.

The FHSS PHY also uses the 2.4 GHz spectrum as the transmission media. PHSS PMD controls the data transmission over the media as directed by the FHSS PLCP sublayer. Gaussian frequency shift key (GFSK) modulation is used to shift the frequency either side of the carrier hop frequency, depending on whether the binary symbol is a 1 or 0. The channels are evenly spaced across a 83.5 MHz spectrum. In the US, Part 15 of FCC rules requires that the FHSS radios hop a minimum of 2.5 hops per second for a minimum hop distance of 6 MHz.

IR PHY differs from DSSS and FHSS in that IR requires line-of-sight or reflection off objects and it cannot pass through walls. IR PMD controls the data transmission over the media as directed by the IR PLCP sublayer. IR PHY uses pulse position modulation (PPM) to transmit binary data at 1 and 2 Mbps.

2.2.1.5 IEEE 802.11a – The OFDM PHY

The IEEE 802.11a PHY is one of the two PHY extensions of IEEE 802.11 and is alternately called orthogonal frequency division multiplexing (OFDM) PHY. The OFDM PHY, which operates at the 5.0 GHz spectrum, enables the PSDU frames to be transmitted at data rates up to 54 Mbps [IEE99-a].

The PHY's PLCP sublayer and PMD sublayer are used exclusively for the OFDM PHY. The PLCP preamble and signal fields are always sent at 6 Mbps, while the data rates supported for the PSDU ranges from 6 Mbps to 54 Mbps. There are a total of 52 subcarriers defined in IEEE 802.11a, where 48 of them are data subcarriers and 4 of them are pilot subcarriers. Although, OFDM reduces the inter symbol interference (ISI) due to its parallel structure, it is subjected to frequency selective fading and thus uses bit interleaving and convolutional encoding to improve bit error rate performance. The encoding rate used for PLCP preamble and signal field is 1/2, while the encoding rate for the PSDU portion ranges from 1/2 to 3/4.

The IEEE 802.11a is considered a major step forward in accommodating the higher data rate applications, made possible only recently by advancements in digital signal processing (DSP) and very large scale integrated (VLSI) technologies. The cost may still be an obstacle at this time, but with proliferation of multimedia applications becoming a reality, the high utilization of the wireless channel using OFDM technology in IEEE 802.11a is becoming increasingly attractive.

2.2.1.6 IEEE 802.11b – 2.4 High Rate DSSS PHY

The IEEE 802.11b PHY is another PHY layer extension of the IEEE 802.11 and is alternately known as high rate direct sequence spread spectrum (HR/DSSS). The improvements in data rate of the PSDU enable either 5.5 Mbps or 11 Mbps, and it is designed to be interoperable with the legacy IEEE 802.11 2.4 GHz RF PHY layers. This is achieved by allowing HR/DSSS networks to fall back to 1 and 2 Mbps when necessary [IEE99-b].

The HR/DSSS PHY uses the same spectrum as specified for IEEE 802.11 direct sequence PHY. For applications used in the same area, the channel center frequencies are also spaced 25 MHz apart, as was the case for IEEE 802.11 direct sequence PHY. Moreover, IEEE 802.11b allows IEEE 802.11 FHSS 1 and 2 Mbps networks to interoperate with HR/DSSS 11 Mbps WLANs. These interoperability efforts, along with enhanced data rates compared to the legacy IEEE 802.11, have triggered increasing acceptance of products conforming to the IEEE 802.11b standard.

2.2.2 HIPERLAN

HIPERLAN is a European standard developed by ETSI. The committee responsible for HIPERLAN is RES-10, which has been working on the standard since November 1991.

The HIPERLAN standard defines part of the bottom two layers of the Open System Interconnection (OSI) model, namely the PHYsical layer (PHY) and the Data Link layer (DLC). Only the medium access control (MAC) sub-layer within the DLC is specific to HIPERLAN.

Two distinct frequency bands are allocated. One is in 5.15 to 5.25 GHz, with potential extensions possible for up to 5.30 GHz, and the other is in 17.1 to 17.3 GHz range.

The MAC scheme used to access the channel is called Elimination Yield Non-Preemptive Priority Multiple Access (EY-NPMA). A single channel access cycle comprises the priority-signaling phase, the elimination phase, the yield phase, the frame transmission phase, the acknowledgment phase, and the guard time [SiS98, FGB97]. In the non-trivial case of when multiple mobile stations are contending for the channel, each station sends out the priority of the message that it intends to send. The priorities are indicated by occupying one of the priority slots within the priority-signaling phase. There are five priority slots, of which the one with the least

listening time has the highest priority. The station with the highest priority will win the right to transmit and the other stations, sensing this situation, will back off from further steps within the channel access cycle. If 2 or more stations send the same priority, then in the following elimination phase, each station randomly prolongs sending the priority pulse and then immediately listens for other stations that may be sending an even longer priority pulse. The winner(s) will be those who have sent the latest pulse (and have not detected any pulse as it releases its priority pulse). In case of a tie, the yield phase is entered where this time each remaining station holds for a random number of slots. If a station detects that the channel is free after this time, then it starts to transmit. There may yet again be multiple stations that have actually started transmitting data at the same time, but the described scheme ensures that the constant collision rate is lower than 3.5%, essentially independent of the number of competing stations [SiS98]. This is considered a major advantage over the CSMA/CA scheme used in IEEE 802.11.

Although HIPERLAN data rates of well over 10 Mbps are achievable for service unit sizes that are big, throughput drops dramatically for small service unit sizes such as ATM cells. This has motivated the ETSI RES-10 committee to push for new HIPERLAN standards, namely HIPERLAN 2, 3, and 4, that would accommodate ATM. As with any standard, HIPERLAN 2, 3, and 4 will take several years to complete and some cooperation between the ETSI and ATM Forum is foreseen to develop standards that will not become obsolete.

2.3 Wireless ATM

In contrast to WLANs described in Section 2.2, wireless ATM (WATM) is a relatively new addition to the various wireless network technologies that exist today. The intrinsic structure that enables multimedia traffic to be sent over the same channel has been one of the most attractive features of the ATM network. WATM is essentially an extension to the ATM broadband network. The general view is that the WATM will feature an infrastructure type of setup due mainly to the quality of service issues that has to met regarding various types of traffic being integrated into a single network over a wireless environment. One of the goals is to minimize the perceivable degradation in performance felt by the user when using a wireless device instead of a wired one. There are many ways to alleviate the problem of unreliability in

the wireless medium. We will focus on what can be done about the MAC layer issues related to the WATM network that would give us some leverage in overcoming the huge gap between wireless and wired channel conditions.

In Section 2.3.1, ATM itself is introduced in order to appreciate how the data characteristics differ from those of conventional LANs. The two most prominent wireless medium access technologies, direct-sequence code division multiple access (DS-CDMA) and time-division multiple access (TDMA), used for WATM are introduced in Section 2.3.2.

2.3.1 ATM

2.3.1.1 ATM Layer

ATM handles multiple streams of data (multimedia) services with different data rates. The term "asynchronous" refers to the fact that each stream of data is able to place its data onto the unique physical ATM stream as need arises without having to worry about irregularities in occurrence. This is in contrast to a synchronous transfer mode (STM), such as time division multiple access (TDMA). In fact, the allowance of this asynchrony makes ATM a suitable candidate for multimedia applications since each data stream generated by multimedia differs in both data rate and characteristics of Quality of Service (QoS) required. The term "cell" refers to the fixed-size basic unit of transmission in the ATM network. A cell comprises a 5-byte header field and a 48-byte information field, totaling 53 bytes as shown in Figure 2.1. The 5-byte header field includes identification (or payload type), control priority, and routing information. The 48-byte information field contains the actual data to be transferred over the network. The fixed cell size reduces the complexity of the network and provides flexibility making it suitable for WAN applications where interconnecting of LANs is a must [RaW94].

ATM switches support two kinds of interfaces, which are user-network interface (UNI) and network-node interface (NNI). UNI is used to connect hosts to an ATM switch, while NNI is used to connect two intermediate ATM switches within the network.

The cell header at the B-ISDN UNI differs from that at the B-ISDN NNI in the use of bits 5 to 8 in byte 1, as depicted in Figure 2.2. At the B-ISDN NNI, these bits are part of the virtual path interface (VPI), whereas at the B-ISDN UNI, they form the generic flow control (GFC) field

[HHS94]. The VPI and virtual channel identifier (VCI) will be explained in Section 2.3.1.2. The GFC mechanism helps to control the traffic flow from ATM at the B-ISDN UNI and alleviates short-term overload conditions that may occur in the customer network. Under normal mode when there is no appreciable accumulation of cells waiting in the ATM multiplexer buffer to be transmitted over the trunk, the GFC is set to the uncontrolled transmission mode (by setting all the bits to '0'). If there is a sudden surge of cells from some or all of the customer equipment (CEQ) devices and the ATM multiplexer buffers reaches near full capacity, then the GFC field is used to subject the cell flow from the various CEQ devices to the controlled transmission mode. However, the exact GFC procedure is not yet defined and some have even speculated it to be abandoned [SiJ95].

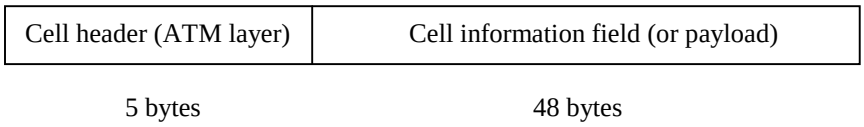


Figure 2.1: ATM Cell Structure

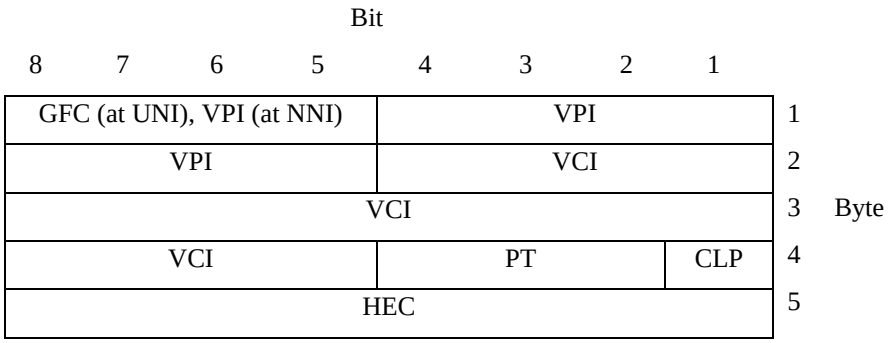


Figure 2.2: ATM Cell Header Structure

The payload type (PT) consists of 3 bits. The PT field is used to provide an indication of whether the cell payload contains user information or network information. When the PT field indicates that the cell contains network information, the network processes the information field of the cell. In the case where user information is to be transferred, the explicit forward congestion indication (EFCI) bit can be set when any network element is deemed to be congested [Cla96].

The cell loss priority (CLP) field consists of one bit, which is used to indicate the cell loss priority. Cells with the CLP bit set are at a risk of being discarded depending on the conditions in the network. The CLP bit may be set either by the user or service provider. A constant bit rate (CBR) connection usually has a high priority whereas a (variable bit rate) VBR connection may require a guaranteed minimum capacity as well as a peak capacity. It should be noted that in the case where the network is severely congested, even high priority cells will be discarded.

The header error control (HEC) field consists of 8 bits. This field is processed by the physical layer to detect errors in the header. The code used for this function is capable of either single-bit error-correction, or multiple-bit error-detection.

2.3.1.2 ATM Virtual Channels and Paths

Virtual channel is 'a concept used to describe a unidirectional transport of ATM cells associated by a common unique identifier value' [ITU91]. This identifier is called the virtual channel identifier (VCI) and of the 5 bytes (or 40 bits) in the ATM header 16 bits are used. A virtual channel connection (VCC) is a concatenation of virtual channel links, which are identified by VCIs [CuS93].

Virtual path is 'a concept used to describe unidirectional transport of ATM cells belonging to virtual channels that are associated by a common identifier value' [ITU91]. This identifier is called the virtual path identifier (VPI) and is also a part of the cell header. The VPI occupies 8 bits of the cell header at the UNI and 12 bits of the cell header at the NNI. A virtual path connection (VPC) is a concatenation of virtual path links, which are identified by VPIs [CuS93].

There are two types of ATM connections, namely the permanent virtual connection (PVC), which is set up through an external mechanism, often the network management, and switched virtual connection (SVC). For PVC, the switches that are placed along the way of the connection are programmed with the appropriate VPI and VCI values during setup time. For SVC, the connection is determined during the signaling phase and thus does not require any manual interaction. Thus, the flexible SVCs are more likely to be used in future applications dealing with multimedia.

The virtual channel concept can be explained more in detail as follows. The basic step in establishing an end-to-end connection is the request for a series of links from source to destination. The decision on whether to grant the request for the use of a link depends on the resources requested and the capacity remaining on the link [CuS93]. Routing translation tables in each VC switching node provide VCI translation information for every cell going into the switch. The routing translation table information is updated each time a new call is set up and remains constant throughout the duration of the call, which is why ATM is said to work in a virtual connection fashion. It should be noted, however, that the VCI will change from link to link (or from VC switch to VC switch).

The virtual path (VP) is a term for a bundle of virtual channel links that have the same endpoints. As mentioned earlier, the concatenated virtual path links form a VPC. The VPC endpoint is a where the VC links originate, translate, or terminate. Routing translation tables in each cross-connect node (VP switch) provide VPI translation for each cell entering a cross-connect. However, VCI values remain unchanged at VP switch.

It should be noted that VC switches terminate both VC links and necessarily the VP links. That is, VPI and VCI translation will occur at VC switches. Another important characteristics of the ATM transmission is that, as a result of the virtual connection, cell order is preserved.

2.3.1.3 ATM Adaptation Layer

The ATM adaptation layer (AAL) performs the necessary mapping between the ATM layer and the next higher layer. Its basic function is to provide adaptation of services of the ATM layer to the upper layers. The need for AAL arises due to the fact that ATM layer itself does not involve itself in distinguishing the actual type of data that it is carrying. Thus, the user payload is carried transparently over the ATM layer without any knowledge of structure of the data unit. As a result, all the functions specific to the services must be handled at a higher layer, which is the AAL. Theoretically, there would have to be an AAL for every service available. However, this would be a practical impossibility. Thus, a small set of AAL protocols or classes is defined in such a way that it would cover all the applications envisaged. ITU-T proposal was made with respect to the following parameters:

- timing relations
- bit rate
- connection mode.

Using these parameters, four classes have been defined as follows:

Class A: Circuit emulation, constant bit-rate video

Class B: Variable bit rate video and audio

Class C: Connection-oriented data transfer

Class D: Connection-less data transfer

Figure 2.3 depicts the relationship between the above three parameters and the four classes.

	Class A	Class B	Class C	Class D
Timeing relation between source and destination	Required		Not required	
Bit rate	Constant	Variable		
Connection mode	Connection oriented			Connection-less

Figure 2.3: Service Classification for AAL

These classes of service are a general concept and these can be mapped into the AAL protocol types as shown in Figure 2.4.

Class	AAL Type
A	AAL 1 (constant bit rate traffic, e.g. 64 Kbps voice and fixed-rate video
B	AAL 2 (variable bit rate traffic , e.g. packetized voice or video)
C & D	AAL 3/4 (Non-real-time data in messaging or streaming modes)
C & D	AAL 5 (Similar to AAL 3/4, but without multiplexing or error detection)

Figure 2.4: Mapping between Classes of Service and AAL Types

The AAL is divided into two sublayers: the convergence sublayer (CS) and the segmentation and reassembly sublayer (SAR).

The SAR sublayer is concerned with the segmentation of higher-layer information into a suitable size for the information field of an ATM cell and for reassembly of the contents in the ATM cell information fields into higher-layer information format.

This is usually insufficient in reconstructing the information sent and further processing must be done, such as cell-delay-variation, end-to-end synchronization, handling of loss and misinserted cells. Such functions are treated in the convergence sublayer (CS). The CS is service specific and a different convergence sublayer exists for each different service. The convergence sublayer is subdivided into two parts, the common part CS (CPCS) and the service specific CS (SSCS).

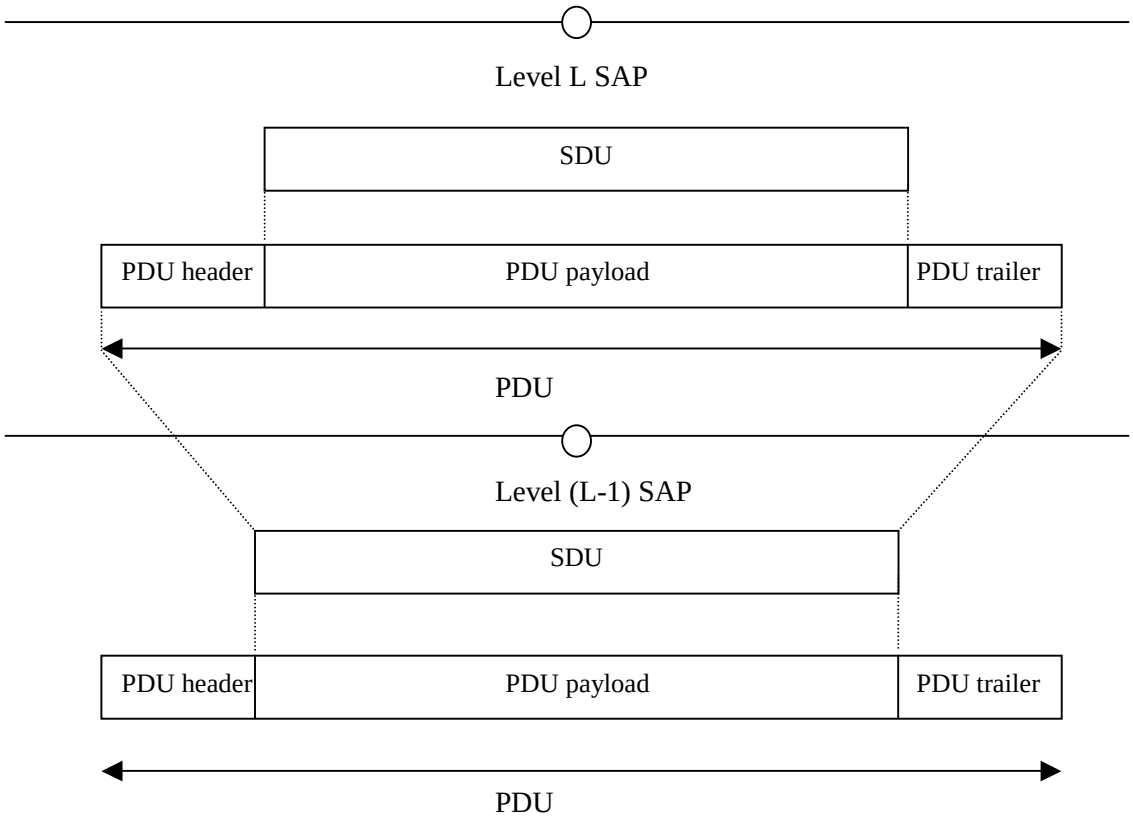


Figure 2.5: Relationship between SAP, PDU, and SDU

Before a more detailed explanation of the AAL is given, some of the commonly used terminology should be mentioned. The three terms adopted by the OSI and used often in describing any layer are service access point (SAP), protocol data unit (PDU), and service data unit (SDU). They are all related and the general concept can be visualized with Figure 2.5. Level L-1 SAP is a place where level L can access the services offered by layer L-1. Similarly, level L SAP is a place where level L+1 can access the services offered by layer L.

The AAL layer consists of the convergence sublayer (CS) and segmentation and reassembly (SAR) sublayer. It should be noted that there is no SAP between the CS sublayer and SAR sublayer. The CS sublayer can further be divided into two parts, which are common part CS (CPCS) and service specific CS (SSCS). However, it is possible that the SSCS sublayer remain null in the sense that it only provides for the mapping of the equivalent primitives of the AAL to CPCS and vice-versa [CuS93]. The structure of the AAL (type 3/4 and type 5) itself is described in Figure 2.6. The data unit naming convention pertaining to AAL in general is shown in Figure 2.7.

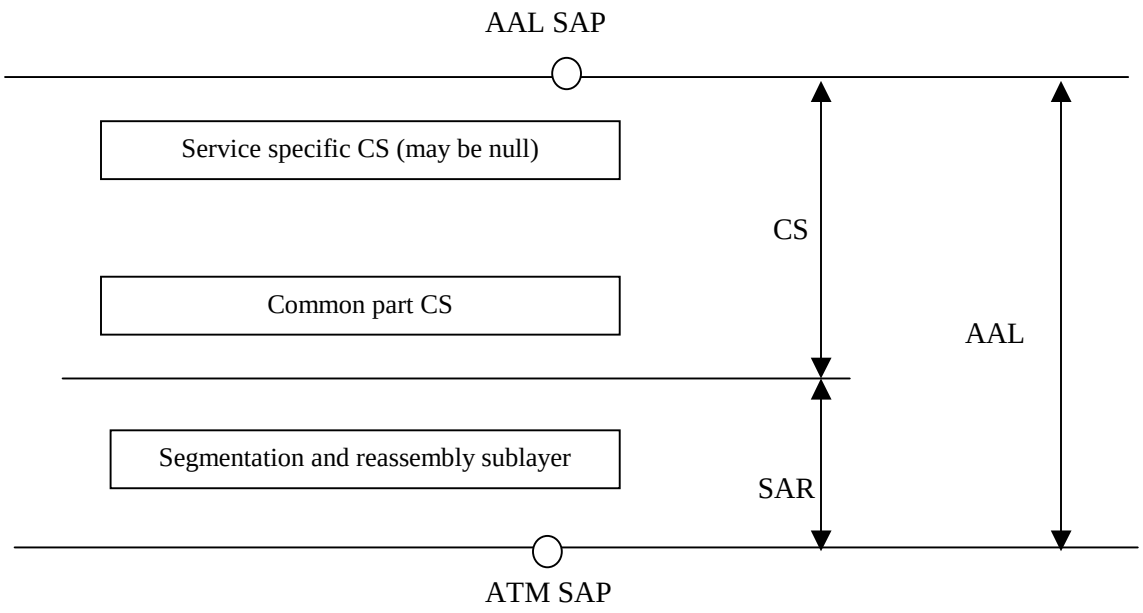


Figure 2.6: Structure of AAL (used in AAL type 3/4 and 5)

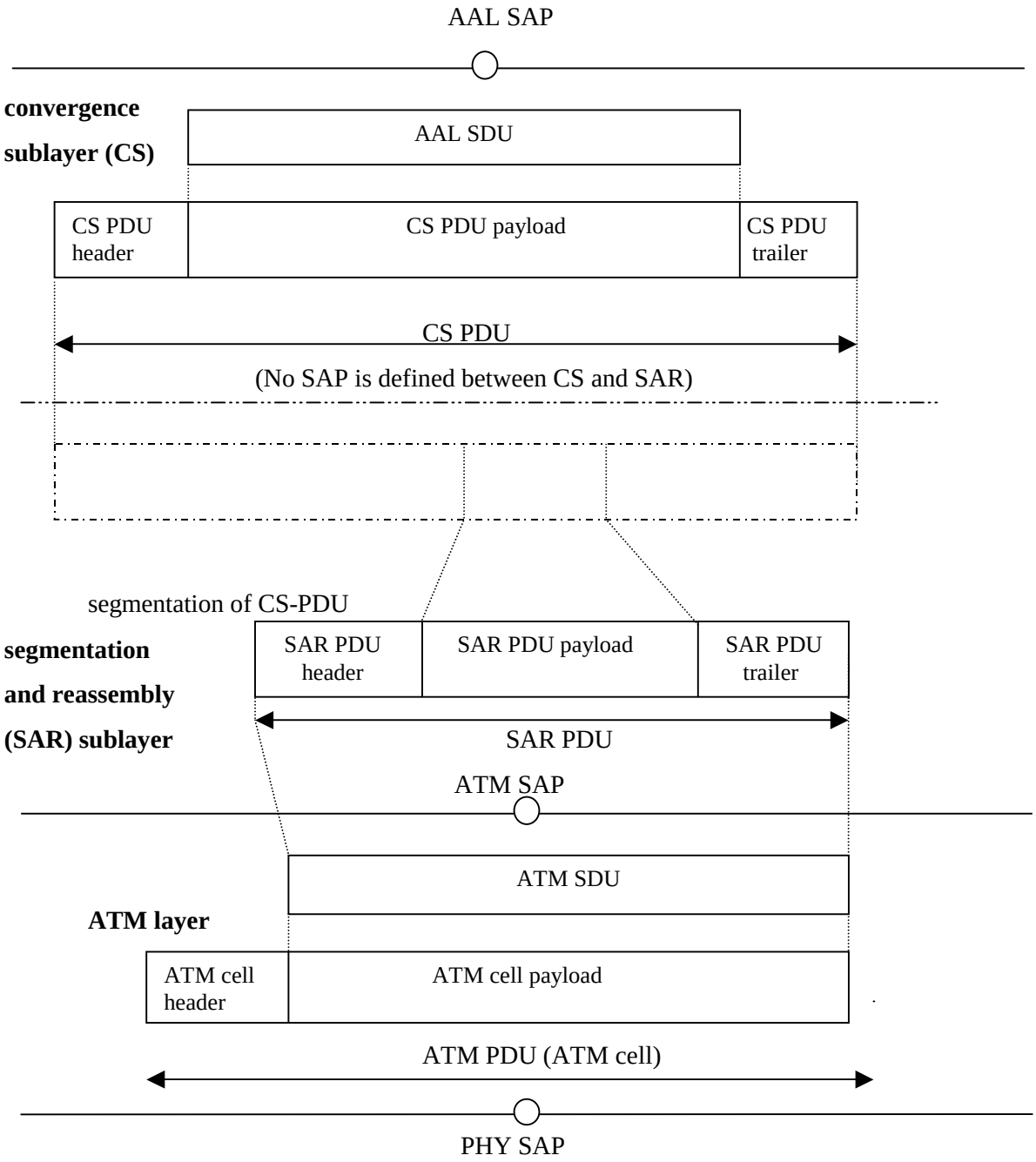


Figure 2.7: ATM and AAL data naming convention

Now, the general naming convention is applied to the AAL layer and ATM layer as depicted in Figure 2.7. It should be noted that there is no SAP defined between CS and SAR.

The more detailed description of each of the AAL protocol is presented in Sections 2.3.1.3.1 through 2.3.1.3.5.

2.3.1.3.1 AAL type 0

Although not officially categorized, AAL type 0 can be considered where the application does not require any special processing by the AAL, which in this case the SAR and CS have no functionality and thus any transfer of data through AAL would happen transparently without modification.

2.3.1.3.2 AAL type 1

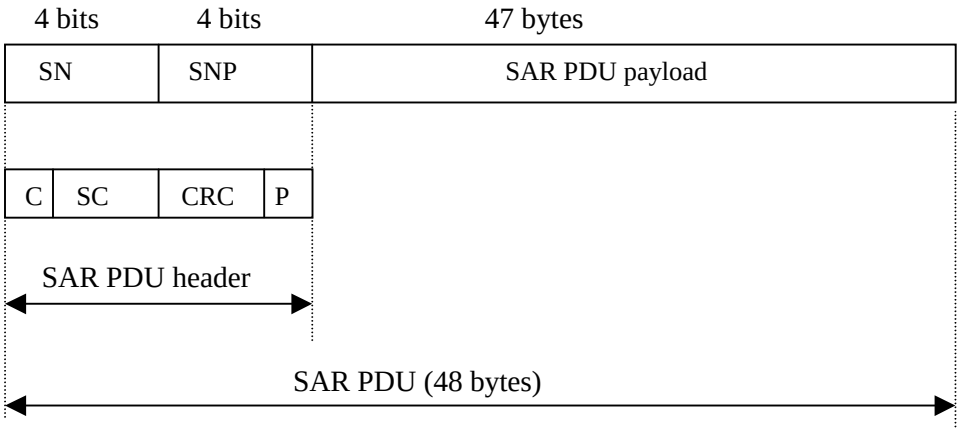
Usually, the CBR (constant bit-rate) services (class A) use AAL type 1 because it receives/delivers SDUs with constant bit rate from/to the AAL users at the above layer. The following layer services are provided to the AAL user:

- transfer of service data units with a constant source bit-rate and their delivery with the same bit-rate
- transfer of timing information between source and destination
- transfer of data structure information between source and destination
- indication of lost or errored information which is not recovered by AAL 1

The functions performed by the AAL are as follows:

- segmentation and reassembly of user infromation
- handling of cell delay variation
- handling of cell payload assembly delay
- handling of lost and misinserted cells
- source clock frequency recovery at the receiver
- recovery of the source data structure at the receiver
- monitoring of AAL PCI (protocol control information) for bit errors
- monitoring of the user information field for bit errors and possible corrective actions

At the SAR level, the sequence number protection (SNP) field, which provides 1-bit error correction and 2-bit error detection, is processed. If the result is right, the SN field is sent to the CS level, which processes it depending on the application. Figure 2.8 describes the SAR PDU format of AAL type 1.



C	Convergence sublayer indication	SAR	Segmentation and reassembly
CRC	Cyclic redundancy check	SC	Sequence count
P	Even parity bit	SN	Sequence number
PDU	Protocol data unit	SNP	Sequence number protection

Figure 2.8: SAR PDU format for AAL type 1

The SAR PDU header contains the protocol control information (PCI). The PCI is subdivided into a 4-bit sequence number (SN) and a 4-bit sequence number protection (SNP) field. The SN consists of a convergence sublayer indication (CSI) bit and a 3-bit sequence count field. CSI bit is used to transfer timing information and/or information about the data structure. The sequence count value of the SN enables the detection or loss or misinsertion of cells. However, the 3-bit sequence count field has its limitations when confronted with systems with high cell loss ratio. The SNP field contains a 3-bit cyclic redundancy check (CRC), which protects the SN field, and an even parity bit, which is calculated over the SAR PDU header.

At the CS level, functions depend largely on the service to be supported. Some of these functions are given below:

- Circuit transport to support both asynchronous and synchronous circuits.
- Video signal transport for interactive and distributive services.
- Voice-band signal transport.
- High-quality audio signal transport.

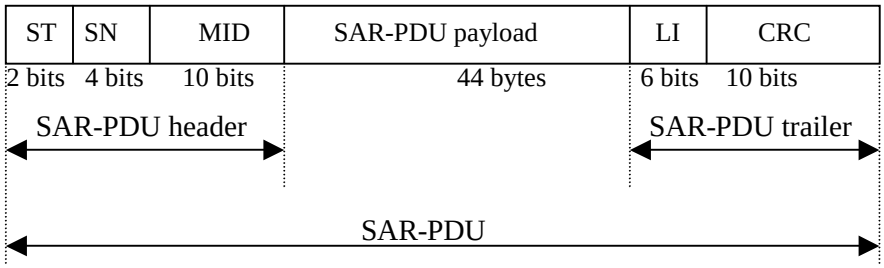
2.3.1.3.3 AAL type 2

AAL type 2 is similar to AAL type 1 except that AAL type 2 is for variable bit rate (VBR) service. With a VBR, the cell fill level may vary between cells, which may require the AAL to handle greater functionality. The partially filled cells are a necessary vice because they are needed to maintain a strict timing relation between source and destination despite the VBR. AAL type 2 is the least developed of the AAL types at this time.

2.3.1.3.4 AAL type 3 / 4

The standards originally foresaw two separate AAL types – 3 and 4, corresponding with the connection classes C and D, as seen in Figure 2.4. During the standardization process both have merged. AAL 3 / 4 provides for packet and frame-based data carriage in either a connection-oriented or a connectionless mode between AAL-service access point (AAL-SAP) across an ATM network and may distribute them in either a bidirectional point-to-point or a unidirectional point-to-multipoint manner. Frames of packets of data are split up by the AAL type 3 / 4 to be carried by one or more ATM cells as necessary. Thus, each SAR-PDU may be a cell containing either the beginning of message (BOM), the continuation of message (COM), or the end or message (EOM), where “message” means the data frame or packet. A two-bit Segment type (ST) field in the SAR-PDU header is used for this purpose. ST field may also be set to single segment message (SSM) where a single ATM cell is sufficient to carry the complete data frame, packet, or block. The multiplexing identification (MID) field allows a number of different AAL sessions to share the same ATM layer connection. Figure 2.9 describes the SAR-PDU format.

A sequence number field of 4 bits is allocated to carry out the modulo 16 sequence function of consecutive SAR-PDUs. The SN of a SAR-PDU would be incremented by one (modulo 16) compared to the previous SAR-PDU belonging to the same AAL connection.



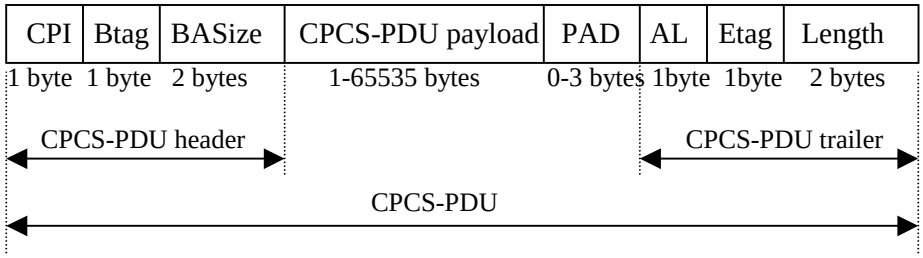
CRC	Cyclic redundancy check	SAR	Segmentation and reassembly
LI	Length indicator	SN	Sequence number
MID	Multiplexing identifier	ST	Segment type
PDU	Protocol data unit		

Figure 2.9: SAR-PDU format for AAL type 3 / 4

A six-bit length indication field (LI) is used to indicate how many of the octets within the SAR-PDU payload (total of 44 bytes) are actually being used to carry user information. The maximum value of 44 bytes will be used for BOM and COM SAR-PDUs. It should be noted that an LI value of 63 (or binary '111111') is used in the special case where an ST indicating an EOM leads to an abortion of partially transmitted CS-PDUs at the receiver. The CRC field is a 10-bit error detection code, which is used to detect received errors in the SAR-PDU header, payload, and LI fields.

The convergence sublayer (CS) of AAL type 3/4 converts variable bit rate (VBR) data streams into a format suitable for segmentation and reassembly by the SAR sublayer of AAL type 3/4 (Figure 2.10). The CPCS provides for transfer of data frames with any length between 1 and 65535 bytes. The common part indicator (CPI) may be used by different service specific sublayers to indicate different meanings of the CPCS-PDU header information, specifically the units for the values in the buffer allocation size (BASize) and length field. The beginning tag (Btag) and ending tag (Etag) are set to the same value in order to ensure that CPCS-PDU headers

and trailers are correctly associated with one another. The value is incremented for each successive CPCS-PDU. The buffer allocation size (BASize) indication field informs the receiving convergence sublayer entity of the maximum buffering requirements which will be required to receive the CPCS-SDU. The CPCS-PDU payload is required by the specification to be an integral multiple of 32 bits (4 bytes) in length, thus padding up to 3 bytes of 0's may be added to the end of the user information. The alignment (AL) field is used for 32-bit alignment of the CPCS-PDU trailer.



AL	Alignment	Etag	Ending tag
BASize	Buffer allocation size	Length	Length of CPCS-PDU payload
Btag	Beginning tag	PAD	Padding
CPCS	Common part convergence sublayer	PDU	Protocol data unit
CPI	Common part indicator		

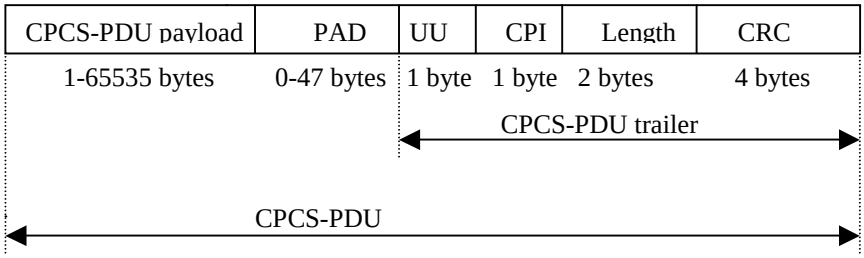
Figure 2.10: CPCS-PDU format for AAL type 3 / 4

CPCS of AAL 3/4 supports both class C and class D adaptation layer services. In class C mode, it provides a frame relaying telecommunication service and requires it to be used in conjunction with a special SSCS such as FR-SSCS. If used as a connectionless network access protocol in the class D mode, there is no need for a SSCS.

2.3.1.3.5 AAL type 5

The AAL type 5 was developed later than AAL 3/4 to serve similar traffic but was simplified in order to accommodate the trend toward high-speed data transfer. The main apparent difference from AAL type 3/4 lies in the fact that no additional overhead is added to the received SDUs at the SAR sublayer. Thus, all of the 48 bytes of each cell are available for

carrying the CPCS-PDU as a payload. Although this simplification leads to better line efficiency, capabilities such as multiplexing are not supported in the AAL itself but are left to the higher layer protocols to perform as needed.



CPCS	Common part convergence sublayer	PAD	Padding
CPI	Common part indicator	PDU	Protocol data unit
CRC	Cyclic redundancy check	UU	CPCS user-to-user indication
Length	Length of CPCS-PDU payload		

Figure 2.11: CPCS-PDU format for AAL type 5

Due to the drastic measure of doing away with any overhead at the SAR sublayer, the job of specifying the segment type is left to the ATM layer, specifically the least significant bit of the payload type (PT) field of the ATM layer cell header (Figure 2.2). This is called the ATM layer user-to-user (AUU) parameter. An AUU value of '0' indicates that the message segment is either a BOM or COM. An AUU value of '1' indicates that the message segment is either an EOM or SSM. This approach, although contributing to the nullification of the SAR-PDU header, and hence the overhead at that level, pays a price of level mixing and is an infringement of the protocol reference model (PRM).

The protocol information of the CPCS is all packed into an 8-byte trailer. The data block to be carried is carried in the CPCS-PDU payload. Padding (PAD) of between 0 to 47 bytes follows it to make sure that the CPCS-PDU has a size equal to an integral number of AAL type 5 SAR-PDUs. The CPCS user-to-user indication (CPCS-UU) field is used to carry certain information transparently on behalf of higher layers (e.g. a service specific convergence layer). The common part indicator (CPI) field is used in conjunction with layer management functions.

For the time being, it is merely used to align the CPCS-PDU payload field. The "length" field indicates the length of the CPCS-PDU payload field. It may be used by the receiver to check for any loss or gain of information during transmission. The CRC is a 4-byte CRC error detection code for detecting errors in the remainder of the CPCS-PDU. It is a standard CRC-32 code (the same as that used in Ethernet, Token Ring LAN, and FDDI).

For telecommunication end devices not specifically designed to operate over ATM networks, the SSCS is needed. In its simplest form, a simple encapsulation of information for carriage by the CPCS may be done. The Internet Engineering Task Force (IETF) used this approach [Cla96].

The advantage of class 5 AAL is that it has lower overhead compared to class 3/4 AAL. Because of this, it is better suited for higher data rate and multiservice traffic applications and thus promoted most by ATM Forum, a very influential consortium of companies (working group). However, a drawback may be that there is no error detection in the segmentation and reassembly (SAR) layer, which is provided in the case for class 3/4.

2.3.2 Wireless ATM Medium Access Control Technologies

The objective of the medium access control protocol is for one node to allocate efficiently a transmitted signal from two or more other nodes [BeG92]. The goal is essentially identical in nature for local area networks (LANs), satellite networks, and radio networks. There are two extremes among the many approaches researchers have attempted in order to achieve this objective. One end of the extremes is the "free-for-all," or random access, approach where any node having a packet to send does so "blindly" in hopes of being collision-free with packets sent from other nodes. Obviously, collisions are common in this case and the strategy implemented for retransmitting collided packets becomes an important issue. On the other extreme is the "perfectly scheduled," or fixed assignment, approach in which there is some order (in the time domain), a unique frequency band (in the frequency domain), or a unique pseudo-noise (PN) code (for spread-spectrum air-interface applications) assigned for each node. Random access protocols, such as ALOHA, can serve bursty traffic more efficiently compared to MAC protocols employing the fixed assignment approach, but the random nature of these protocols usually means there will be problems when quality-of-service (QoS), such as end-to-end delay and delay

variation, is taken into account. The fixed assignment approach has a permanent channel assignment during the course of a connection, but its efficiency in terms of throughput is very poor for bursty traffic.

Wireless ATM requires that certain QoS, such as maximum cell transfer delay, peak-to-peak cell delay variations, and cell loss rate, be satisfied. This requirement essentially precludes any random access protocols from being considered as the medium access control protocol. On the other hand, the bursty nature of the variable bit rate (VBR) service or the available bit rate (ABR) service makes the use of any fixed assignment techniques inefficient. Naturally, the only option feasible is some sort of compromise between random access and fixed assignment approaches. This compromise comes in the form of demand assignment MAC protocols. When a demand assignment MAC protocol is used in a wireless network, the channel capacity is distributed to users on a demand basis. The demand for bandwidth is implemented by the mobile units requesting a reservation whenever the need for transmission arises.

The demand assignment MAC protocol can be subdivided according to the two most popular multiple access techniques used, TDMA and CDMA. Each method has its advantages and disadvantages [RaW 94]. Section 2.3.2.1 summarizes the various demand assignment MAC protocols based on the TDMA multiple access technique reported in the literature. Section 2.3.2.2 describes some demand assignment MAC protocols based on the DS-CDMA multiple access control protocol that has been proposed in the literature. Based on the findings in Section 2.3.2.1 and 2.3.2.2, a comparison of the various MAC protocols is presented in Section 2.3.2.3.

2.3.2.1 Wireless ATM based on TDMA MAC Protocols

There are many variations of TDMA demand assignment MAC protocols for wireless ATM, and it is helpful to further categorize them into ones with polling MAC protocols [ATE95, MFM96, Aca96, ChS97] and the others with reservation-based MAC protocols [GVG89, BBF94, BDT94, QiL96, Ray96, ThG97, PaM98].

The polling protocols utilize the method where the base station (BS) takes control of distributing the transmission permissions to the mobile units. At the times the mobile units are polled, they can send either their request for transmission or the data themselves. Poll signals can also be used in setting various parameters to combat fading in the radio physical channel

layer [Aca96]. The uplink and downlink channels are usually separated in time division duplex (TDD), but there are some, including the work in [ATE95], where the frequency division duplex (FDD) method is utilized.

The reservation-based TDMA protocols all have a packet request phase to reserve a slot within a frame to send the actual data. The reservation can be explicitly specified at the beginning of each frame or it can be implicitly set for as long as there are messages waiting to be sent from the mobile unit. One of the representative papers of wireless ATM MAC protocols based on TDMA reservation is the multiservice dynamic reservation (MDR) TDMA proposed in [RaW94]. Here once again, the uplink and downlink are separated using the TDD method. In the uplink, the request slots provide a communication mechanism for a mobile station to send a reservation request, and message slots are used to send wireless ATM cells (essentially an ATM cell plus a wireless ATM header appended). Request slot packets are used for initial access and are sent in slotted ALOHA contention mode. Meanwhile, message slots on the uplink are filled with ATM cells with a data payload of 48 bytes. Acknowledgments for uplink request slot packets are given in the next downlink control packets. Also given in the downlink control packets are the announcements for downlink message transmissions and allocation for uplink message slots.

In general, the reservation-based TDMA protocols have better throughput compared to the polling-based TDMA protocols but with the price of having less flexibility in terms of admitting, for example, last-minute updates that may be needed for higher priority preemptive packets.

2.3.2.2 Wireless ATM based on DS-CDMA MAC Protocols

Spread-spectrum systems have been around since the mid-1950's. Most of the earlier works involved the military, where anti-jamming tactical communications, anti-detection from hostile forces, and cryptography were some of the major applications [Sch82].

Direct sequence system, which is short for "directly carrier-modulated, code sequence modulation," is the most popular form of spread spectrum systems [Dix94]. In the direct-sequence spread spectrum system, the original signal is mixed with a spreading code that is known only to the authorized sender and receiver, respectively. The sender would spread the

original code with the spreading code, also commonly referred to as the "Pseudo-random Noise (PN) code." This mixed signal would be difficult for the hostile forces to detect since the signal would have been spread across the spectrum and appear as random noise. The authorized receiver, however, would be able to decode the signal by mixing in the same PN code and correlating the signal. It is, however, acknowledged that a third party would be able to decipher a linear code such as a maximal length sequence code.

The DS-CDMA system builds on this basic direct sequence spread spectrum concept by taking advantage of the pseudo-random properties of the spread codes and enabling multiple streams of data to be sent simultaneously. The multiple signals would all use the same spread spectrum and each of the receivers would attempt to detect the assigned signal stream by using its unique PN code. The theoretical limitation on the number of multiple signal streams is set at the number of PN codes that could be generated that are orthogonal to each other. In practice, complete orthogonality is seldom kept in order to increase the number of code sequences available.

It should be noted that the WATM has not been standardized yet and research is still comparatively in its infancy. There are many open issues to be dealt with including how to deal with the various kinds of fading, multipath, and shadowing inherent to the wireless communication environment. Fading is one of the major causes of bit errors in wireless communication. The DS-CDMA air-interface technology has a well-supported reputation for suppressing the negative effects of multipath and some aspects of fading; therefore it is one of the candidates for the air-interface MAC protocols to be used in a WATM network.

Various DS-CDMA techniques used for commercial applications are explained in [AzA94, Agh94, ArL96]. The authors categorize the different techniques according to different mappings. The pure CDMA approach is the simplest method, where all signals with different bit rates are spread over the whole allocated bandwidth. Although this method would certainly work for lightly loaded situations, as the load increases the multiple access interference increases and link quality degradation occurs. FDM/CDMA and TDM/CDMA systems are also described, but, as pointed out earlier, the problem of a relatively low maximum data rate that a particular mobile unit can send or receive is still not resolved. Finally, a multi-code DS-CDMA concept is presented where a multiple sub-stream structure is used to support high rate service by dividing

the original traffic into multiple sub-streams transmitted in parallel over the wireless channel. The multi-code DS-CDMA concept is described in various articles ([IPO95, IGi95, INa96, LKZ96, WHT97, Hue00, FMW00]) but with varying dynamic assignment schemes. The methods proposed in [LKZ96] and [FMW00] are addressed and compared below.

Authors in [LKZ96] realized that, although the multi-code DS-CDMA structure in itself was of much value in terms of resolving the required high bit rate problem, the quality of service (QoS) could still suffer due to multi-users sending too much data simultaneously. Hence, the demand assignment access control using the Distributed Queuing Request Update Multiple Access (DQRUMA) protocol was incorporated into their version of the multi-code DS-CDMA wireless packet system. Their main focus was to combine the advantages of the random access and fixed assignment methods, and utilize the given channel as much as possible. The random access portion of the DQRUMA protocol assumed a slotted-ALOHA scheme, where there were a fixed number of transmitter receiver pairs allocated for the access requests. A piggyback request field was devised to minimize the contention encountered in the access request phase by requesting for access for the next time frame contention-free whenever there were packets remaining in the queue after sending the data packets in the current time frame. The BS collected all the requests for access from each of the MUs in the system and provided transmit permissions to each of the MUs sequentially until the maximum capacity of the channel was deemed consumed. This DQRUMA multi-code DS-CDMA system seems to be the most sophisticated form of air-interface available for DS-CDMA WATM systems.

A new link-layer transmission strategy using multi-code CDMA was introduced in [FMW00] for the third generation wireless CDMA, but the concept portrayed in the article is generic enough for application to WATM and thus included here. The authors implemented a two state Markovian channel, similar to the Gilbert-Elliott channel ([Ell63]) where there are “good” and “bad” states. The “bad” state is entered when a packet is first received in error at the BS. The “good” state is entered when a packet is received correctly for the first time after being in a bad state. Here, it is assumed that in the “bad” state no communication is possible. The multiple codes are used to recover from gaps on the wireless link after a fading period by using an automatic repeat request (ARQ)-type retransmission. It was assumed that the number of possible CDMA codes within one cell was much higher than the number of active users. Thus,

each wireless terminal (WT), after registering with the BS, was issued a unique set of PN codes from the BS. To maximize the SNR value, the sender side data link entity aborts transmission of data packets of a data link segment if the segment is likely to fail to meet the prespecified delay bound. Four types of simultaneous MAC packet transmission (SMPT) schemes were tested against an ordinary single stream sequential transmission scheme. The four SMPT schemes examined were Slow Start, Fast Start, Slow Healing, and Fast Healing. These are categorized into either a Self Healing or a Start Up mechanism. The Self Healing mechanism is used only if sequential transmission (using one channel) falls behind in time due to channel errors. The Self Healing uses the multi-code capability to send on additional channels after the "bad" periods. Fast Healing uses all available resources immediately after detecting a "good" channel state, while the Slow Healing uses the resources incrementally. The Start Up mechanism utilizes the additional channels whenever the channel is in a good state. Fast Start transmits on all channels, while Slow Start transmits incrementally. The results show that Slow Start and Slow Healing mechanisms performed the best overall in terms of throughput.

SMPT mechanisms in [FMW00] differed from DQRUMA mentioned in [LKZ96] in that each WT (or MU as termed in [LKZ96]) used the multiple channels at its own discretion, while the DQRUMA protocol relied on the BS to handle the number of multiple channels that could be used for each MU. The advantage of using SMPT mechanism was that delay and jitter performance could be adjust in a more timely fashion than the DQRUMA method. However, the DQRUMA method had the advantage of being able to control the total amount of traffic being admitted to the network, thus minimizing the packet transfer errors over the wireless link.

2.3.2.3 Comparison of Wireless ATM MAC Protocols

The question of whether to use a TDMA-based MAC protocol or the DS-CDMA MAC protocol has been a hot issue in various wireless network applications and it is no exception in the wireless ATM environment. Few in-depth works are known that compare between TDMA- and DS-CDMA-based WATM MAC layers. There is, however, an interesting work done by [WGJ93] that performs a comparative study of a packet DS-CDMA system and a Dynamic-TDMA (D-TDMA) network with integrated voice and data in a personal communication network (PCN) environment. Although the setting is a bit different than for a wireless ATM system, the

insight into the wireless interface issues make it worthwhile to pursue. This article assumes an integrated voice and data PCN environment.

The D-TDMA protocol described here follows the basic structure of a demand assignment TDMA MAC protocol where a frame is divided into request slots, voice slots, and data slots. The basic channel access scheme follows a combination of "circuit mode" reservation of slots over multiple TDMA frames for voice calls, and for data messages, a dynamic first-come-first-served assignment of remaining capacity for data messages is performed. A reservation for channel assignment is made in random access (slotted ALOHA) mode using a randomly selected request slot (among the N_r request slots). The request packet can encounter a collision, and, in this case, does not receive acknowledgment. For voice call requests that are not acknowledged for maximum waiting time for access $(W_{v\max})$, the call is "blocked." Data users do not encounter blocking, but may continue retransmissions of requests until an acknowledgment is received.

The packet DS-CDMA considered here uses a sequence of fixed-length packets on the channel for both voice and data. Access to the channel is completely unconstrained, and a user begins transmission with a "receiver directed" PN code whenever it is ready to send data. Data traffic is sent out as a single contiguous burst at the available peak DS-CDMA channel speed. On the other hand, voice or other constant bit rate (CBR) stream traffic is sent as a periodic sequence of packets with delay cycle adjusted to match the requirements of the CBR source.

The authors in [WGJ93] find that the DS-CDMA has significantly higher system bandwidth efficiencies (2 or more times than the D-TDMA MAC protocol). One of the reasons given for the superior bandwidth efficiency for the DS-CDMA system is due in part to the fact that, unlike the TDMA system, where a "cluster" composes N_f separate frequency bands for frequency coordination between cells, the total bandwidth is available for spread-spectrum operation. This implies that the DS-CDMA user has the advantage of being able to use "soft hand-off" instead of the more traditional "hard hand-off" when migrating over cell boundaries. Moreover, it is mentioned that packet DS-CDMA provides random access operation with near-perfect statistical multiplexing of traffic and also that this system exhibits the unique features of zero channel access delay and near-perfect statistical multiplexing of any arbitrary traffic mix. However, the DS-CDMA method could expect longer delay for data services such as file transfer

and multimedia due to peak transmission speed limitation. Also, increasing traffic load in the DS-CDMA system will have the effect of degrading packet loss rate (degraded voice quality). In a TDMA-based system, this will lead to higher probability of voice blocking. Here, we can see that the CDMA system has the advantage of being able to handle degradation of the wireless channel "gracefully" while TDMA is only able to make or reject a connection.

Taking into account the discussions given in Sections 2.3.2.1, 2.3.2.2, and this section, we come to a conclusion, that despite the many advantageous characteristics of DS-CDMA, the biggest obstacle preventing the use of DS-CDMA as the MAC protocol for wireless ATM seems to be that the maximum data rate from a mobile unit is inherently limited due to the spreading of frequency done in a DS-CDMA system. In light of this fact, we have concluded that the multi-code CDMA MAC protocol introduced by [LKZ96] has the promise to be able to take advantage of most of the merits of using a DS-CDMA system and to serve as the MAC protocol for wireless ATM networks.

It is, however, noticed that most of the works in the literature focus more on providing optimum performance in terms of delay and throughput without addressing the issue of how to deal with situations arising from burst errors characteristic of the wireless link due to fading or shadowing. Optimal performances claimed by these works can only be achieved under "normal" settings. The APMS-CDMA MAC protocol developed for this research extends upon the studies done in [LKZ96] by assuming a more realistic situation that could be represented by clusters of bit errors occurring from time to time such as described in [Gil60, Ell63].

2.4 Summary

This chapter presented the background related to the MAC protocols proposed for wireless ATM. It was noted in Section 2.1 that wireless ATM is a relatively new concept whose standard has not yet been mandated, unlike the IEEE's 802.11 (Section 2.2.1) and ETSI's HIPERLAN (Section 2.2.2). In Section 2.3.1, the ATM itself was looked into to appreciate the functionality of the ATM network, and various proposed wireless ATM MAC protocols were probed and compared. In Section 2.3.2, various proposed wireless ATM MAC protocols were investigated. It was concluded that, although TDMA-based MAC protocol has received more attention, DS-CDMA-based MAC protocols had many advantages that could make it a better

choice. In Section 2.3.2.2, special attention was given to the work done in [LKZ96], where the one disadvantage (the low maximum data rate) associated with the conventional DS-CDMA was overcome by structuring multiple substreams within a mobile unit. Section 2.3.2.3 compared the two main air-interface methods used for the wireless ATM MAC protocols, the TDMA and the DS-CDMA. The comparison work concluded that, among proposed DS-CDMA MAC protocols, and perhaps among all proposed MAC protocols, multi-code DS-CDMA in [LKZ96] had the potential to be one of the better wireless ATM MAC protocols. It was noted that this research extends upon the basic structure provide in [LKZ96] so that effects of link degradation are taken into consideration to provide a more robust MAC protocol for wireless ATM.

Chapter 3: Research Objectives and Methodology

This chapter discusses the objectives and the methods employed to achieve the objectives. Section 3.1 states the problem definition and objectives to be attained. Section 3.2 lists some of the assumptions made to limit the scope of the research so that an investigation of the proposed method of research can be evaluated more thoroughly. Section 3.3 states the performance metrics to be used to evaluate the effectiveness of the newly proposed method.

3.1 Problem Definition and Objectives

Compared with wired ATM, wireless multiple access and mobility management are two major topics for implementing WATM. This research focuses on improving the wireless multiple access aspect of WATM.

3.1.1 Problem Definition

In Chapter 2, we discussed the background technologies upon which WATM is built and it was noted that, due to inherent susceptibility to fading and shadowing, WATM's performance can sometimes fluctuate. Since ATM was originally developed without much regard to wireless access consideration, the BER and reliability of the physical network was presumed to be quite low and very high, respectively. This presents a problem when migration to WATM is considered. First of all, the expected, and thus tolerated, BER in the wireless ATM network can

be as high as around 10^{-3}, whereas in the wired ATM network the BER is usually 10^{-9} or less. Thus, it is inevitable that WATM has additional overhead in the form of a WATM header appended to the ATM cell to provide space for more channel coding, making it more robust against bit errors caused by the radio channel and thus lowering the packet error rate (PER).

However, the more serious consequence of the relatively hostile environment of a radio channel is that during periods of deep fading or shadowing, there may be a drastic deterioration in the bit-error rate experienced by the receiver, which will necessitate a more powerful measure. This is the key area of interest for this research.

3.1.2 Research Objectives

The foremost objective of this research is to determine how to handle the periods of deep fading and shadowing (the “bad” mode) in a way that probability of bit error is reduced and the overall throughput in the network increased. A subsequent objective is how to determine when the environment should be deemed "bad" so that this special handling has to occur, and just as important how the release from the “bad” mode to the “good” mode should be done. In this research, a WATM with multi-code CDMA (MC-CDMA) air-interface, as introduced in Chapter 2, will be used as the basic structure for its WATM MAC protocol study. The network measurements will be done only for uplink (i.e., MUs to BS traffic) since that is where the bottle-neck is using an infrastructure architecture where the BS functions as a master that has all the knowledge concerning the MUs surrounding it.

Under “good” circumstances, the multiple sub-streams would be effectively used in transferring the data by allocating *different* portions of the source data to each of the multiple sub-streams. However, as soon as the system perceives that the wireless channel is undergoing a drastic decline in reliability, one of several rake-in modes is automatically set by the BS to carry the *same* portion of the source data several times (e.g., 2, 3, or more replications). This increases the chance that the data will get through without error. The advantage of this method is that the existing hardware structure is utilized and it does not require complex decoding infrastructure, deployment of which is discouraged in a mobile unit. This method has its disadvantages in that more channel bandwidth is consumed for the same amount of data being transmitted.

Thus, deciding when to transform to a duplicate sub-stream mode, which we will call the "rake-in" mode from this point on, is just as important as the implementation itself. The criteria should be set carefully so that the best possible throughput is achieved. A method is devised where transmitted errors are monitored and compared against a set of parameters that will determine whether the network should be in the normal operating mode or in the rake-in mode.

3.2 Network System Under Investigation

The following sub-sections list the assumptions made about the network in this research effort. Assumptions enable a realistic yet tractable model for the network under study.

3.2.1 Network Topology

Since the purpose of this research is to show a relative improvement in throughput performance within a given DS-CDMA WATM network, a single cell assumption is made. A single WATM cell contains one BS and multiple MUs.

3.2.2 Number of MUs

The number of MUs that are within a WATM cell has a significant effect on the performance of the system. Variation in the number of MUs in the system affects the load in the system and the performance in the contention mode also depends on the total number of MUs. The number of MUs is varied from 5 to 90 depending on the context of the application. For instance, one set of simulations fixes the total number of active MUs to 5, but then varies the load within each MU to vary the total load in the system. Another set of simulations fixes the load within each MU but varies the number of MUs in the system from 10 to 90 in increments of 10 MUs to implement 0.1 to 0.9 normalized load in increments of 0.2.

3.2.3 Normalized Offered Load

Normalized offered load is the amount of traffic introduced into the system relative to the total system capacity. Levels 0.1, 0.3, 0.5, 0.7, and 0.9 are used as reference points for all of the

simulations. The MUs in the system have identical traffic characteristics and load. The normalized offered load is the combined load offered by all the MUs in the system.

3.2.4 Channel Model

The channel model follows the assumptions widely accepted for DS-CDMA systems, which are the interference model under Additive White Gaussian Noise (AWGN) [Pur77, PSM82, MoL89, GeG91, Rap96], with burst error characteristics of the Gilbert-Elliott model [Gil60, Ell63, WiM99].

3.2.5 Traffic Model

The assumed traffic model will encompass voice (a CBR), video (an rt-VBR or an nrt-VBR), and bulk data transfer (an ABR), essentially enabling the multimedia traffic for each mobile unit. Among the four disciplines, the CBR and rt-VBR traffic are categorized as real-time traffic, while the nrt-VBR and ABR traffic are categorized as non-real-time traffic. For all the simulations in this study, the QoS of real-time traffic are met by allowing only the those packets that are within 20 mS since its creation to be transferred. The QoS of non-realtime traffic will be met by ensuring that every means is attempted to lower the PER, while having a more lenient time constraint of 1 second. Allowing retransmissions up to the MAX_RETRAN limit improves upon the chance that the packet error rate will be less than the 10^{-3} limit. The stochastic models for these traffic types are addressed in more detail in Chapter 4.

3.2.6 Propagation Delay

Since WATM assumes a microcell-like environment [KrN96], the propagation delay is assumed to be zero with respect to packet transmission time.

3.2.7 Frame Length

A single frame consists of an AR packet, a PGBR packet, and a Data packet. The AR packet is 31 bits, the PGBR packet is 15 bits, and the Data packet is 511 bits in length. The time

it takes to send a frame is referred to as the frame time. The Data packet includes the BCH error correction encoding inside its wireless ATM header. Detail of the derivation of this value is presented in Chapter 5.

3.2.8 Queues in MUs

As part of the QoS enhancement scheme, the queues in the MUs are subdivided into four subqueues, each of them representing each of the four types of traffic (CBR, rt-VBR, nrt-VBR, and ABR).

3.2.9 MAC and Physical Layer Implementation

As mentioned in the discussion given in Chapter 2, the multi-code CDMA structure will be used as the physical layer implementation. The APMS CDMA protocol will be used as the MAC protocol. The DQRUMA MC-CDMA MAC protocol [LKZ96] will be used as the conventional protocol against which our results will be compared.

3.3 Performance Metrics

This research effort deals with an infrastructure model, where the central BS possesses all the information regarding the state of the mobile units. Therefore, the downlink transmissions are more easily facilitated than the uplink. Thus, we only investigate the uplink performance, where there are greater challenges. Throughput is almost always the most important measure of performance in a computer network [Jai91], and this research effort follows the convention. For this research effort, relative throughput is measured in terms of the total number of packets received correctly versus the total number of packets that were generated from the transmitting side. However, due to the widespread use of applications dealing with real-time data, the delay and the number of dropped packets incurred by each type of traffic will also be investigated. The delay variation (jitter) metrics, however, will be assumed resolvable through proper buffering of real-time data, given that the values are within maximum tolerable delay. Although the conventional DQRUMA MAC protocol, from which this dissertation springboards, does not incorporate any retransmission scheme, for fair comparison, retransmission factor will be varied

equally for both DQRUMA and our APMS MAC protocols. Thus, the number of packets exceeding the maximum retransmission limit will also be part of the performance metrics examined. The performance metrics are measured while keeping the QoS constraints, as described in Section 3.2.5, for real-time and non-real-time traffic.

3.4 Summary

Section 3.1 presented the problem definition and objective of this research effort. The harsh reality faced by the wireless link was described to enlighten the need for a more robust wireless ATM MAC protocol that had the ability to adapt to the changing conditions of the channel. Section 3.2 defined the assumptions made about the network under investigation. In Section 3.3, throughput, delay, the number packets dropped, and the number of packets exceeding maximum retransmission limit, were identified as the metrics by which the performance of the newly developed protocol will be measured.

Chapter 4: APMS-CDMA Wireless ATM MAC Protocol

This chapter describes the Adaptive Parallel Multiple Substream CDMA (APMS-CDMA) MAC protocol developed as a result of this research effort. APMS-CDMA encompasses the whole MAC protocol and a portion of the physical layer as reconfiguration of the rake-in receiver is needed. Any CDMA air-interface is, to a certain extent, involved in the MAC layer as well as the physical layer, so this is not unusual. The APMS-CDMA MAC protocol adjusts the necessary facilities of the two layers to enhance the overall throughput of a WATM network. Section 4.1 introduces some of the harsh realities of the wireless network channel characteristics and justification for the introduction of APMS-CDMA MAC protocol. Section 4.2 describes the hardware structure of the APMS-CDMA wireless ATM network. Section 4.3 illustrates the bandwidth on demand access scheme used. The criterion for the various rake-in modes used to enhance the performance is discussed in Section 4.4. The performance improvement brought about by the rake-in mode is analyzed and a bit-level simulation is done to verify the correctness of the analysis in Section 4.5. The multimedia traffic characteristics and allocation method to be employed is introduced in Section 4.6. The retransmission scheme at the MAC layer is described in Section 4.7.

4.1 Link Environment in a Wireless ATM Network

The most noticeable demerit that a wireless network suffers in comparison to a wired network is the bit errors. This is especially more so in a WATM network, as depicted in Figure

4.1. When ATM networks were first developed, a reliable physical layer medium, such as SONET, was assumed. This assumption is not true in a wireless arena, where an increase of several orders in bit error rate is common. The bit errors are caused by many factors, including multipath fading, shadowing, co-channel interference, and adjacent channel interference. It should be noted that bit error rates in excess of a certain level degrades the network throughput, not linearly, but exponentially. This is due to the fact that a reasonably applicable channel coding method can only correct a few bit errors in a packet, and any errors bits in excess of the correctable bits must be retransmitted for any conventional computer network data.

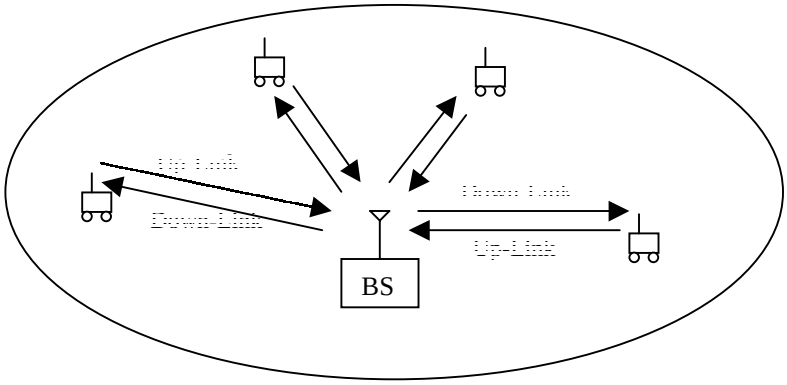


Figure 4.1: Wireless ATM Network Cell

If the wireless channel is well behaved, a retransmission scheme will suffice for conventional computer network data. However, in hostile environments, the network with retransmissions will trigger more influx of data. The result is more crowding of the limited bandwidth of the network, and further degradation of the wireless network environment, causing more bit errors. In an extreme case, a viciously degrading cycle leading to serious network congestion may occur. It becomes even more challenging for real-time applications where no retransmission is allowed. In this case, once packets are lost there is no way to recover. For interactive multimedia applications, the user will experience jumbled noise and/or a static video screen for a period of time. Thus, the resulting effect for both real-time and non-real-time traffic is that in a wireless channel environment there will be periods of burst errors, where the wireless link will be very faulty compared to its normal state. The burst errors are the result of variations in relative path losses and shadowing. These large-scale fadings can be modeled with log-normal distribution, and they will be generally slow enough to be controlled [GJP91]. Moreover, large-

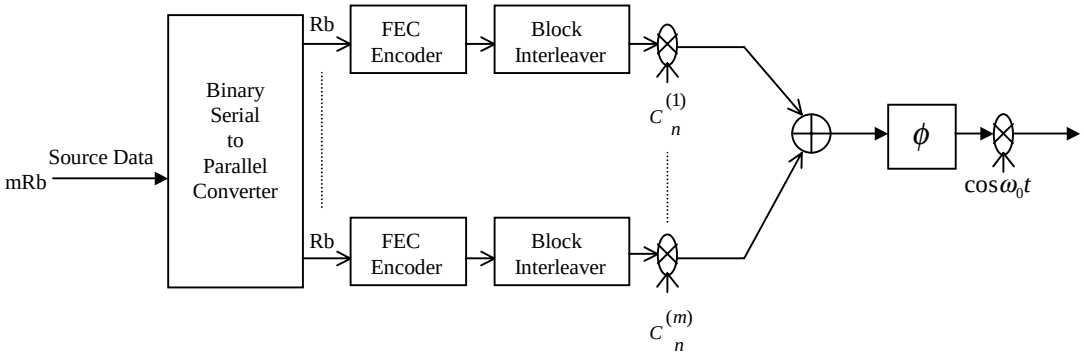
scale fading exhibits reciprocity, which means if link distress is experienced in the down-link, it is highly likely that the uplink is also in a state of great distress. This is where APMS-CDMA can help improve the overall performance in the wireless network.

The key feature in APMS-CDMA is the ability to change the usage of the substreams according to the severity of the air-interface conditions. In normal well-behaved mode, the substreams are loaded with *different* portions of the current packet at hand to enhance the data rate between the BS and the mobile unit. However, as soon as the system perceives that the wireless channel is undergoing a drastic decline in reliability, one of several rake-in modes is automatically set by the BS to carry the *same* portion of the source data several times (e.g. 2, 3, or more replications). If the substreams are converted to the rake-in mode, the probability of bit error will decrease substantially depending on the number of duplicate data employed for the rake-in mode. Consequently, the resulting data bit rate will also be lowered in proportion to the number of the duplicate packets being sent. Thus, APMS-CDMA protocol should resort to the rake-in mode only when the bit error rate exceeds a carefully chosen set of criteria.

In the APMS-CDMA protocol, the criteria setting mechanism to choose between the "normal" mode and "rake-in" mode is of crucial importance. The window-limited past history of the detected errors will be used as the source and will be analyzed. Based on the analysis, the APMS-CDMA protocol will determine whether it should be in the "normal" mode or switch to one of the "rake-in" modes, and vice versa. The details of the criteria for determining the modes are addressed in Section 4.4.

4.2 Structure of APMS-CDMA Wireless ATM Network

The two main constituents of the APMS-CDMA WATM network are the mobile units and the BS. There are multiple mobile units for a BS in a wireless ATM network cell. Each mobile unit has the capability to transmit its source data through *m* multiple sub-streams, as shown in Figure 4.2. The receiver part of the APMS-CDMA mobile unit, which is also capable of receiving *m* multiple sub-streams, is described in Figure 4.3. However, the BS decides the actual number of multiple sub-streams that are allowed to be active for each link between the mobile unit and the BS, based on the network reliability that is constantly being assessed.



mRb: Source Data Rate

Rb: Data Rate for each Sub-stream

$C_n^{(1\sim m)}$: Spreading (PN) Codes

ϕ: Power Control Variable

Figure 4.2: APMS-CDMA Mobile Transmitter Structure

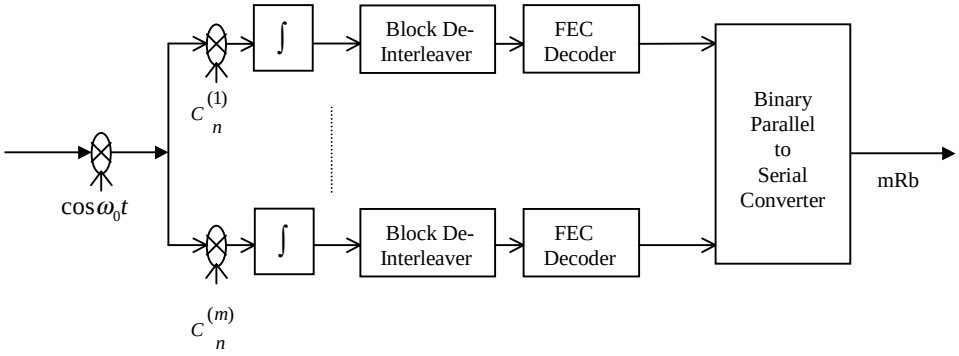


Figure 4.3: APMS-CDMA Mobile Receiver Structure

For each mobile unit, the BS will issue a unique, primary, pseudo-noise (PN) code. Upon receiving the primary PN code, the MU will use the Walsh functions to generate *m* sub-streams of data that are orthogonal to each other [IGi95]. The method is described in the following equations.

If we let C_n^{PN} to be the primary PN code, then we can generate *m* orthogonal spreading codes by using the following multiplication,

$$C_n^{(i)} = C_n^{PN} \times W_i \qquad (4.1)$$

where W_i's for varying values of i are from a set of orthogonal Walsh codes. Since, W_i is orthogonal to W_j for values $i \neq j$, $C_n^{(i)}$ is orthogonal to $C_n^{(j)}$ for $i \neq j$ [IGi95]. The additional

advantage of having orthogonal codes at a mobile unit is that this orthogonality is maintained at the receiver since the propagation properties are the same on the parallel codes [LKZ96]. The result is that there is no self-interference among substreams originating from the same MU.

4.3 Bandwidth on Demand Access

Since all the mobile units admitted into a system do not use the full capacity available all the time, the available channel capacity can only be fully exploited by employing the bandwidth on demand (BoD) access method. Much of the BoD access technique employed in our system follows the DQRUMA protocol mentioned in [LKZ96], but with the necessary changes made to the protocol to enable our proposed adaptive rake-in mechnism. In order to achieve BoD access, there are several steps that must be followed. Section 4.3.1 presents the Access Request phase from the mobile unit transmitter. Section 4.3.2 explains how the Access Request Acknowledgment is sent back from the BS. The piggyback request is explained in Section 4.3.3. The mechanism behind the Transmit Permission and Acknowledgment rendered by the BS is described in Section 4.3.4. The overall flow of the BoD access mechanism is presented in Section 4.3.5.

4.3.1 Access Request

The Access Request (AR) packet is sent from the MU to the BS when there is a need for the first transfer of data. The criterion for the "first transfer" is when a series of packets arrive at an empty queue in the mobile unit. A pool of PN codes is reserved for this access request phase. This is described as “transmitter-receiver” pairs in [LKZ96], and we follow the specification of there being 25 “transmitter-receiver” pairs available for the contention-based access request phase. Each MU having a packet(s) to send attempts to transmit its AR packet to the BS using one of the PN codes in the pool. Walsh code W_1 is used in association with the given primary PN code to generate the PN code to be used in BoD access request to the channel. It should be noted that because Walsh code W_1 is used does not suggest only the access request for the first sub-stream is conveyed, but rather W_1 is used in the access request for all the sub-streams in the mobile unit. The Access Request packet comprises three fields as described below.

The MID (Mobile ID) field identifies the MU that is requesting access. The MSS (Multiple Sub-streams) field specifies the total number of sub-streams that the MU intends to use. The MSSRT (Multiple Sub-streams for Real-Time traffic) field specifies the number of real-time packets that needs to be sent out during the next time frame. It follows that the value of MSSRT is always less than or equal to the value of MSS. The concept of specifying the real-time traffic needs is unique to our system, which has been implemented to give higher priority to real-time traffic in terms of acknowledge response. Figure 4.4 shows how the APMS-CDMA channel is structured in the Access Request mode.

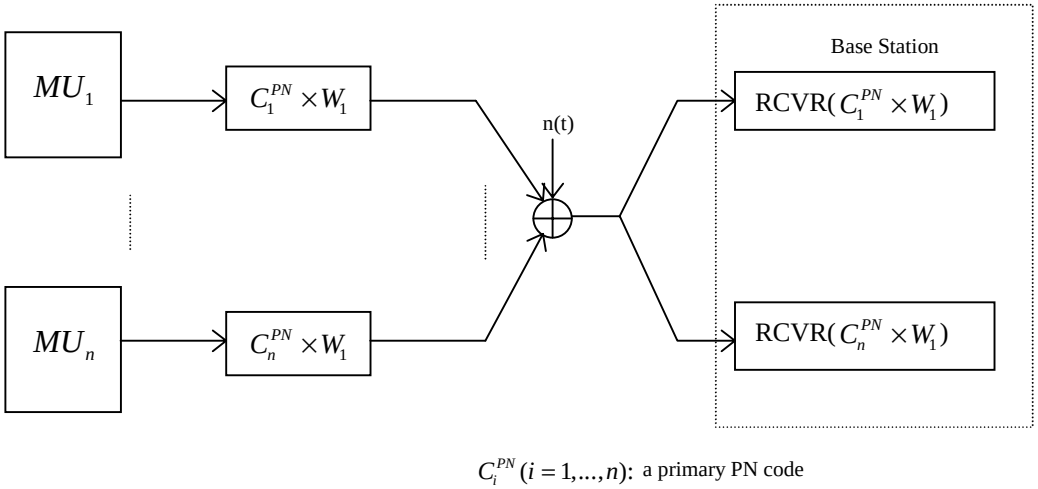


Figure 4.4: APMS-CDMA Channel Structure in the Access Request Phase

4.3.2 Piggybacked Access Request

Once the queue inside the mobile unit is foreseen to be filled with more than one packet after an MU sends out a packet(s), the request for access in the next time slot is done automatically by the Piggybacked Access Request (PB_Req) packet. The advantage of having this feature is that all the potential access requests that would otherwise have depended solely on the access request packet could be taken care of contention free by this PB_Req packet, thus lowering the potential bit error rate in the access acquisition process. Another likely advantage is that, when a mobile unit uses more than two multiple sub-streams for its transfer mode, the probability that the PB_Req packet will suffer from bit errors is greatly reduced since the contents would be the same and there would be code diversity among different substreams.

4.3.3 Access Request Acknowledgement

The AR packet sent from an MU, as mentioned in Section 4.3.1, must contend with the AR packets sent from other MUs. There is a collision when two or more MUs send AR packets using the same PN code. In that case, none of the AR packets will be discernable, and thus, no acknowledgment will be sent to the MUs involved in the collision. However, when a valid AR packet is received from an MU with a unique PN code, an Access Request Acknowledgment (ARA) control packet is issued by the base station on the forward link to let the MU know that the access request was received successfully.

The BS sends the ARA packets using the PN codes the mobile units used in the access request phase. The APMS-CDMA Access Request Acknowledgment transmitter structure at the base station is shown in Figure 4.5.

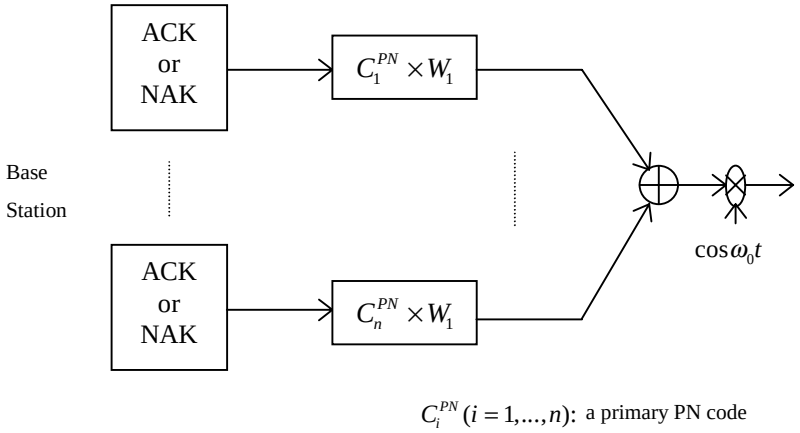


Figure 4.5: APMS-CDMA Access Request Acknowledgment Transmitter Structure at the Base Station

4.3.4 Transmit Permission with Packet Transmission Acknowledgment

The BS keeps track of all the Access Requests, makes the decision on the following issues for each MU, and places that information in the Transmit Permission packet. First, the BS decides how many multiple substreams in each of the mobile units are allowed (MSSA) in the next transfer. Second, it decides how many of those MSSA are multiple substream real-time allowed (MSSRTA) for the next transfer. Third, the rake-in level (RL) that is to be used by the

MU in the next uplink is indicated. These decisions are made in accordance with the total channel capacity allowed and the estimation of the channel states done at the BS. (Detailed explanation of the Transmit Permission allocation is given in Section 5.5.) The channel state estimation record is kept for each MU separately. Finally, the acknowledgment for the previous packet transmission from the mobile unit is specified. The transmission acknowledgment will be given by field M that is $m_{\max}$ bits long, where $m_{\max}$ is the maximum number of multiple sub-streams that is sent from any of the mobile stations. The least significant bit (LSB) of M represents the first substream tested while the most significant bit (MSB) of M represents the final substream tested. A bit that is returned with binary 1 means the data sent through the corresponding sub-stream was received correctly. A binary 0 means that either no data was sent by the MU or the receiver at the BS did not receive the data correctly.

The PN codes used for Transmit Permission with Packet Transmission Acknowledgment (TPPTA) packets are the same ones chosen for the uplink transfers. That is, the TPPTA packets generated in response to the AR packets will use the PN codes that were used for the AR packets. On the other hand, TPPTA packets that are generated as a result of the uplink data traffic will adopt the PN codes used for the uplink traffic.

Figure 4.6 describes the configuration of the channel during the Transmit Permission and Transmission Acknowledgment.

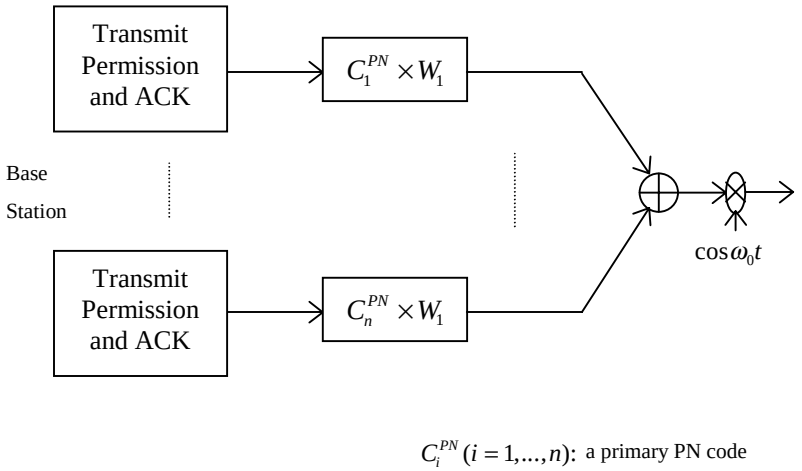
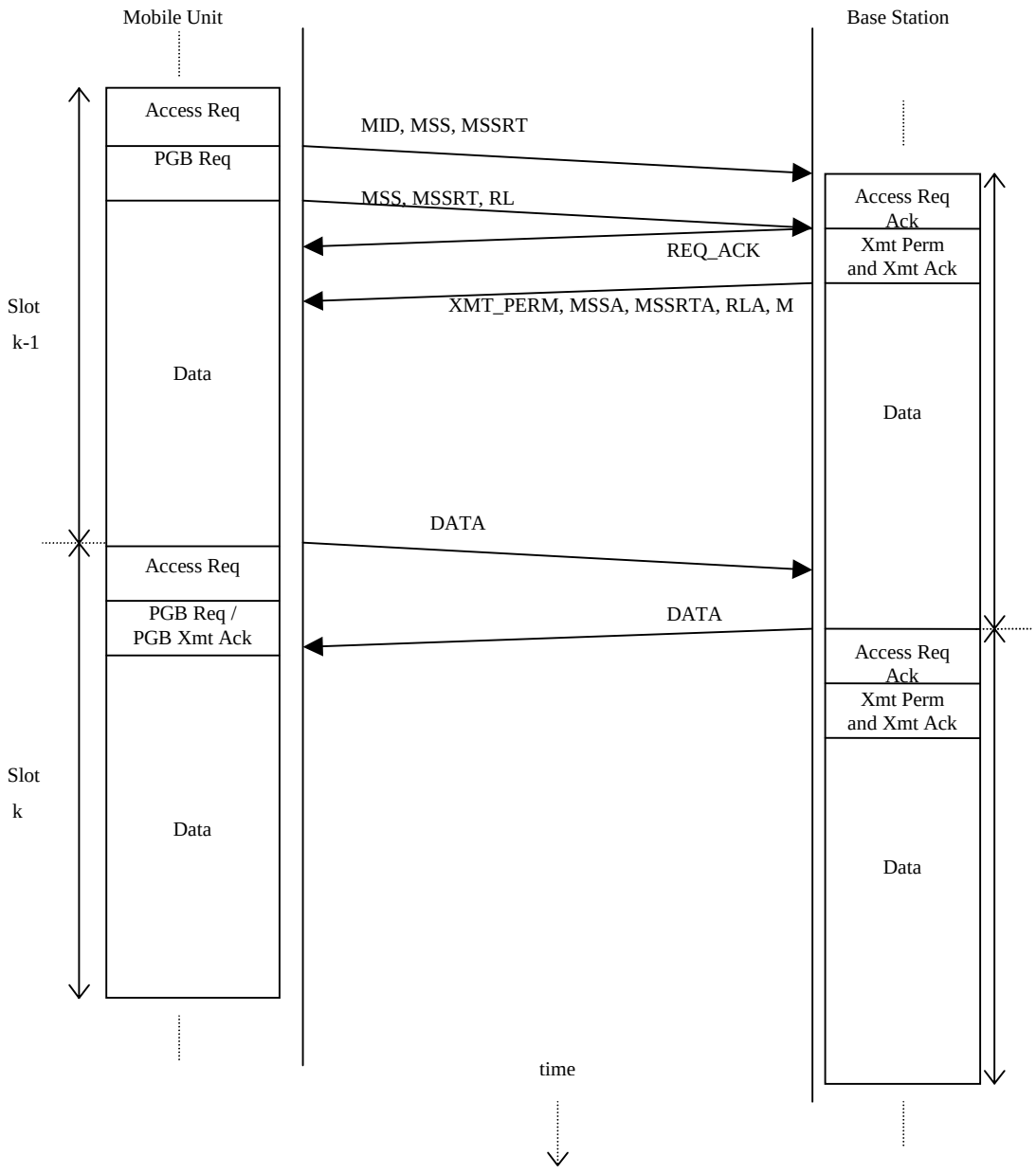


Figure 4.6: APMS-CDMA Transmit Permission with Packet Transmission Acknowledgment Transmitter Structure at the Base Station



MSSRT – Multiple Substreams for Real-Time

MSSRTA – Multiple Substreams for Real-Time Allowed

RL – Rake-in Level

MSS – Multiple Sub-Stream

RLA – Rake-in Level Allowed

MSSA – Multiple Sub-Stream Allowed

M – Transmission Acknowledgment Vector

REQ_ACK – Access Request Acknowledgment

MID – Mobile ID

Xtm_Perm – Transmission Permission

Figure 4.7: APMS-CDMA the Protocol Flow of the Bandwidth on Demand (BoD) Access channel

4.3.5 Protocol Flow of the Bandwidth on Demand Access Channel

Figure 4.7 illustrates the protocol flow of the Bandwidth on Demand (BoD) access channel, as presented in Sections 4.3.1 to 4.3.4. The left to right arrows describes packets generated from the MUs, while the right to left arrows describes packets originating from the BS.

4.4 Criteria for Various Rake-in Levels

As mentioned in Section 4.1, a wireless network suffers a greater bit error rates in comparison to a wired network. Thus, it is not unusual for a wireless channel to become unreliable using conventional methods of any wireless access technology. Fortunately, the multiple sub-stream structure considered in this research provides us with the necessary hardware infrastructure to alleviate this problem. A rake-in mode concept was described in Section 4.1.

An important issue, then, becomes what the criterion should be for declaring the wireless channel to be in a distressed state. This criterion is of critical importance because the overall throughput of the system, to a great extent, depends on the accurate assessment of the channel conditions because the “rake-in” mode that is used during the “bad” periods uses up more channel capacity than the “normal” mode where no “rake-in” mode is used. There are two factors that drive the solution toward a single answer. One factor to be considered is that a false alarm is worse than one being called too late. That is, if a lone error packet causes the BS to declare a “bad” state for a particular channel, many times this will create too many instances that call for the “rake-in” mode, when in fact it was just a single packet that happened to be in error. The other factor to be considered is that we do not want more than two time frames to pass with “bad” packets, since error packets in two frames in a row is clearly an indication of a “bad” channel state. Thus, the “triggering” criterion for the rake-in mode is set to more than 60% of packets being in error over two consecutive time frames. Pilot simulations indicated that evaluating over two consecutive time frames resulted in best performance for each combination of simulation factors for the APMS protocol. This criterion for “triggering” the rake-in modes was used throughout the simulations.

Another important issue is determining the criterion for the “releasing” the channel from the “rake-in” mode. This issue is not as clear-cut as the “triggering” for the “rake-in.” This

would depend largely on the characteristics of the channel. Yet, considering the fact that two time frames are expended before a "rake-in" mode is declared, based on pilot studies, it was shown that a minimum of 20 time frames were needed to overcome the overhead incurred for the "rake-in" mode triggering. This is because we do not want a situation where there would have to be multiple transitions to and release from "rake-in" modes. On the other hand, any unnecessary prolonging of the "rake-in" should be avoided as much as possible because of the already mentioned extra consumption of channel capacity required for "rake-in" mode. For this research, the value of "RELEASE" is left as a factor that is varied from 20 to 40 in increments of 10, as explained in Section 5.5.

For this research effort, there will be two circumstances under which errors could occur in packets. Section 4.4.1 addresses the case where bit errors occur in the payload data fields, triggering the rake-in mode setting. In Section 4.4.2, we describe the scenario where the control fields are corrupted and what effect this has in our system.

4.4.1 Bit Errors Occurring in the Payload Data Fields

In our system, we have assumed that up to 3 bit errors can be corrected using the BCH block code in a 511-bit frame. Any number of bit errors exceeding three in a frame results in the MU or the BS declaring that the packet has been sent with an error across the wireless channel. Even then, however, depending on the nature of the source data type, retransmission does not have to be mandatory. For instance, CBR data, such as voice, over the wireless channel can tolerate, and sometimes requires that some uncorrected errors be ignored since timely arrival of the not-so-perfect data is better than having totally correct data with a prolonged delay caused by retransmission. However, there are some types of data (e.g. ABR) that are not delay sensitive but require absolute correctness. In those cases, retransmission is a must for any transferred data with errors that could not be covered by the forward error correction (FEC) scheme. For the cases where real-time traffic such as CBR and rt-VBR are involved, the retransmissions are not carried out but the "rake-in" mode is set to one of the rake-in levels. The "rake-in" mode reverts to the "normal" mode after RELEASE period elapses in the "rake-in" mode in order to prevent excessive use of the "rake-in" mode. The "rake-in" mechanism is mostly targeted to the errors occurring in the payload data packets.

4.4.2 Bit Errors Occurring in the Control Fields

The control phase of an uplink timeframe consists of the Access Request phase and the Piggybacked Access Request phase.

Whenever an Access Request is met with a NAK or no response, the MU automatically sends out another updated Access Request. The main reason for errors occurring during the Access Request phase can be attributed to collisions occurring due to multiple MUs initiating requests using the same PN code. Another possibility is the situation where there are more mobile units that need to send AR packets than the maximum channel capacity allows. Since the AR packets are assumed to be protected by a much more secure code rate, they have much less of a chance of being in error, barring any PN code collisions. This is also because, as stated in Section 4.3.2, Access Requests for connections having more than one packet queued in the mobile buffer will not have to use the AR packet, but can instead use a contention-free Piggy-backed Access Request field during data transmissions. Thus, the multiple access interference during the Access Request period is minimized and consequently the bit error rate as well.

The Piggy-backed Access Request field is a part of the data packets allocated through BS supervision and thus are contention-free. Also, the Piggy-backed Access Request field is replicated for as many times as the number of multiple substreams in use during the data packet transfer. Thus, the "rake-in" mode is always enabled for the Piggybacked Access Request field, resulting in a very secure transfer. Therefore, in this research effort, the errors in the control fields are assumed to occur only due to the packet collisions among the AR packets.

4.5 The Effect of the Rake-in Modes Analyzed and Simulated

The advantage of using a CDMA system for air-interface is that the potential multipath signals are resolvable because the channel bandwidth is much less than the chip signal bandwidth [Rap96]. The conventional RAKE receiver takes advantage of this and searches for several resolved multipaths and uses one of the appropriate summation schemes (selection diversity, maximal ratio combining, or equal gain combining) to combine the statistics of each of the branches to achieve a reduction in bit error. However, the multipath signals are not always guaranteed and thus the performance of the rake-in receiver is not always reliable. Even when

there are multipaths that are resolvable, when nonidentical resolvable paths are combined by the RAKE receiver, the effect of nonidentical fading on system performance can be detrimental. Another problem that a regular RAKE receiver faces is the synchronization errors it can have due to time varying nature of multipaths. Noting this shortcoming, this research has focused on generating an artificial mulipath that is guaranteed, more or less, since the same signal is duplicated from the multi-substream transmitter with identical time delay for each signal. The resulting side-effect is that the source data rate is immediately cut in proportion to the rate of duplication, but the bit error rate encountered can be reduced to factors that are much more than the number of duplication. Significant reduction in the bit error rate is achieved because "rake-in" receivers work in ways that actually deals with the soft decision making, which adds the analog values of each streams instead of the binary values. For instance, consider the case where BPSK modulation is used. In this case, each correlator in the receiver usually has outputs that are either near +1 or –1. Thus, assuming symmetrical construction of the correlator, decision-making point would be at 0. In other words, any positive output from the correlator would be interpreted as a "1" having been sent from the transmitter, whereas any negative output from the correlator would be interpreted as a "0" having been sent. Note that the binary value of 0 corresponds to –1 in the physical output of the correlator. The graphical description for the Additive White Gaussian Noise (AWGN) channel PDF and the decision making point is shown in Figure 4.8 and Figure 4.9, respectively.

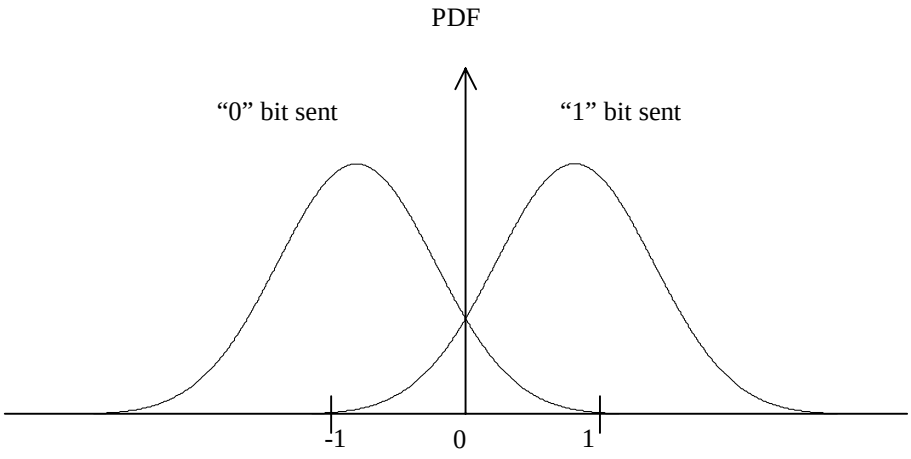
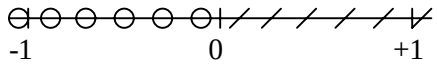


Figure 4.8: PDF for Bits Sent under AWGN Channel



○ : decision making of "0" bit having been sent

／ : decision making of "1" bit having been sent

Figure 4.9: Decision Making Point for a Regular CDMA Correlator

Now, suppose that a binary value of 1 is sent, it is conceivable that an error could result in a decision statistic value of -.1. Suppose that a second, independent bit of the same binary value of 1 is sent, and that the signal is correlated with an output of 1.0. Adding -.1 and 1 would give us .9 – which we would interpret as a "1" being received. The summed value is positive; the decision-making is correct. Instead of this simplistic example, an accurate statistical analysis is given in the following section.

4.5.1 The Analysis of Rake-in Mode

It is noted that the acceptable level of the bit error rate is 10^{-3} [GJP91] and the rake-in mode would be triggered only when the wireless channel has deteriorated further. For purposes of this analysis, we have looked into a bit error rate range of 10^{-2} to 2×10^{-1}, which would certainly warrant a change to the rake-in mode.

For this research, the Gaussian probability density function is used as the general curve representing the probability of error occurring for a signal path of the rake-in receiver. The Gaussian probability density function (PDF) is as follows,

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-m)^2}{2\sigma^2}} \tag{4.2}$$

where m is the median and σ^2 is the variance of the signal. The cumulative distribution function (CDF) of the signal is thus given by

$$F(a) = \frac{1}{\sqrt{2\pi\sigma}} \int_{-\infty}^{a} e^{-\frac{(x-m)^2}{2\sigma^2}} dx = Q(\frac{m-a}{\sigma}) \tag{4.3}$$

where the Q function is defined by

$$Q(z) = \frac{1}{\sqrt{2\pi}} \int_z^{\infty} e^{-\lambda^2/2} d\lambda \tag{4.4}$$

If we assume that a "1" is being sent from the transmitter, the value "m" should be set to 1. Also, setting "a" to zero in Equation 4.3 gives the probability that the signal which was assumed to have been sent with a "1" will be received in error. This is shown in the following equation.

$$BER = F(a)|_{a=0} = F(0) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{0} e^{-\frac{(x-m)^2}{2\sigma^2}} dx = Q(\frac{m}{\sigma}) = Q(1/\sigma) \tag{4.5}$$

Since the Q function is not a rational function, we are forced to approximate Q. For BER between 10^{-2} and 2×10^{-1}, we can set the range of the value σ as follows.

$$10^{-2} \le Q(1/\sigma) \le 2 \times 10^{-1} \tag{4.6}$$

Using the Q function approximation equation given in [Cou93], the $1/\sigma$ value ranges from .83 to 2.3, and thus it follows that

$$.43 \le \sigma \le 1.2 \tag{4.7}$$

Table 4.1 lists the values of BER corresponding to values of $1/\sigma$ for the regular DS-CDMA correlator.

Table 4.1: BER Corresponding to $1/\sigma$ in a Regular DS-CDMA Correlator

$1/\sigma$	σ	BER
.9	1.11	.184
1.1	.909	.136
1.3	.769	.0968
1.5	.667	.0668
1.7	.558	.0446
1.9	.526	.0287
2.1	.476	.0179
2.3	.435	.0107

Now, we take the next step to see how the rake-in receiver mode could be used to effectively improve the BER. The rake-in receiver works in such a way that it looks at all of the fingers available and then decides what value was really sent. First, let us assume that we have access to two "fingers" coming into the rake-in receiver.

If we have two "fingers" to work with, the decision making will have to be done at a line where the analog addition of two signal values is considered. To help visualize the analytical process, Figure 4.10 is given. Here, x1 and x2 represent the two fingers coming into the rake-in receiver. Each of the signals x1 and x2 is assumed to have identical PDF characteristics as depicted in Figure 4.8.

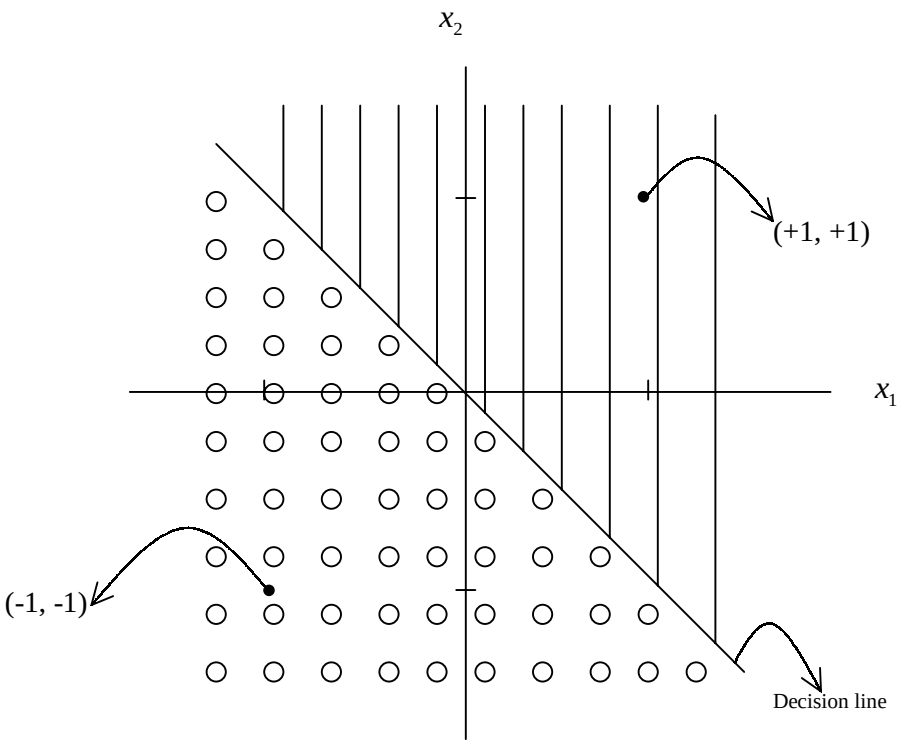


Figure 4.10: Decision Area for Two Finger Rake-in Mode

Since we have assumed symmetrical properties for +1 and –1 (0 in binary representation) values being transferred, it suffices to do analysis of the BER when +1 is being sent and generalize the result to hold true for the BER when –1 is being transferred. To obtain the BER for a signal +1 being sent in two fingers, we need to double integrate the area represented by "o"

in Figure 4.10 with the PDF values of both signal x1 and x2 multiplied. We thus obtain Equation 4.8, BER_2, which represents the BER when two fingers are available.

$$BER_2 = \int_{-\infty}^{\infty}\int_{-\infty}^{-x_2} pdf(x_1)\cdot pdf(x_2)dx_1dx_2$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{-x_2} \left(\frac{1}{\sqrt{2\pi}\sigma}\cdot e^{-\frac{(x_1-m)^2}{2\sigma^2}}\right)\cdot\left(\frac{1}{\sqrt{2\pi}\sigma}\cdot e^{-\frac{(x_2-m)^2}{2\sigma^2}}\right)dx_1dx_2$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{-x_2} \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^2\cdot e^{-\frac{(x_1-1)^2}{2\sigma^2}}\cdot e^{-\frac{(x_2-1)^2}{2\sigma^2}}dx_1dx_2 \qquad (4.8)$$

However, Equation 4.8 does not have an analytical solution, so we turn to a numerical analysis method to calculate BERs.

Approaching in a similar way, the BER can be calculated for cases of three or more fingers. Equation 4.9 and Equation 4.10 represent the BER equations for cases of three and four fingers, respectively.

$$BER_3 = \int_{-\infty}^{\infty}\int_{-\infty}^{-\infty}\int_{-\infty}^{-x_2-x_3} pdf(x_1)\cdot pdf(x_2)\cdot pdf(x_3)dx_1dx_2dx_3$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{-\infty}\int_{-\infty}^{-x_2-x_3} \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^3\cdot e^{-\frac{(x_1-1)^2}{2\sigma^2}}\cdot e^{-\frac{(x_2-1)^2}{2\sigma^2}}\cdot e^{-\frac{(x_3-1)^2}{2\sigma^2}}dx_1dx_2dx_3 \qquad (4.9)$$

$$BER_4 = \int_{-\infty}^{\infty}\int_{-\infty}^{-\infty}\int_{-\infty}^{-\infty}\int_{-\infty}^{-x_2-x_3-x_4} pdf(x_1)\cdot pdf(x_2)\cdot pdf(x_3)\cdot pdf(x_4)dx_1dx_2dx_3dx_4$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{-\infty}\int_{-\infty}^{-\infty}\int_{-\infty}^{-x_2-x_3-x_4} \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^4\cdot e^{-\frac{(x_1-1)^2}{2\sigma^2}}\cdot e^{-\frac{(x_2-1)^2}{2\sigma^2}}\cdot e^{-\frac{(x_3-1)^2}{2\sigma^2}}\cdot e^{-\frac{(x_4-1)^2}{2\sigma^2}}dx_1dx_2dx_3dx_4$$

(4.10)

Calculation of BER for five and beyond is not feasible within the practical time using numerical analysis since five or higher dimension integration is involved. (Even for a four-dimension integration the expended time surpasses 400 hours using SUN Sparc workstation machine.) Table 4.2 gives the BER results for various rake-in levels using a numerical analysis method on Equations 4.5, 4.8, 4.9, and 4.10. The table shows that the rake-in mode can cause significant improvement in BER.

Table 4.2: BERs of Various Rake-in Levels Using Numerical Analysis Method

$1/\sigma$	σ	BER	BER_2	BER_3	BER_4
.9	1.11	0.184063	0.101543	0.059509	0.035921
1.1	.909	0.135676	0.059901	0.028375	0.013904
1.3	.769	0.096813	0.033000	0.012173	0.004662
1.5	.667	0.066822	0.016951	0.004688	0.001350
1.7	.588	0.044581	0.008107	0.001618	0.000337
1.9	.526	0.028732	0.003607	0.000500	0.000072
2.1	.476	0.017878	0.001491	0.000138	0.000013
2.3	.435	0.010736	0.000572	0.000034	0.000002

BER – Bit error rate in normal mode

BER_2 – Bit error rate using double rake-in mode

BER_3 – Bit error rate using triple rake-in mode

BER_4 – Bit error rate using quadruple rake-in mode

In order to validate the results of the analysis, we have conducted two alternative methods of calculating the effectiveness of the rake-in mode.

Table 4.3: BERs of Various Rake-in Levels Using OPNET Simulator

$1/\sigma$	σ	BER	BER_2	BER_3	BER_4
.9	1.11	0.182992	0.101591	0.059641	0.03578
1.1	.909	0.135741	0.060471	0.02829	0.01427
1.3	.769	0.097131	0.03323	0.01232	0.00505
1.5	.667	0.066771	0.01716	0.00492	0.0014
1.7	.588	0.04483	0.00807	0.00177	0.00038
1.9	.526	0.02869	0.00326	0.00056	0.00013
2.1	.476	0.01834	0.00139	0.00012	3E-05
2.3	.435	0.01073	0.00056	1E-05	1E-05

BER – Bit error rate in normal mode

BER_2 – Bit error rate using double rake-in mode

BER_3 – Bit error rate using triple rake-in mode

BER_4 – Bit error rate using quadruple rake-in mode

The first alternative method is the bit level simulation using the OPNET simulator. The simulation results are shown in Table 4.3. The results closely match Table 4.2, validating the analysis of various rake-in levels represented by Equations 4.5, 4.8, 4.9, and 4.10. Unlike the analysis method, extension of the rake-in mode to more than four fingers can be done with minimal added complexity using the OPNET simulator.

Table 4.4: BERs of Various Rake-in Levels Using BER = Q(sqrt(SNR))

$1/\sigma$	σ	BER	BER_2	BER_3	BER_4
.9	1.11	0.18416	0.10161	0.059554	0.035954
1.1	.909	0.13574	0.05994	0.028393	0.013914
1.3	.769	0.09686	0.03302	0.012181	0.004665
1.5	.667	0.06685	0.01696	0.004691	0.001351
1.7	.588	0.04459	0.00811	0.001619	0.000337
1.9	.526	0.02874	0.00361	0.0005	7.24E-05
2.1	.476	0.01788	0.00149	0.000138	1.34E-05
2.3	.435	0.01073	0.00057	3.4E-05	2.12E-06

BER – Bit error rate in normal mode

BER_2 – Bit error rate using double rake-in mode

BER_3 – Bit error rate using triple rake-in mode

BER_4 – Bit error rate using quadruple rake-in mode

The second alternative validation method is inspired by the DS-CDMA system BER equation (Equation 4.11), based on the signal-to-noise-interference ratio (SNIR) estimation assuming an AWGN environment [Pur77, PSM82, MoL89, GeG91, and Rap96].

$$BER = Q\left(\sqrt{1/\left(\frac{N_0}{2E_b} + \frac{K-1}{3N}\right)}\right) \tag{4.11}$$

Here, we focus on the fact that each of the substreams used in the rake-in mechanism adds to the effective SNIR. The effective SNIR is increased in proportion to the number of substreams used in the rake-in mode. The term inside the square root in Equation 4.11 represents the effective energy per bit against noise and co-channel interference. Since each of the substreams used in

the rake-in mode is identical in strength, the effect of having "n" multiple substreams can be represented as shown below in Equation 4.12.

$$BER_n = Q\left(\sqrt{n/\left(\frac{N_0}{2E_b} + \frac{K-1}{3N}\right)}\right) \tag{4.12}$$

where "n" represents the rake-in level, or the number of substreams used for the rake-in mode. Thus, we have our second validation results, as shown in Table 4.4. The second alternative method is also easily scalable to rake-in levels beyond 4.

The values obtained from the two validation methods (Tables 4.3 and 4.4) clearly show close proximity to the values resulting from the numerical analyses of various rake-in levels represented by Equations 4.5, 4.8, 4.9, and 4.10. The significance of having the second validation method approved is that we do not need to rely on a computationally complex numerical analysis method for obtaining the improvement factors the various rake-in levels provide.

4.6 Resource Sharing with Multimedia Traffic

As introduced in Chapter 1, multimedia is the de facto application for the future wireless networks. It is necessary to implement a resource sharing mechanism for the various types of media expected. There are mainly three different types of media – Constant Bit Rate (CBR), Variable Bit Rate (VBR), and the Available Bit Rate (ABR). It is crucial to understand the basic building blocks of each type of medium that is used in a wireless ATM model. The CBR service characteristics will be looked into in Section 4.6.1, the ABR in Section 4.6.2, the VBR in Section 4.6.3, and the integrated media model with a resource sharing algorithm in 4.6.4.

4.6.1 CBR Service

Although an attempt to model a CBR service to allocate a slot periodically for the entire connection does exist [GVG89], the current trend is that there are only applications where the CBR lasts for a burst at a time for either voice or video [VRF99, BLP99]. Nonetheless, since the bit rate is constant, for example over the talkspurt, voice applications and some video

applications are categorized as CBR applications. For this research, a voice application will be used as a representative application for CBR.

Voice activity is usually characterized by its ON-OFF behavior with a voice encoder having the voice activity detector generating the ON-OFF pattern [VRF99]. The ON periods represent the talkspurts in a span of a voice connection. The OFF periods represent the silent phases. The key QoS parameters to be observed are the maximum cell delay and cell delay variation. Thus, a mechanism allowing the mobile to quickly alert the base station of the beginning of a talkspurt is necessary. The end of a talkspurt can be handled much more easily requiring only that a bit or two of a piggybacked message from the last packet notify the base station. Since the CBR service is usually associated with real-time traffic, a voice packet that exceeds a maximum transfer delay (MTD) will be dropped to avoid any unnecessary crowding of the air-interface. Using a leaky bucket algorithm between the air-interface and the wired network rectifies the cell delay variation (CDV). For this research, only the MTD is enforced and the CDV is assumed to be resolvable using a form of leaky bucket algorithm.

The talkspurt and silence durations are usually assumed to be exponentially distributed with specified mean times for each, where the silence to talkspurt ratio being six to four and with many papers referring to the actual times as 1.35 and 1 second, respectively [SrW99, NGT91, CaM95].

4.6.2 ABR Service

The data traffic application is the most representative one of an ABR application in literature [KiW96]. The main characteristics of data sources are that they arrive at a buffer in a burst or groups of packets. Thus, it is likely that these burst of cells will arrive at the WATM MAC data buffer approximately at the same time since, in general, the packetization rate will be much faster than the channel rate. For this research effort, the ABR traffic is modeled with exponentially distributed series of bursts, and the number packets generated per each burst is geometrically distributed with parameter α [AcK97, KiW96]. The probability mass function representing the number of packets in a burst is shown in Equation 4.13 below.

$$f(n) = \alpha \cdot (1-\alpha)^{n-1},\ n \geq 1, \tag{4.13}$$

where “n” is the number of packets in a burst that is geometrically distributed, the mean value is $\mu = 1/\alpha$, and the variance is $\sigma^2 = \frac{1-\alpha}{\alpha^2}$.

4.6.3 VBR Service

Multiple on-off sources are successfully used to characterize voice and VBR coded video sources [LeM93]. The advantage of using this method is that a relatively small number of parameters is needed to predict the queuing behavior of a VBR traffic. The number of On-Off sources modeling the VBR traffic has the maximum size of ten with each on-off source having a data rate of 32 kbps during the “on” state. The active and inactive durations are assumed to be exponentially distributed with specified mean times for each. VBR service is further categorized into “real-time” or “non-real-time” according to the needs of an application. Interactive services, such as video conferencing would be considered a “real-time” service, while a one-way streaming video would be considered a “non-real-time” application with a more lenient time constraint. Actual parameters (e.g., mean values, time constraint) to the distribution are discussed further in Chapter 5, where the actual simulation method is addressed.

4.6.4 Integrated Multimedia Service

Sections 4.6.1, 4.6.2, and 4.6.3, dealt with the typical characterization of each type of various sources of media. This research effort deals with the mix of traffic types by allocating four subqueues inside a queue of an MU. Each subqueue is responsible for taking in one of the four traffic sources (CBR, rt-VBR, nrt-VBR, and ABR). The advantage of using multiple subqueues is that it enables the BS to prioritize the transmit permission based on real-time or non-real-time needs more easily. Also and, perhaps, more importantly, once an MU gets the transmit permission from the BS, it is much easier for the MU to prioritize exactly what type of service should be given the permission to be sent out.

4.7 Retransmission Scheme for Non-Real-Time Traffic

As part of the APMS protocol, a retransmission scheme is facilitated at the MAC layer for the non-real-time traffic. The conventional DQRUMA protocol described in [LKZ96] does not explicitly show any results relating to retransmissions of data packets, because their simulations assume perfect wireless link as long as the number of simultaneous streams are within their set bound. A related work in [Liu96], however, does mention that retransmission of packets should be done when an error does occur. This research incorporates bursty channel conditions that are hostile for periods of time and thus permits retransmissions for non-real-time traffic. For fair comparison, retransmissions are also allowed for the non-real-time traffic in the conventional DQRUMA protocol. It is noted here that retransmissions for real-time traffic are not allowed due to stricter time constraints.

For non-real-time packets that are received in error, the M field of the TPPTA packet will be returned with certain bits that are zero. The packets corresponding to the bits that are zero and are non-real-time are checked to see if the maximum number of retransmissions has already been expended. If so, the packet will be lost in order to prevent excessive burdening of the network resources. If not, the packets that were in error are inserted back into the waiting queue for retransmission.

4.8 Summary

Section 4.1 described the link environment expected in a wireless ATM network setting and the features of APMS-CDMA wireless ATM MAC protocol that enables the robustness against bit errors during burst bit error periods. Section 4.2 presented the hardware structure of the APMS-CDMA MAC protocol based wireless ATM network. In Section 4.3, the bandwidth on demand access protocol used in APMS-CDMA MAC is described. The criterion used in deciding the various rake-in modes to be implemented was addressed in Section 4.4. The effect of the rake-in mode was analyzed and simulated in Section 4.5 to verify the effects of the rake-in mode to be used in APMS-CDMA MAC protocol. It was discovered that, for typical BERs for wireless links, their values reduced to nearly one tenth their original values for each additional stream of data available for “rake-in” mode. Also, it was noted that the analysis matches the

preliminary simulation results with respect to the various rake-in modes. In Section 4.6, the various traffic models (CBR, VBR, and ABR) were surveyed for use in our simulation models. Section 4.7 explains the retransmission scheme that is employed in our APMS protocol in an effort to satisfy the non-real-time QoS performance.

Chapter 5: Simulation Assumptions and Algorithms

The simulation of the APMS and DQRUMA protocols assumes a single cell WATM network, where there is a central base station (BS) and up to 90 MUs that will be allowed to move at walking speeds around the cell. It will be assumed that there are 25 transmitter-receiver pairs with dedicated PN codes to handle access request (AR) packets, which is consistent with the value provided in [LKZ96].

Section 5.1 describes how the PN code length and the number of simultaneous active users (channels) are related. Section 5.2 summarizes the assumptions made in the simulation. Section 5.3 sets forth the enhanced algorithm used to maximize the throughput, while letting only the packets that observe the predefined time constraint to be allowed to be transmitted by uplink. Section 5.4 addresses the method proposed to verify the contribution made by this research effort.

5.1 Channel Capacity and the Processing Gain (PN Code Length)

For a DS-CDMA application, direct calculation of the capacity has proven beyond the capability of all learned scholars who have attempted it [Web98]. However, there is a consensus among CDMA researchers that the capacity of the channel is not particularly limited by bandwidth, but rather more by multiple access interference, in normal cases, from other users inside the current cell and also from users associated with cells that are near the current cell.

There are articles dealing with reducing multiple access interference using multiuser detectors, but the subject is beyond the scope of this research. Interested readers are referred to [Mos96] and the references within.

To set up the resource sharing criterion, we first explore the commonly referred to Gaussian Approximation equation (Eqn 5.1) that is used to estimate the performance of the Direct Sequence Spread Spectrum CDMA with the assumption of perfect power control and synchronization of signals from different users ([Pur77, PSM82, MoL89, GeG91, and Rap96]).

$$P_e = Q\left(1/\sqrt{\left(\frac{N_0}{2E_b} + \frac{K-1}{3N}\right)}\right) \qquad 5.1$$

P_e: Bit error probability E_b / N_0: SNR for desired user

N: Processing Gain K: Number of simultaneous streams

We can see from Eqn 5.1 that if E_b / N_0 is assumed to be around 15dB, as specified in [WiM99], then the decisive factor for the bit error rate (P_e) is (K-1)/N. This should be clear, since as the number of simultaneous streams in a CDMA system increases, the denominator value in Equation 5.1 increases, and thus the resulting value for the Q function decreases. It is noted that the PN sequence generation can take on many different algorithms, but as the processing gain becomes large, nearly any randomly generated sequence will have approximately equivalent properties [Mil95]. Equation 5.1 is based on this assumption and this approach is followed for this research effort.

The target application areas for WATM covers the diverse media of data network, voice, video on demand, and real-time multimedia. In general, a packet loss rate of 10^{-3} is often seen as an upper limit for subjectively acceptable reproduction of multimedia data, such as digital voice, although exact values can vary with the specific voice compression and receiver interpolation methods used ([WGJ93, GJP91]). If we assume, as stated above, that an acceptable bit error probability (P_e) is around 10^{-3} for a wireless system, processing gain N is 128, and that the target E_b / N_0 ratio is 15dB [WiM99], we have the following equation:

$$P_e = Q\left(1/\sqrt{\left(\frac{N_0}{2E_b} + \frac{K-1}{3N}\right)}\right) \leq 0.001, \qquad 5.2$$

where N_o is the noise spectral density, E_b is the energy per bit, and N is the spreading gain.

Using the Q-function table we have,

$$1/\sqrt{\left(\frac{N_0}{2E_b}+\frac{K-1}{3N}\right)} \geq 3.09\text{, or} \qquad 5.3$$

$$K = 3N \times (0.1047 - \frac{N_0}{2E_b}) + 1 = 35.13 \qquad 5.4$$

The parameters and assumptions related to the channel capacity allowed in this research are summarized below:

- N (Processing Gain): 128 chips per source bit
- E_b / N_0 = 15 dB
- Maximum Number of Simultaneous Streams (channels): 35
- Perfect equivalent power control and synchronization assumed

5.2 Forward Error Correction and Error Detection

The BCH code was chosen for the powerful feature of implementing the forward error correction (FEC), as follows ([Pro95]).

For a total packet length of $n = 2^j - 1$, where $j \geq 3$, the number of correctable bits is $t \geq \frac{2^j - 1 - k}{j}$, where t is the error correcting capability and k is the actual number of input bits.

For example, if we let j = 9 to set the total length of the data packet to 511 bits and let k = 484, then we have $t \geq \frac{511-484}{9} = 3$. In this case, we are capable of correcting up to 3 bit errors. Now, if we assume a 10^{-3} bit error rate, we can calculate the approximate packet error rate as follows:

$$Pk\_Error = {}_{511}C_4 \times (.999)^{511-4} \times (.001)^4 = 1.69 \times 10^{-3} \qquad 5.2$$

This packet error rate would be marginally acceptable [WGJ93]. However, the bit error rate fluctuates depending on the channel condition and will cause unacceptable packet error rates at

times. The APMS MAC protocol, devised for this research effort, explained in Chapter 4, attempts to alleviate this hazard brought on by the often faulty wireless link.

For our simulations, we assume the data packet length to be 511 bits long, including the 27 bits of BCH forward error correction code capable of correcting up to 3 error bits. Error detection is done by employing a 12-bit CRC code, which is part of the 484-bit packet body. It is capable of detecting all single burst-errors up to length 12, 99.95% of all error bursts of length 13, and 99.976% of all error burst of length greater than 13 [Wic95].

5.3 Wireless Channel Model

The wireless channel has been studied from the time radio was first invented, and needless to say, the scope of the coverage extends through the entire gamut of abstraction. For the channels that we are interested in, they are slowly changing compared to the length of a symbol period. Thus, the small-scale fading issues are assumed to be taken care of, to the best of the components' ability, using such techniques as equalization and interleaving. Our concern is more on the burst errors that are characteristic in the MAC layer. The conservative static error assumption, often for purposes of theoretical analysis, do not accurately account for the burst errors encountered in wireless channels [WiM99].

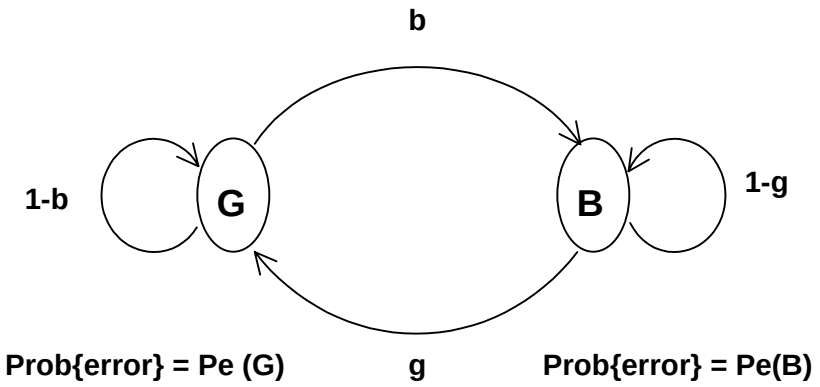


Figure 5.1: Gilbert-Elliott Burst Error Model

One of the classical models that does incorporate memory into the channel is the Gilbert-Elliott model [Ell63], as illustrated in Figure 5.1. The Gilbert-Elliott model is used when

representing systems that undergo periods of burst error conditions [Ell63, WiM99]. The model is composed of two Markovian states, the "good" (G) and the "bad" (B). The "good" state represents the normal state, while the "bad" state represents the error-prone state. In the G state, the channel conditions are normal, meaning the bit error rate encountered during that state where the BER is determined by the number of used channel and spreading gain, as given in Equation 5.1. However, during the B state, the channel experiences a drastic decline in reliability and produces burst errors (Pe(B)). Some typical relationships between them have been given in [WiM99], and the values are repeated in Table 5.1.

Table 5.1: Pe(G) and Pe(B) Pairs Originally from [WiM99]

Pe(G)	Pe(B)
8.11e-3	0.266
6.93e-3	0.243
5.70e-3	0.219
4.49e-3	0.195
3.35e-3	0.171
2.33e-3	0.147
1.49e-3	0.125
8.69e-4	0.105
4.37e-4	0.087

The probability of transition from the G state to the B state (Prob(G->B)) is denoted by "b," while the probability of transition from the B state to the G state (Prob(B->G)) is denoted by "g." It follows that "1-b" represents the probability that the channel will stay in the G state, and "1-g" represents the probability that the channel will stay in the B state. It is noted here that the Gilbert-Elliott model [Ell63] differs from its parent model, the Gilbert model [Gil60], in that the probability of bit error in the G state in the Gilbert model is zero while in the Gilbert-Elliott model, it is not. However, both attempt to model the bursty error nature of the wireless link. For this research effort, we adopt the parameter values presented in [WiM99] that are within the scope of our research. Also, in order to facilitate the simulation, we have adopted a time

modulated model as specified in [FMW00], [BBK97], and [DRT97]. The time spent in the G and B states are exponentially distributed with different means according to the following equations [WiM99]:

$$\overline{T}(G) = 1/b \quad 5.3$$

$$\overline{T}(B) = 1/g \quad 5.4$$

Equations 5.3 and 5.4 depict the number of time units that are expected to pass before a transition to another state occurs. For this research, the time units are set to the duration a frame.

5.4 Simulation Algorithms

The BS maintains a central table containing 1 row of information fields for each MU in the system. Table 5.2 illustrates the components of a row in the central table. Each row corresponds to bandwidth on demand information about an MU. Without the loss of generality, the rows are assumed to be in ascending order of the mobile ID. Each row has information fields that are divided into input and output fields; the input fields are used for storing information gathered from the mobile units (MUs), while the output fields are made available to store information pertaining to the transmit permission packet sent on the downlink. The input fields include the mobile ID (MID), update information, the number of multiple substreams requested (MSS), the number of multiple substreams requested for real-time traffic (MSSRT), and rake-in level (RL). The output fields include the number of multiple substreams allowed (MSSA), the number of multiple substreams allowed for real-time traffic (MSSRTA), and the rake-in level allowed (RLA).

Table 5.2: BS Central Table

MID	Update	MSS	MSSRT	RL	MSSA	MSSRTA	RLA
0-99	(y/n)	(0-12)	(0-12)	(1-6)	(0-12)	(0-12)	(1-6)

The APMS MAC protocol utilizes the information gathered in the central table to generate transmission permissions to the MUs. The mechanism that is employed has a great effect on the proper working of the APMS MAC protocol. For instance, any data packets transferred during the "bad" period have little chance of being accepted error-free at the receiver if the packets are sent individually. On the other hand, sending data packets in replication during a "good" channel period does not serve the overall throughput of the channel well. It is, thus, necessary to conduct estimations of the channel conditions. For this research effort, we implement a simple, but effective, method of estimating the channel condition without any physical overhead. The chance that a packet is received in error is quite small (maximum packet error rate of 1.69×10^{-3}, see Section 5.2) during the "normal" state of the channel. However, the situation changes when an MU is experiencing a "bad" state. It is very likely that tens of packets will be received in error in a burst during the "bad" periods, according to articles [WiM99] and [Ell63]. This has led us to implement the channel estimation by declaring that the "bad" periods are when more than 60% of the packets received during the last two packet time frames are in error. Once the "bad" period is declared, the data being transferred are replicated several times and the decision statistics are combined at the output of the correlators for an improved chance of obtaining an error-free packet, as described in Section 4.5. The number of replications (RL) has some impact on the overall throughput of the system, and it is left as one of the factors in the simulation. The ensuing question then becomes when the "bad" state should revert back to the "good" state (RELEASE). RELEASE is also left as one of the factors in the simulation.

With the information given above, it is now possible to formulate a traffic-scheduling algorithm that enhances the overall throughput of the system, as illustrated in Figure 5.2. The traffic-scheduling algorithm is as follows. First, the total available capacity is inserted into the variable Max_K. Then, the total number of streams that are requested is compared against Max_K. It is noted here that the total number of streams that are requested includes the streams that would be replicated due to the rake-in mode. If it is found that the total requests are below Max_K, all requests are accommodated by assigning MSSA with MSS, and RLA with RL for all MUs that are currently active.

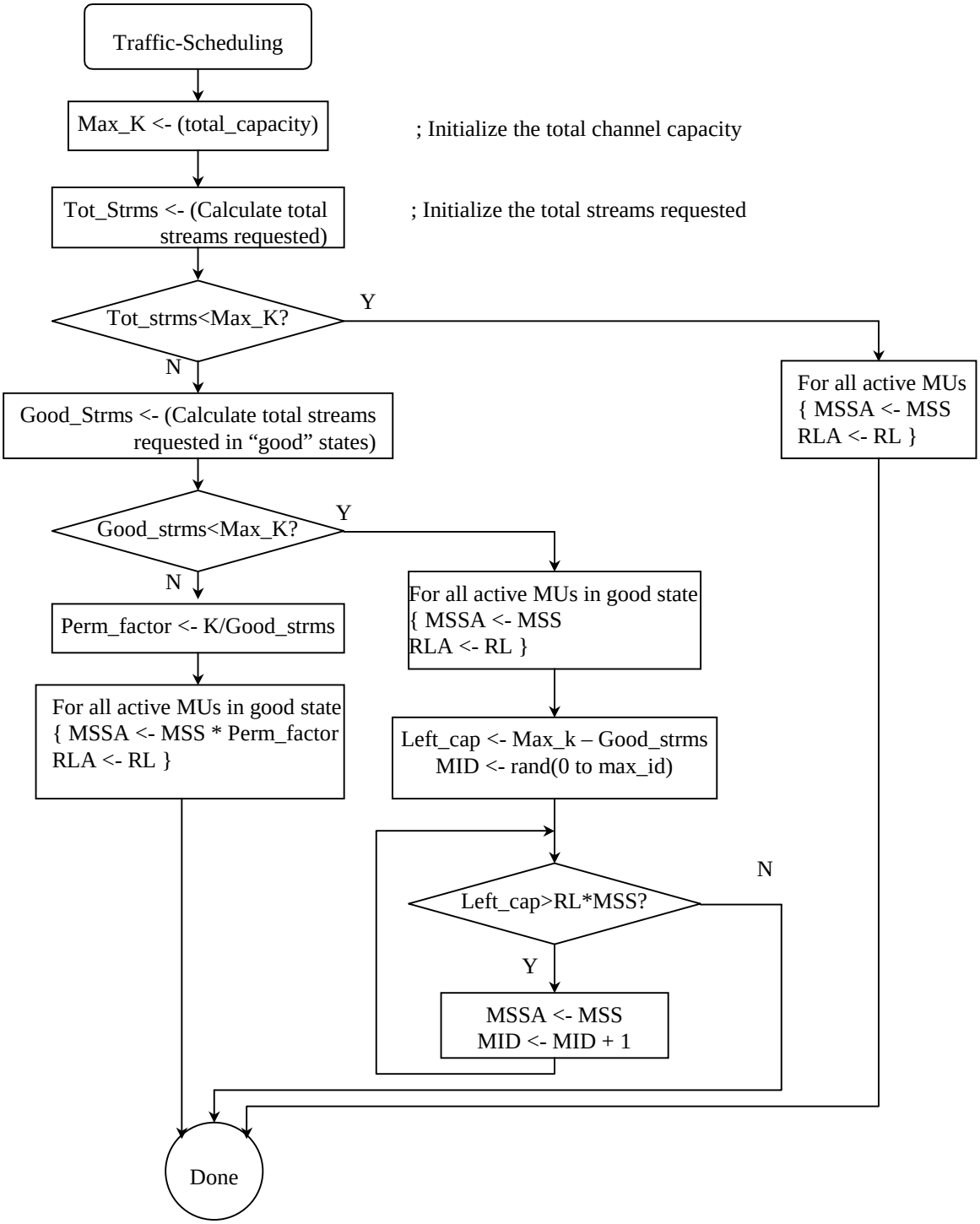


Figure 5.2: Traffic Scheduling Algorithm for Each Time Frame

However, if the total requests exceed the total allowed capacity, Max_K, then the BS needs to prioritize the allocation of permissions. This is done by, first, allowing the requests from MUs in the "good" mode to be granted permissions if the total requests from MUs in the "good" mode do not exceed the total capacity of the channel, Max_K.

If, however, the total number of requested streams that are in the "good" mode exceed Max_K, a permission factor, Perm_factor, is calculated based on a ratio between the Max_K and the total number of requested streams that are in the "good" mode. All permissions regarding the multiple requests will go through a statistical tapering by a factor of 1-Perm_factor. In other words, the total number of "good" streams admitted into the system will be limited to Max_K statistically. The "bad" streams will, in this case, not be allowed to send any of their packets. The reason for this is that the "bad" streams requires more bandwidth, by a factor of the rake-in level (RL), and during highly-loaded situations, transferring only the "good" streams will already have consumed the allowed channel capacity.

Now, we focus on the case when the total capacity resulting from the requests for the "good" streams is below Max_K. In this case, there is some channel capacity left for granting permissions to the "bad" streams. The selection process for the permissions given to the MUs in the "bad" mode begins by randomly selecting the initial MID. If the current MU is in the "bad" mode, the channel checks whether the capacity left is greater than the number of streams the MU is requesting. If so, all the requests will be granted permission for that MU. If not, only a portion of the requests or none, depending on the residual capacity, will be granted permission.

The proper estimation of the channel condition ("good" or "bad") used in the above algorithm is also of crucial importance. The determination of whether the channel is in a "good" or a "bad" period is updated every frame cycle. The channel estimation algorithm for each frame cycle is depicted in Figure 5.3. First of all, the MID is initialized to 0. The maximum number of active MUs is represented by the MU_MAX variable. Without the loss of generality, the channels corresponding to the MID is checked whether the current estimation is "good" or "bad."

If the current estimation of a channel is "good," the past 2 most recent frames are examined to see whether the packet error rate (PER) was 60% or more. If so, the current channel is estimated to be in the "bad" period. If not, the channel is still estimated to be in the "good" period.

If, however, the current estimation of the channel is “bad,” then PER for the last RELEASE number of frames is calculated. If the PER is below 0.001, the channel is assumed to be in the “good” period again. If not, the channel is still assumed to be in the “bad” period.

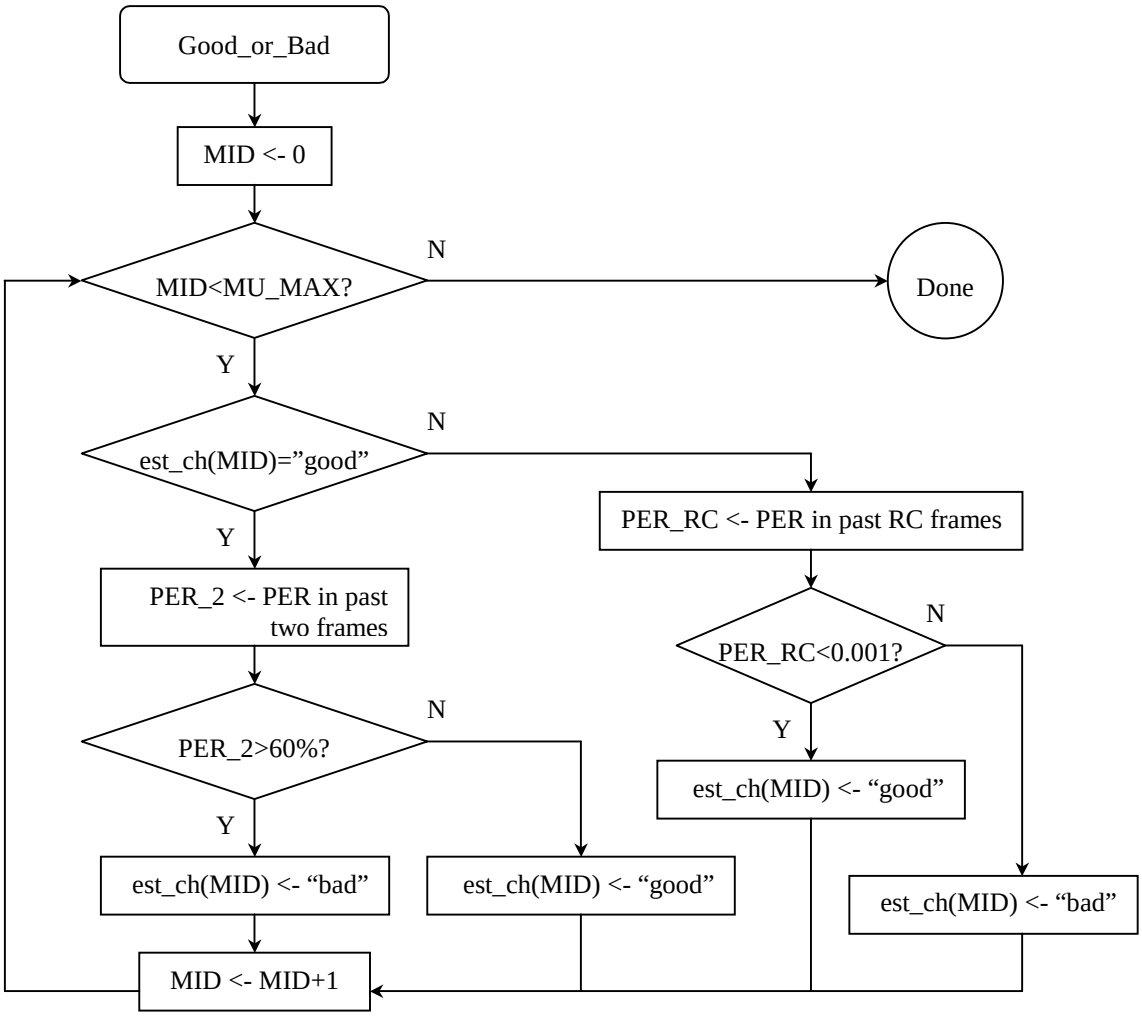


Figure 5.3: Channel Estimation Algorithm

The backoff scheme used for the AR packets follows what is employed in the conventional DQRUMA protocol [Liu96] - the harmonic backoff scheme. That is, the first AR packets send their packets unconditionally. However, once a collision is detected, the probability of generating subsequent AR packets follows the harmonic series. The algorithm is illustrated in Figure 5.4. The harmonic backoff scheme effectively prevents over saturation of the network resources.

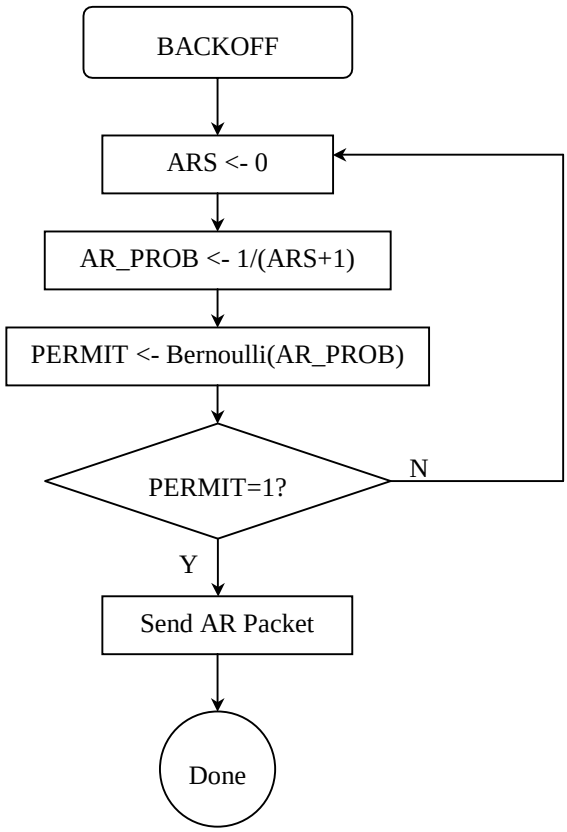


Figure 5.4: Backoff Algorithm

5.5 Simulation Factors

In this section, we define the factors that impact the overall simulation results. These factors are varied as shown below in Table 5.3.

Table 5.3: Simulation Factors and Factor Levels

Factors	Factor Levels
LOAD	0.1, 0.3, 0.5, 0.7, 0.9
LD_METHOD	Method 1, Method 2
MAX_RETRAN	0, 3, 5, 7
RL	1, 4, 6
CHAN_EST	1: Estimation used, 0: Optimum case
RELEASE	20, 30, 40

LOAD is the fraction of the maximum capacity of the channel (normalized load), as calculated in Section 5.5. We investigate five levels ranging from 0.1 to 0.9. There are, however, several ways that a load could be given, and this factor is defined as the *LD_METHOD*. One way of varying the load is to vary the load within an MU, while fixing the total number of MUs that are active, which is Method 1. Another way of varying the load is to vary the number of MUs that are active in the system, while fixing the load within an MU, which is Method 2. Method 1 permits us to investigate the effect of the rake-in mode in the APMS protocol with minimal influence from the random access scheme, which is not the main topic of this research. Method 2, however, is also examined to verify that the APMS still works under circumstances where there is a wide range in the number of MUs that are active. By including both methods, we are able to assess the practicality of the APMS under various scenarios of loading.

MAX_RETRAN corresponds to the maximum number of retransmissions that are allowed for non-real-time traffic. There are four levels, including the case where no retransmission is allowed. The case for no retransmission allowed (*MAX_RETRANS* = 0) is included because of the fact that the conventional DQRUMA MAC protocol does not experiment with retransmissions at the MAC layer. Thus, although our newly proposed APMS MAC protocol explicitly requires retransmissions for non-real-time traffic, for fair comparison against APMS MAC protocol, the case for no permitted retransmissions is included.

The *RL* factor refers to the rake-in level (RL) that is used when a channel is detected to be in the "bad" mode. There are three levels, where the level of 1 effectively simulates the conventional DQRUMA protocol, while the other two values correspond to the levels set when the rake-in mode is used for the APMS protocol.

The *CHAN_EST* factor determines how the channel state is detected. Level 1 is the case where channel state estimation is achieved by observing the packet error rates, without any special devices explicitly devoted to channel state detection. Level 0 represents the theoretically optimal case, where a perfect knowledge of the channel state is assumed. Although the main goal of this research is to show the improved throughput performance of the channel estimated APMS protocol against the conventional DQRUMA protocol, simulating and comparing against

the optimal channel assessment case is valuable for future research and is thus included in every throughput performance comparison.

The *RELEASE* factor represents the number of time frames checked for the error-free packets during one of the rake-in modes. A rake-in mode will return to the normal mode only when all the packets transferred during the past *RELEASE* time frames are error-free. Three levels of *RELEASE* are investigated.

5.6 Simulation Parameters

The parameters that are used in the simulations are described in the following sub-sections.

5.6.1 Simulation Length

The simulation length was chosen by doing pilot simulations that run for different lengths of time for the two extreme normalized load cases, 0.1 and 0.9, for both the DQRUMA and APMS protocols. Other factors, as described in Section 5.5, were fixed to the values as follows:

Load Method: Varying the number of users to vary the normalized load

Max Retrans: 5

Rake-in Level: 4 (for the APMS case)

Chan_Est: 1

Release: 30

Five sets of seeds were used for four different time periods, 15, 30, and 45 seconds. The results showed that their outputs were statistically identical using t-paired tests. We chose 30 seconds as the total simulation length for all the simulations.

5.6.2 AR Packet Length

The AR packet length is set to 31 bits. This is in accordance with the value specified in [LKZ96].

5.6.3 Piggyback Request Packet Length

The Piggyback Request packet length is set to 15 bits. The length of this packet is shorter than the AR packet length due to implicit identification of the MID through the PN codes that are used. The conventional DQRUMA protocol, described in [LKZ96], lumps the Piggyback Request packet as part of the Data packet. However, that prevents the timely decoding of the piggyback request information. For this research effort, the piggyback request is treated as a separate packet. The characteristics of the original Piggyback Request packet are kept intact in that there is no contention for requests when using the Piggyback Request packet.

5.6.4 Data Packet Length

The Data packet length is set to 511 bits. The data packet excludes the Piggyback field but includes the BCH coding information, as described in Section 5.2, to correct up to 3 bit errors. The conventional DQRUMA protocol does not specify how many bits can be corrected and only indicates that all transmitted packets are assumed to be correct or corrected by the BCH code.

5.6.5 Processing Gain

The processing gain chosen for this research effort was 128, as mentioned in Section 5.1. This relatively high value will facilitate the testing of the APMS protocol in a wideband CDMA environment, where many users are allowed to be active simultaneously [DTS00].

5.6.6 Eb/No Ratio

The post-processing Eb/No ratio was set to 15 dB, in order to stay abreast with the figures mentioned in [WiM99] concerning the Gilbert-Elliott bursty wireless channel model.

5.7 Model Verification

It is important to verify that a simulation model functions as it is intended to. It is usually called debugging in simulations [Jai91]. The OPNET Modeler provides an external debugging tool that enables the user to freeze frame at any event and at any process module. Extensive care was exerted in verifying the correct transfer of various packets between the MUs and the BS. Each of the stream interrupts and self-interrupts was trailed and the state variables examined at each breakpoint to verify proper flow of the algorithms described in Section 5.4. This thorough approach enabled the verification of the simulation model with high confidence.

5.8 Model Validation

A model validation is a step taken to see if the developed model represents the real system [Jai91]. Toward this end, extensive analysis and simulations were done for the Rake-in Mode at the bit level. At the packet level, simulation parameters were altered to match the configurations given in the conventional DQRUMA protocol and produced results that were very close to the ones given in [LKZ96] for various types of traffic.

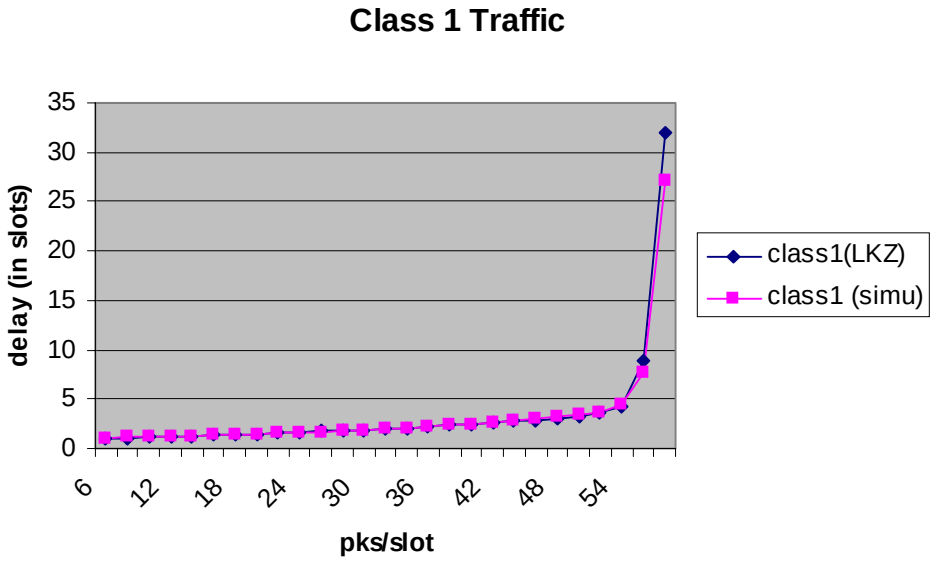


Figure 5.5: Validation for Class 1 Traffic

The results are shown in Figures 5.5, 5.6, and 5.7. For the figures mentioned, Class 1 traffic is produced by having 100 MUs with Bernoulli burst arrivals with 5 deterministic packets per burst. Class 2 traffic is produced by having 100 MUs with Bernoulli burst arrivals with 1 deterministic packet per burst. Mixed traffic refers to the case where there are 50 MUs of Class 1 traffic and 50 MUs of Class 2 traffic in the system. The factor varied was the load; more specifically the average aggregate number of packets that were produced per time slot from bursts of data from the 100 MUs. Each time slot corresponds to the transmission time for the AR and Data Packet (including the Piggyback Request field). The performance metrics were the number of time slot delays an MU experiences before being able to transfer its data packets to the BS. It is assumed in [LKZ96], that no packet errors occur as long as the maximum channel capacity is not infringed upon, and thus throughput is not tested.

For Figure 5.5, our simulation results match very closely the values given in [LKZ96]. The only point where there is any noticeable difference is at 58 packets per slot condition, which is where the system starts to break down.

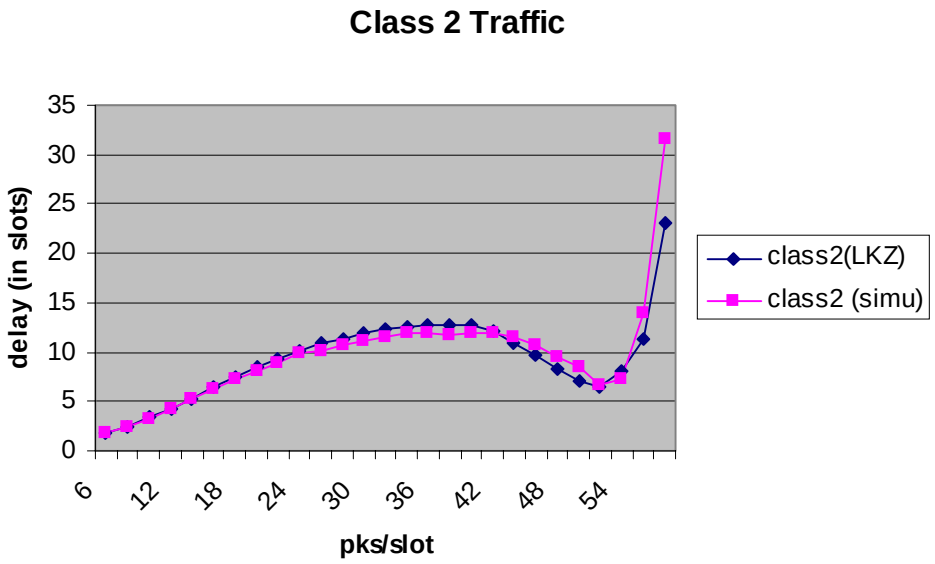


Figure 5.6: Validation for Class 2 Traffic

Figure 5.6 provides a more interesting curve because the bursts come in at a deterministic one packet per burst, and thus, at the low and modest load conditions, most access requests are

done using AR packets, which are subjected to contention. It is obvious that, at the lower levels of load, the transfer delay between an MU and the BS would be low since the level of contention would be low. However, results are intriguing for the final one third of the system load, where the transfer delay actually begins to decrease until the maximum system capacity is reached. This phenomenon occurs because, during heavier load conditions, there is a better chance that the MU will have packets waiting in the queue by the time transfer permission packet is received from the BS, thus utilizing the contention-free piggyback request mechanism.

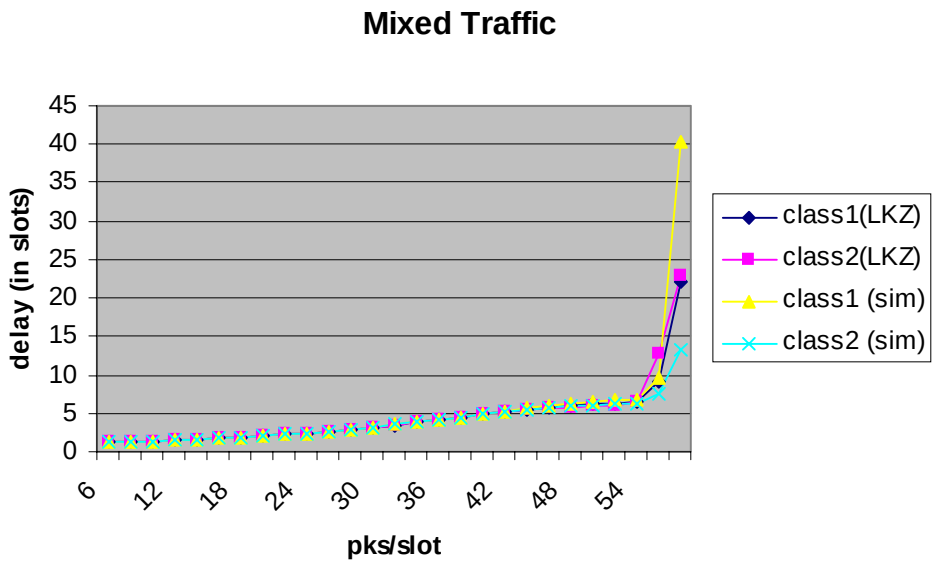


Figure 5.7: Validation for Mixed Traffic

The mixed traffic simulation results, shown in Figure 5.7, also closely resembled the values presented in [LKZ96], until the maximum capacity breaking point was reached.

Clearly, all the simulation results closely resembled the values illustrated in [LKZ96], thus validating our simulation model.

5.9 Summary

This chapter highlights the issues pertinent to the simulation system design and development. In Section 5.1, the channel capacity was derived from previous works available in

the literature related to the analysis of DS-CDMA wireless technology. Error detection and correction methods were discussed in Section 5.2. The modeling of the bursty nature of the wireless channel was described in Section 5.3. The simulation algorithms defining the key aspects of the APMS protocol were depicted in Section 5.4. The simulation factors that were varied as part of the comprehensive investigation into the performance of APMS, were described in Section 5.5. The parameters that were used in the simulations are listed in Section 5.6. The simulation model verification method was explained in Section 5.7. The simulation model validation against the conventional DQRUMA model was presented in Section 5.8. Having verified and validated the simulation model, the simulation results for various simulation factors were obtained with high confidence.

Chapter 6: Simulation Results

This chapter presents the simulation results obtained through this research effort. It compares the performance between the conventional DQRUMA protocol and the APMS protocol developed through this work. APMS performance with optimal knowledge of the channel condition is included in the figures concerning throughput performance in order to compare against the APMS protocol with channel estimation. This chapter is divided into four main sections. Section 6.1 discusses the performance of the two protocols when the total normalized load is dependent upon the variation in the load within each MU. Section 6.2 examines the case where the total number of MUs that are active in the system varies while the average load within an MU is fixed. The performance of the mixed traffic cases is divided into two sets of simulations in order to identify the performance of the response variables when the effects of the contention mode are minimized compared against the case where the contention mode has a more fluctuating effect. Both Section 6.1 and Section 6.2 assume that the number of various traffic (CBR, rt-VBR, nrt-VBR, and ABR) is evenly distributed. The performance of DQRUMA and APMS is discussed in the context of the four performance metrics (throughput, mean delay, dropped packets, and maximum retransmissions exceeded) discussed in Section 3.3. A complete investigation of the performance is done by conducting a full factorial simulation with factors presented in Section 5.5. Section 6.3 examines three special cases. The first special case is where APMS performance is compared against DQRUMA performance when retransmission is allowed for the DQRUMA protocol. The second special case conducts the study where only the non-real-time (nrt-VBR and ABR) traffic are generated. The third special

case is where the loads are generated exclusively from real-time (CBR and rt-VBR) traffic. Section 6.4 summarizes the simulation results obtained.

6.1 Multi Traffic Simulations with Load Variations within MUs (Method 1 Simulation)

Both the DQRUMA and APMS protocols go through the contention phase when MUs are sending their first packets. The most influential factor affecting the contention phase performance is the number of MUs that are active in the system. In this section, however, we fix the number of MUs that are active in the system to 5 when investigating the performance of DQRUMA and APMS. This will enable us to focus on the throughput performances that are under identical conditions concerning the contention mode. The load within an MU, however, is varied to cover the range of the LOAD factor (0.1, 0.3, 0.5, 0.7, and 0.9). This method of varying the load corresponds to Method 1 of the LOAD_METHOD factor, which is described in Section 5.5. Other factors are dealt with comprehensively by exploring the performance of the response variables for all the combinations mentioned in Section 5.5.

Throughput is almost always the most important performance metric in any type of computer network, and the convention is followed in this research effort. Other response variables, such as delay, dropped packets, and the number of packets exceeding the maximum retransmission limit, will also be investigated to gain a comprehensive viewpoint of the performance improvements and tradeoffs that are done between DQRUMA and APMS protocols.

The simulations in this section, thus, involve varying the following level of factors: 5 levels of LOAD, 4 levels of MAX_RETRANS, 3 levels of RAKE_IN_LEVEL, 2 levels of CHAN_EST, and 3 levels of RELEASE. RAKE_IN_LEVEL of 1 and MAX_RETRANS of 0 address the special case where the DQRUMA protocol is tested. In this case, the CHAN_EST and RELEASE do not apply and are “don’t care” conditions. Each valid combination is repeated five times with different simulation seeds provided by the OPNET Modeler in order to find an appropriate confidence interval. The response variables obtained for the five simulation runs are then averaged.

This comprehensive approach results in the following simulation runs for the APMS simulation model: all levels of LOAD (5), three levels of MAX_RETRANS (3), two levels of RAKE_IN_LEVEL (2), both levels of CHAN_EST (2), all levels of RELEASE (3), and 5 levels of simulation seeds. This yields 900 simulations. The simulation model can also fix the RAKE_IN LEVEL factor to 1 and the MAX_RETRANS to 0 for implementing the DQRUMA model. In this case, all the LOAD (5) and 5 simulation seed levels are used to form another 25 simulations. The total simulation runs for this section was thus 925.

Table 6.1: The Average Relative Throughput Results for Method1 Simulations

RTRN	RL	RLS	CBR	rt-VBR	nrt-VBR	ABR	Average	Rank
0	4	20	0.97893	0.97246	0.97723	0.96501	0.97341	
0	4	30	0.98218	0.97436	0.97436	0.97348	0.97610	
0	4	40	0.98402	0.97306	0.98577	0.97867	0.98038	
0	6	20	0.97882	0.96156	0.97480	0.96754	0.97068	
0	6	30	0.98440	0.96607	0.98076	0.97218	0.97585	
0	6	40	0.98638	0.96598	0.98333	0.97594	0.97791	
3	4	20	0.97783	0.97116	0.99968	0.99932	0.98700	
3	4	30	0.98222	0.97397	0.99954	0.99980	0.98888	
3	4	40	0.98359	0.97305	0.99954	0.99984	0.98900	1
3	6	20	0.97885	0.96116	0.99940	0.99858	0.98450	
3	6	30	0.98323	0.96405	0.99945	0.99951	0.98656	
3	6	40	0.98558	0.96444	0.99944	0.99931	0.98719	
5	4	20	0.97806	0.97165	0.99964	0.99977	0.98728	
5	4	30	0.98203	0.97350	0.99946	0.99966	0.98866	
5	4	40	0.98361	0.97304	0.99954	0.99963	0.98896	3
5	6	20	0.97896	0.96139	0.99942	0.99935	0.98478	
5	6	30	0.98305	0.96339	0.99936	0.99978	0.98640	
5	6	40	0.98557	0.96394	0.99957	0.99958	0.98717	
7	4	20	0.97851	0.97173	0.99961	0.99969	0.98738	
7	4	30	0.98203	0.97352	0.99946	0.99974	0.98869	
7	4	40	0.98367	0.97306	0.99955	0.99969	0.98899	2
7	6	20	0.97849	0.96047	0.99930	0.99950	0.98444	
7	6	30	0.98300	0.96366	0.99932	0.99983	0.98645	
7	6	40	0.98562	0.96389	0.99957	0.99959	0.98717	

6.1.1 Relative Throughput

Our new method resulted in dramatic improvements in the overall throughput while keeping the delays within tolerable ranges for real-time and non-real-time services: 20mS and 1 second, respectively.

Using our APMS protocol, among the levels of factors mentioned in Section 5.5, the "best" selection turned out to be a RAKE_IN_LEVEL value of 4 during "bad" channel states, a RELEASE count of 40 from a "rake-in" mode, and a MAX_RETRAN value of 3. The best performing combination was determined by averaging the relative throughputs for the plots as shown in Table 6.1. Detailed descriptions of the data, including the t-Test results, are shown in Appendix B.

With these selections, the non-real-time services, nrt-VBR and ABR, showed near perfect performance in terms of throughput despite the bursty, error-prone channel conditions. This is shown in Figures 6.1 and 6.2, respectively.

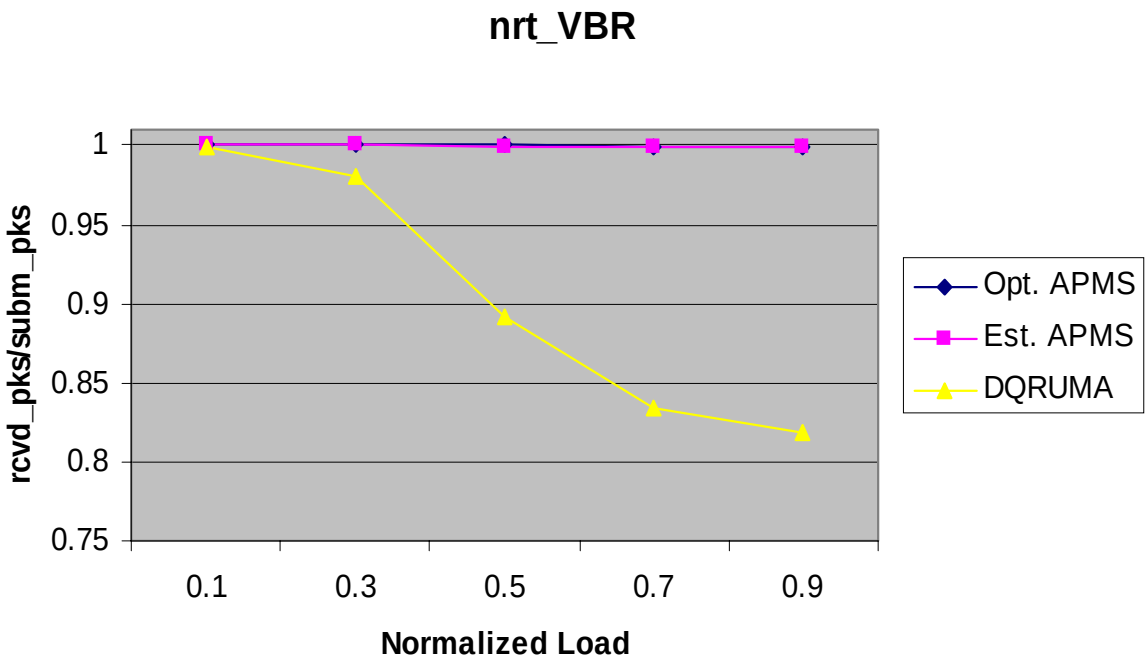


Figure 6.1: Relative Throughput for nrt-VBR Traffic

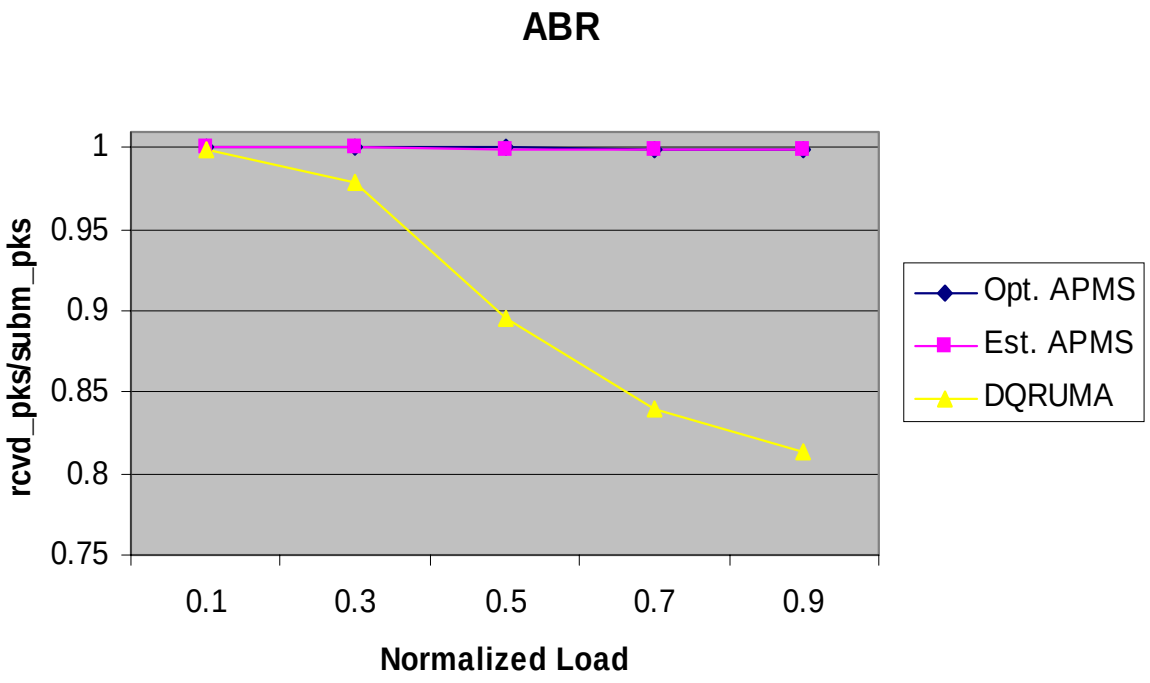


Figure 6.2: Relative Throughput for ABR Traffic

These results were achieved while keeping the total number of retransmissions allowed to be 3 or less and the maximum tolerable delay of 1 second. In all the throughput figures, "Opt." APMS plots represent the case where the BS had perfect knowledge of the channel condition, while "Est. APMS" plots represent the case where the channel conditions were estimated through counts of error packets, as described in Section 5.4. The former represented the theoretical case where APMS had direct and perfect channel state knowledge, and the latter represented what was achievable in practice without any special devices to detect channel conditions directly. For Figures 6.1 and 6.2, the "Est. APMS" plots are practically identical to the "Opt. APMS" plots.

These results were possible through two major improvements in dealing with bursty error conditions: the rake-in mode and the retransmission scheme. The rake-in mode enabled the packets in the bad mode the opportunity to replicate their data according to the RL of 4. This greatly improved the chances that the packets would be received error-free at the receiver. Also, since these were non-real-time traffic, a more lenient time constraint allowed for the opportunity to retransmit packets that were received in error. These results for the non-real-time traffic are

significant because the APMS protocol actually gives higher priority to real-time traffic in terms of allowing transmission permission.

The relative throughputs of the real-time traffic services (CBR and rt-VBR) employing the APMS protocol also showed significant improvements over the conventional DQRUMA protocol, as shown in Figures 6.3 and 6.4.

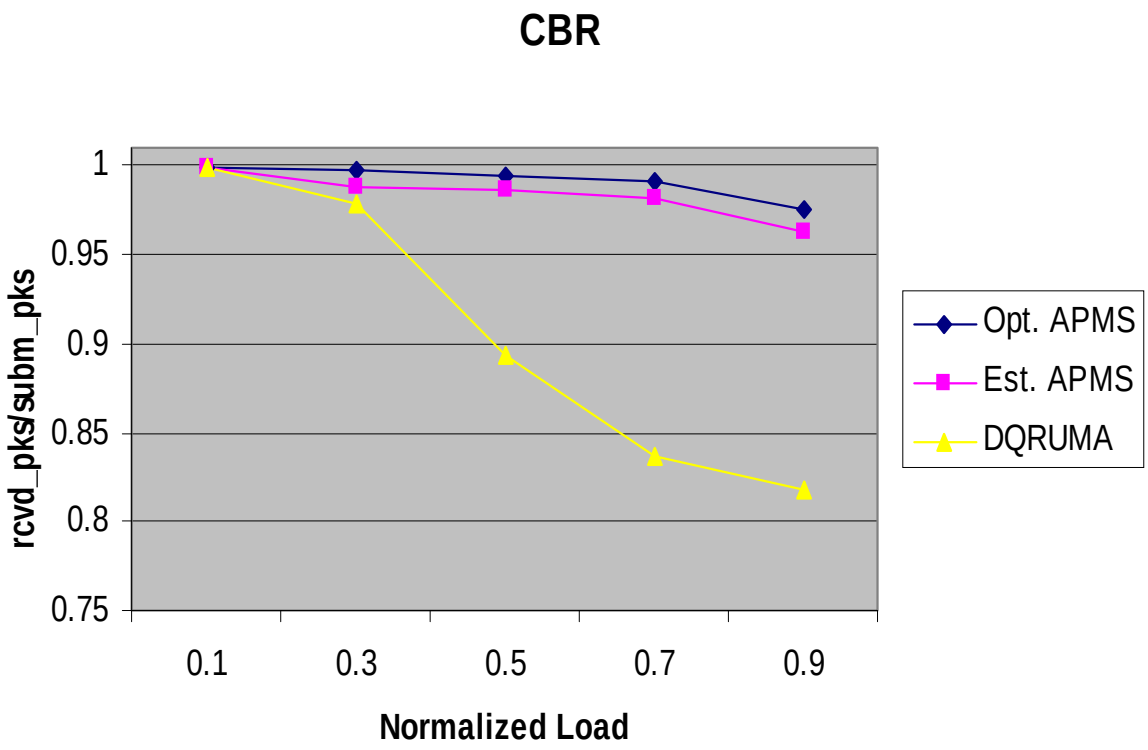


Figure 6.3: Relative Throughput for CBR Traffic

For the CBR traffic on both DQRUMA and APMS protocols, as the normalized system load exceeded the 0.1 level, the relative throughput (received packets vs. submitted packets) decreased due to the "bad" periods. For the conventional DQRUMA protocol, the relative throughput was cut by as much as 19 % of the maximum value. However, using our APMS protocol, the decline was observed to be less than one fourth the value. Alternately, 75 % of the packets that were received in error using the DQRUMA protocol were received error-free by our protocol. It is important to note that this increase in throughput is significant because most

packet errors occur during the “bad” time periods, which is approximately 19 % of the total transmission time [WiM99]. However, as the load increases, the overhead of the APMS manifests itself, leaving a gap between the “Opt.” and “Est.” plots. This is attributed to the fact that the stricter time delay constraint of 20 milliseconds and no permissible retransmission forced the system to take the loss due to the channel state estimation overhead period involved in estimating the channel condition.

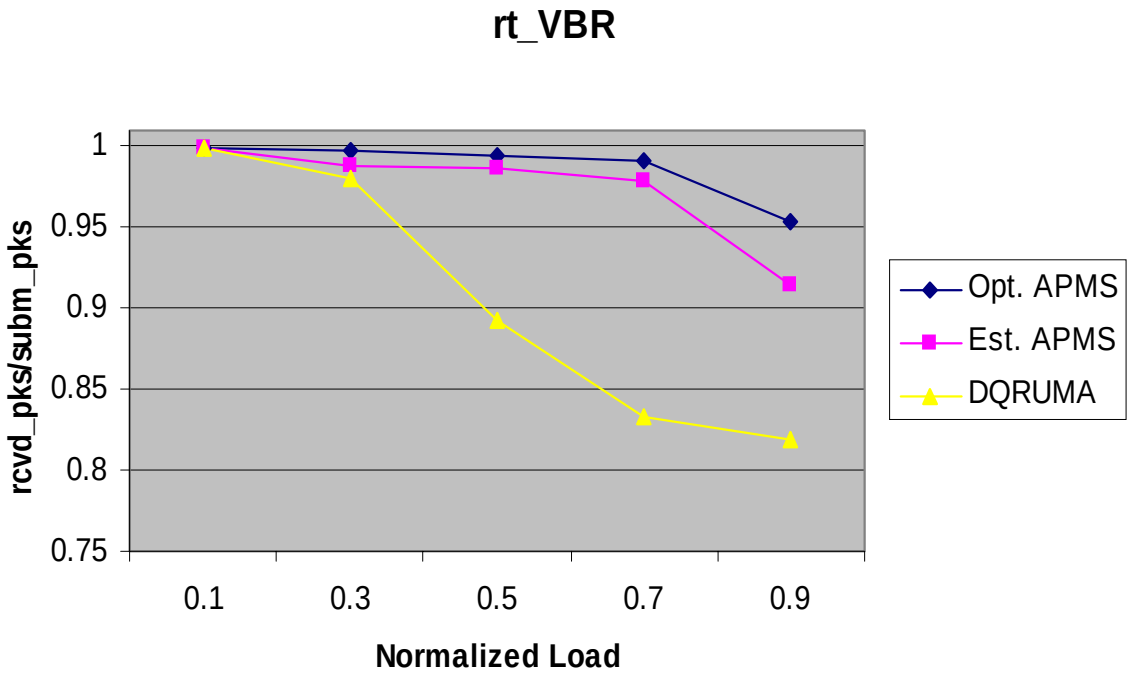


Figure 6.4: Relative Throughput for rt-VBR Traffic

For rt-VBR traffic, we can see similar trends appear, but with slightly less success as the normalized load approaches higher values. This is attributed to the fact that rt-VBR traffic is given a lesser priority than CBR traffic within an MU queue service, and also due to the fact that rt-VBR consumes approximately 10 times more load than CBR in a given period. This results in more rt-VBR traffic being dropped from the queue due to the maximum delay constraint of 20 milliseconds.

Overall, however, it is seen that more than 85% of the total packets in the system that were in error using the DQRUMA protocol are converted into valid packets when the APMS protocol is used, dramatically improving the relative throughput of the system.

The confidence intervals for the mean of the throughputs, our primary response variable, are gathered using the t-variate with *n*-1 degree of freedom, where *n* equals the number of samples that were taken. The 100(1-α)% confidence interval is given by the following equation:

$$(\bar{x} - t_{[1-\alpha/2;n-1]} s/\sqrt{n}, \bar{x} + t_{[1-\alpha/2;n-1]} s/\sqrt{n}), \qquad 6.1$$

where, $t_{[1-\alpha/2;n-1]} s/\sqrt{n}$ is the $(1-\alpha/2)$-quantile of a *t*-variate with *n*-1 degree of freedom.

The confidence interval values obtained for the "Est." APMS in Figures 6.1 to 6.4 are shown in Table 6.2, with the values of $n = 5$ and $\alpha = 0.10$.

Table 6.2: The Confidence Interval for the Average Relative Throughputs

Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00042	0.00022	8.7E-05	5E-05	8E-06	3.7E-06	0.0	0.0
0.3	0.00122	0.00039	0.00043	0.00016	9E-06	1.2E-05	0.0	0.0
0.5	0.00123	0.00027	0.00045	0.00021	2.9E-05	5.4E-05	0.0	0.0
0.7	0.00064	0.00083	0.00129	0.00093	0.00027	0.00036	0.00052	0.00089
0.9	0.00316	0.00079	0.00659	0.00322	0.00045	0.00035	0.00019	0.00038

The low numbers indicate that the sample means of throughput derived through five replications closely match the true mean. The confidence interval tables for each combination of the simulation factors are presented in Appendix C.

6.1.2 Average Delays

We now investigate the secondary response variables, starting with the average delay. The average delay for the DQRUMA protocol with five different levels of load is shown in Figure 6.5.

The average delays of the four types of traffic on five different levels of load for the best performing APMS protocol (as determined in Section 6.1.1) are presented in Figure 6.6. Here, it is seen that the delays for the APMS case are relatively higher as the total load of the system increases. This is to be expected since the channel estimation algorithm is an integral part of the APMS protocol, and traffic that is perceived to be in the "bad" mode is withheld when the total channel capacity is not able to sustain all requests for data transfer. Their values are still well within the tolerable constraints imposed, which are 20 mS for real-time traffic and 1 second for non-real-time traffic.

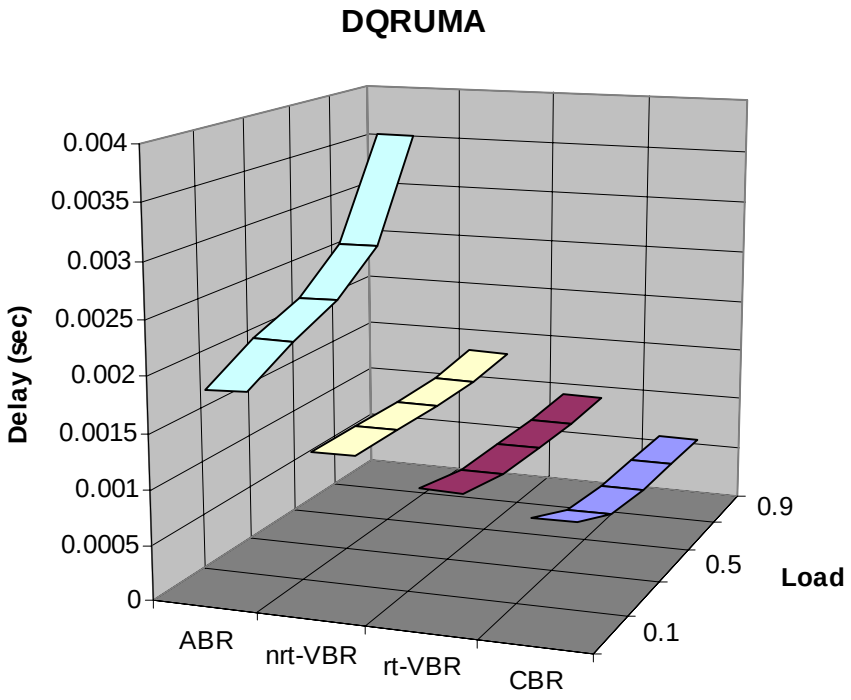


Figure 6.5: The Average Delays for the DQRUMA Protocol (in seconds)

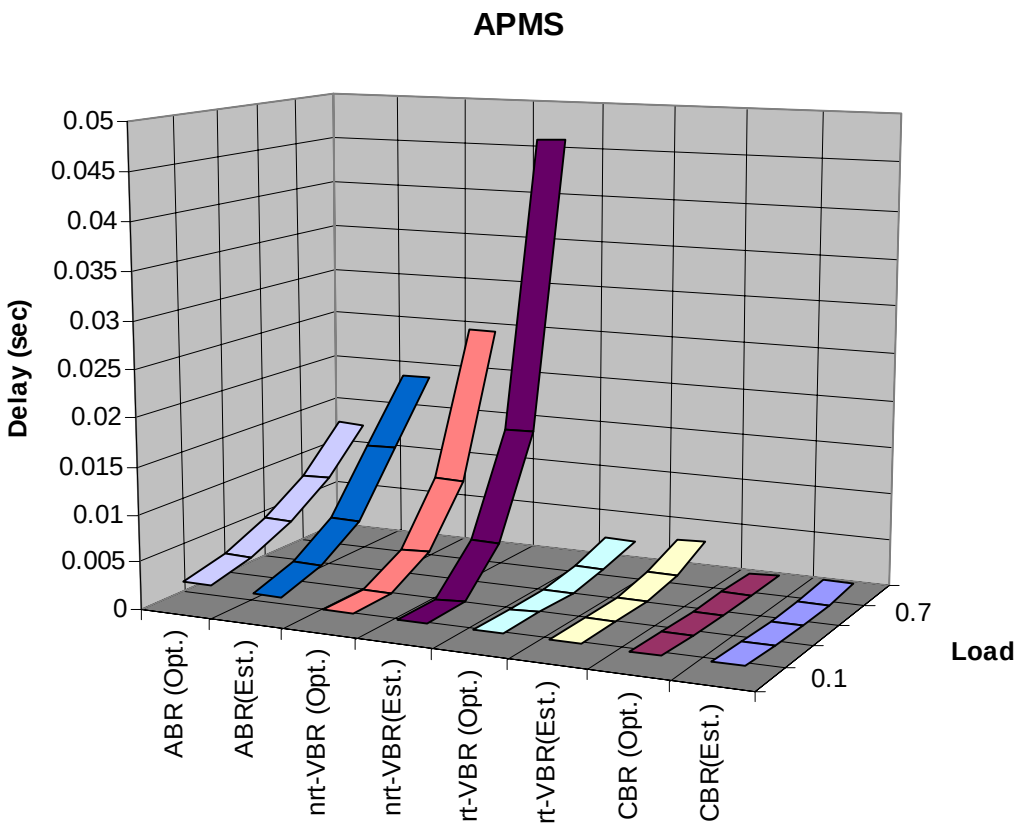


Figure 6.6: The Average Delays for the APMS Protocol (in seconds)

6.1.3 Number of Packets Dropped

The second secondary response variable of interest is the number of packets dropped due to the time constraints imposed. DQRUMA experienced no dropped packets due to the time constraints since the packets are "blindly" transmitted as soon as transmit permission is received from the BS. For APMS, no packets are dropped due to time constraints until the normalized load approaches the higher levels of 0.7 and 0.9. At the 0.7 normalized load level, 0.00052% of the CBR traffic and 0.325 % of the rt-VBR traffic are dropped due to the time constraint of 20 mS imposed on real-time traffic. The non-real-time traffic (nrt-VBR and ABR) are spared of any droppings due to the non-real-time time constraint of 1 second. At the 0.9 normalized load level,

0.0614% of the CBR traffic, 7.14% of the rt-VBR traffic, and 0.0083% of the nrt-VBR traffic are dropped. The ABR traffic is still spared of any droppings in this case. It is seen that, as the normalized load approaches the maximum capacity of the channel, the aggravation caused by the bursty channel condition becomes more difficult to control due to the lack of channel redundancy necessary to implement the APMS protocol. It is seen here that the bulk of the throughput decline witnessed in Figures 6.1 through 6.4 were actually packets that were dropped due to the time constraints imposed on them, rather than packets that were lost due to the "bad" states.

6.1.4 Number of Packets Exceeding Retransmission Limit

The third secondary response variable of interest, which only applies to the non-real-time traffic using the APMS protocol, is the number of packets exceeding the maximum retransmission limit. There were no packets that exceeded the retransmission limit of 3. This is due to the fact that, for Method 1 simulation, the loss in throughput for the higher levels of load occurs more because of the lack of opportunity to send and less because of errors during the transfer of packets.

6.1.5 Overall Comparison of APMS against DQRUMA in Method 1 Simulation

It was witnessed through the response variables that the APMS protocol adaptively adjusted to the channel condition by either utilizing the rake-in mode or avoiding the sending of packets when there is a high possibility of the channel being in the "bad" state. This adaptive nature of the APMS protocol sometimes resulted in dropped packets due to excessive prolonging of sending data from the time of creation. Nevertheless, the overall number of dropped packets for APMS were minimal compared to the number of packets that would have been lost if they had been sent without regards to the condition of the channel, as was the case for DQRUMA. Through this investigation, we have been able to verify that the APMS protocol greatly enhances the relative throughput, the primary response variable, of the system.

6.2 Multi Traffic Simulations Varying the Number of Active MUs (Method 2 Simulation)

Section 6.1 dealt with the case where the total number of active MUs in the system is limited to 5 (Method 1 simulations). This kept the number of collisions for the AR packets in the contention mode to a minimum. In this section, however, we investigate the case where normalized load within an MU is fixed, while the total number of active MUs in the system is varied from 10 MUs to 90 MUs in increments of 20, or from 0.1 to 0.9 normalized loads in increments of 0.2 (Method 2 simulations). This arrangement will necessarily increase the occurrence of collisions experienced by the AR packets, and it is investigated how this will affect the throughput of the system for both the APMS and DQRUMA protocols.

The simulation factors mentioned in Section 5.5 still hold for all the instances of simulation. The simulation factors are handled with the same thoroughness as in Section 6.1 by exploring the performance of the response variables for all the combinations mentioned in Section 5.5.

This comprehensive approach, once again, results in the following simulation runs for the APMS simulation model: all levels of LOAD (5), three levels of MAX_RETRANS (3), two levels of RAKE_IN_LEVEL (2), both levels of CHAN_EST (2), all levels of RELEASE (3), and 5 levels of simulation seeds. As in Section 6.1, this yields 900 simulations. The simulation model, similarly, also fixes the RAKE_IN LEVEL factor to 1 and the MAX_RETRANS to 0 for implementing the DQRUMA model. The LOAD (5) and 5 simulation seed levels are used to form another 25 simulations, as was the case in Section 6.1. The total simulation runs for this section was thus 925, the exact same number as was run in Section 6.1.

6.2.1 Relative Throughput

The throughputs obtained for Method 2 of our simulation showed that, compared to the DQRUMA protocol, our APMS method was able to improve upon the relative throughput of the system, but with both results suffering greater loss in comparison with their corresponding simulation results obtained in Section 6.1. The 25 transmitter-receiver pairs allocated to the access request period limit the number of simultaneous AR packets from the MUs that are able to reach the BS without collision. The relative throughput results for each type of traffic and their

average for Method 2 simulations are shown in Table 6.3. A detailed collection of all the relative throughput results is shown in Appendix E.

Table 6.3: The Average Relative Throughput Results for Method2 Simulations

RTRN	RL	RLS	CBR	rt-VBR	nrt-VBR	ABR	Average	Rank
0	4	20	0.91833	0.91955	0.91723	0.92892	0.92101	
0	4	30	0.93014	0.93225	0.93075	0.94614	0.93482	
0	4	40	0.93300	0.93600	0.93551	0.95040	0.93873	
0	6	20	0.92454	0.92284	0.91466	0.91653	0.91964	
0	6	30	0.93296	0.93270	0.92403	0.92962	0.92983	
0	6	40	0.93520	0.93428	0.92582	0.93432	0.93241	
3	4	20	0.92373	0.92374	0.98018	0.98723	0.95372	
3	4	30	0.93175	0.93435	0.98398	0.98668	0.95919	
3	4	40	0.94004	0.94180	0.98865	0.99110	0.96540	1
3	6	20	0.92118	0.92178	0.97935	0.98610	0.95210	
3	6	30	0.93466	0.93386	0.98743	0.99229	0.96206	
3	6	40	0.93565	0.93533	0.98408	0.98720	0.96057	
5	4	20	0.92317	0.92432	0.98383	0.98918	0.95513	
5	4	30	0.93131	0.93382	0.98511	0.98737	0.95940	
5	4	40	0.93760	0.93896	0.98452	0.98689	0.96199	
5	6	20	0.92192	0.92211	0.98404	0.98807	0.95403	
5	6	30	0.93240	0.93255	0.98684	0.99121	0.96075	
5	6	40	0.93777	0.93586	0.98568	0.98940	0.96218	2
7	4	20	0.92321	0.92357	0.98405	0.98740	0.95456	
7	4	30	0.93382	0.93586	0.98705	0.99012	0.96171	
7	4	40	0.93507	0.93642	0.98179	0.98569	0.95974	
7	6	20	0.92211	0.92299	0.98518	0.98950	0.95494	
7	6	30	0.93450	0.93361	0.98854	0.99207	0.96218	3
7	6	40	0.93303	0.93338	0.98160	0.98621	0.95856	

In Table 6.3, the best performing combination of the simulation factors are RL value of 4, RELEASE count of 40, and MAX_RETRAN value of 3. This combination coincides with the best performing simulation combination obtained for Method 1, as shown in Table 6.1.

The relative throughput responses obtained for the non-real-time (nrt-VBR and ABR) traffic portion of the APMS and DQRUMA protocols are shown in Figures 6.7 and 6.8. Here, when the number of active MUs in the system is relatively small, the relative throughput for APMS protocol maintains near perfect performance, for up to the 0.3 normalized load level. The

DQRUMA protocol is seen to be performing well only at the 0.1 normalized load level. As the normalized load increases, the gap in the relative throughputs for the two protocols widens until the DQRUMA protocol reaches the steady state probability of the channel being in the "bad" mode.

An interesting phenomenon observed is that, as the normalized load increases, the performance of the "Est." APMS protocol actually surpasses the performance of the "Opt." APMS case. At first glance, this would be incredulous, since the "Opt." APMS is based on the premise that the BS has perfect knowledge of the channel conditions, while the "Est." APMS assumes the channel conditions are estimated through counting the number of packets in error. However, the fact that there are 90 MUs for 0.9 normalized load means that the delay incurred by the collisions during the access request periods will be quite significant. Thus, although the rake-in mode is available for the non-real-time traffic more quickly for the "Opt." APMS, it is likely that there is greater possibility of excessive waiting time experienced by the non-real-time traffic. This is because of the extra channel capacity needed by the rake-in mode, which in turn means there is greater probability that the maximum channel capacity will be reached more quickly with fewer amounts of the original data. The time constraint imposed on the non-real-time traffic, however lenient it is, forces many packets to be dropped.

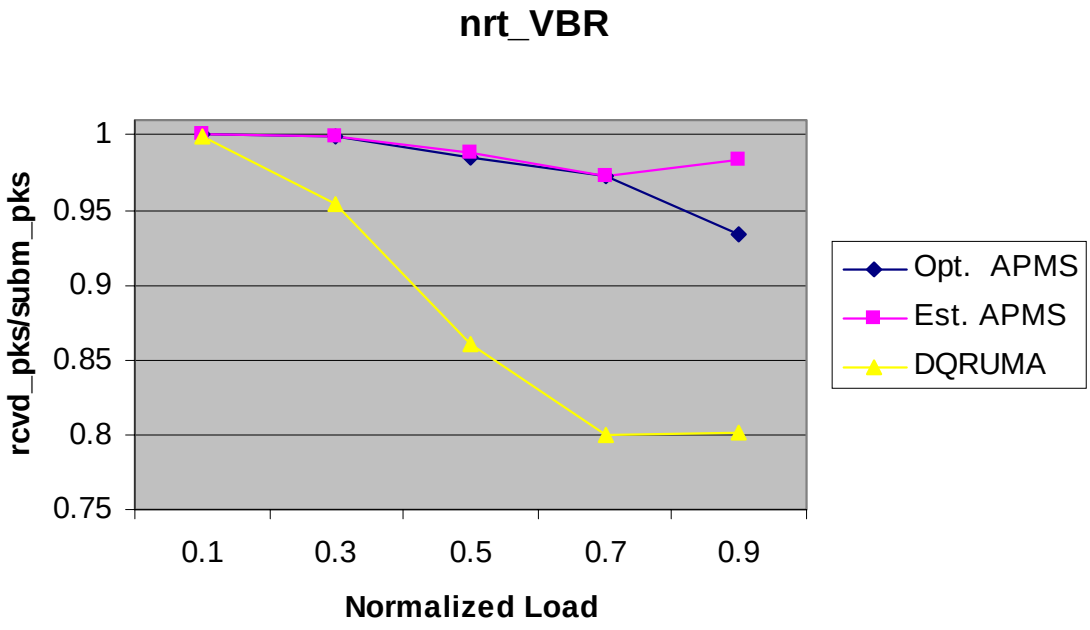


Figure 6.7: Relative Throughput for nrt-VBR Traffic

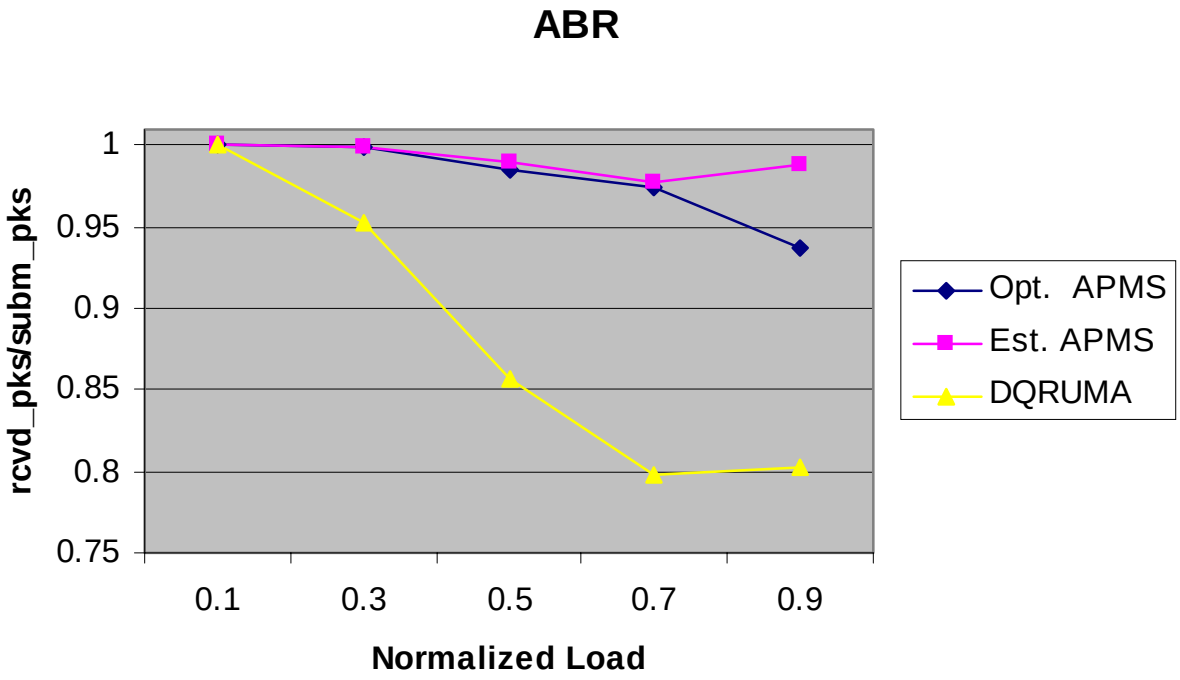


Figure 6.8: Relative Throughput for ABR Traffic

For the "Est." APMS, however, the number of packets dropped because of time constraints is roughly only one fifth the value for "Opt." APMS. This is primarily due to the criterion that more than 60% of the packets in two previous consecutive time frames must be in error before a rake-in mode, which takes up more channel capacity, is declared. Thus, the threshold needed to trigger the rake-in mode will be more robust, resulting in a lower number of packets being dropped due to excessive time delay. The slight rise in the "Est." APMS relative throughput at the 0.9 load level compared to the 0.7 load level can be explained by the fact that the piggy-backed requests in the non-contention mode are more likely to occur at the load near the maximum capacity level. The relative throughput for the "Opt." APMS, however is not able to benefit from this as more packets are dropped due to the surpassed time constraints for the non-real-time packets. In a large scale, though, the improvements are still impressive with near perfect performance at the lower load levels (0.1 and 0.3 load levels) and from around 85 to 95% of the packets that were in error for the DQRUMA protocol converting to error-free packets for the APMS protocol at the higher levels (0.5, 0.7, and 0.9 load levels).

The relative throughputs of the real-time traffic services (CBR and rt-VBR) employing the APMS protocol also showed statistically improved performance over the conventional DQRUMA protocol, as shown in Figure 6.9 and Figure 6.10.

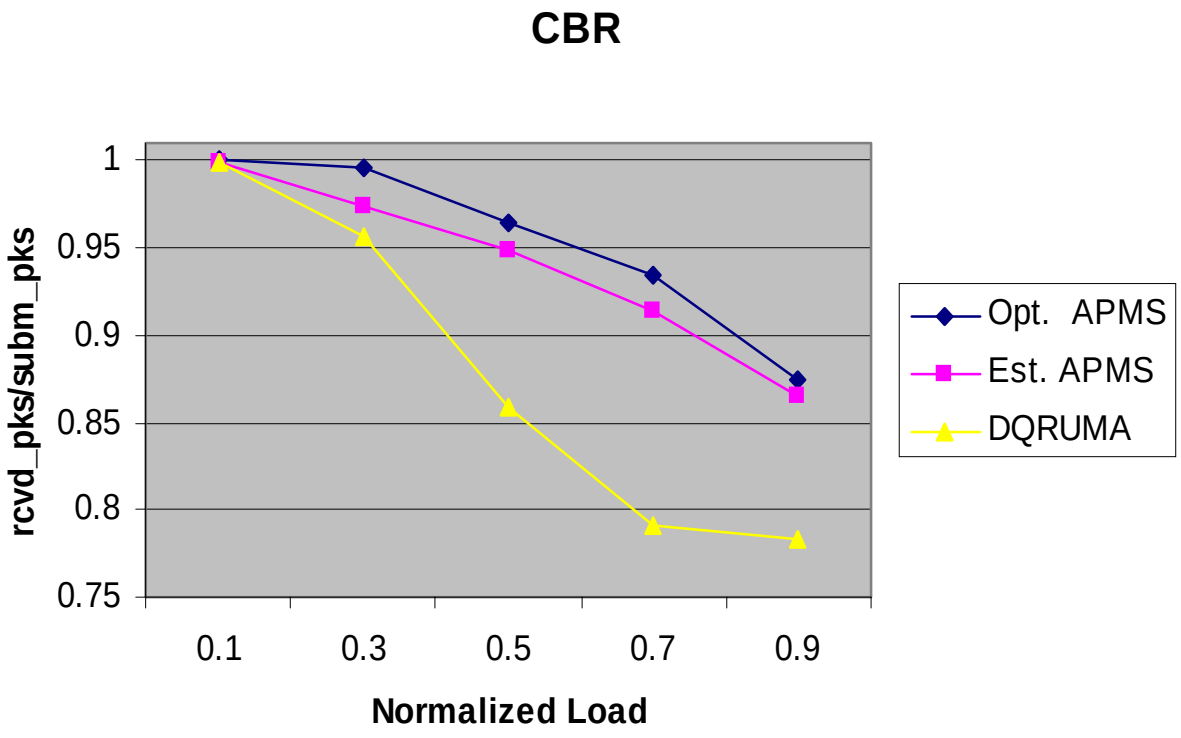


Figure 6.9: Relative Throughput for CBR Traffic

For the CBR traffic, as the normalized system load exceeded the 0.1 level, the relative throughput for both protocols decreased because of the "bad" periods and increasing number of collisions involved for AR packets. There were, however, noticeable differences in handling the adverse effects. Improvements brought on by employing the APMS protocol ranged from 9 to 12% over the conventional DQRUMA protocol at the higher load levels (0.5, 0,7 and 0.9 load levels). Although still a statistically significant result, compared to the improvements reported for the CBR traffic using Method 1, the result for Method 2 exhibited a less remarkable improvement for using the APMS protocol over the DQRUMA protocol. There are two significant factors contributing to the relatively lower improvements.

The first is that, given the same normalized load levels, the average traffic per MU is significantly less for Method 2 simulation than Method 1 simulation, causing the channel estimation mechanism to be relatively less accurate for Method 2 simulation. This means there would be more packets lost due to the bursty error conditions because of not implementing the rake-in mode, or equally devastating, imposing a rake-in mode when actually the channel state was in "good" mode, wasting the precious channel bandwidth that could have been utilized more productively for other traffic. The bulk of the loss in relative throughput is cause by this first factor of having rake-in modes at inaccurate times.

Second, there is a greater chance that the time constraint imposed for the real-time traffic will be reached. Thus, CBR data packets will be dropped, because of the higher probability of AR packets being in collisions.

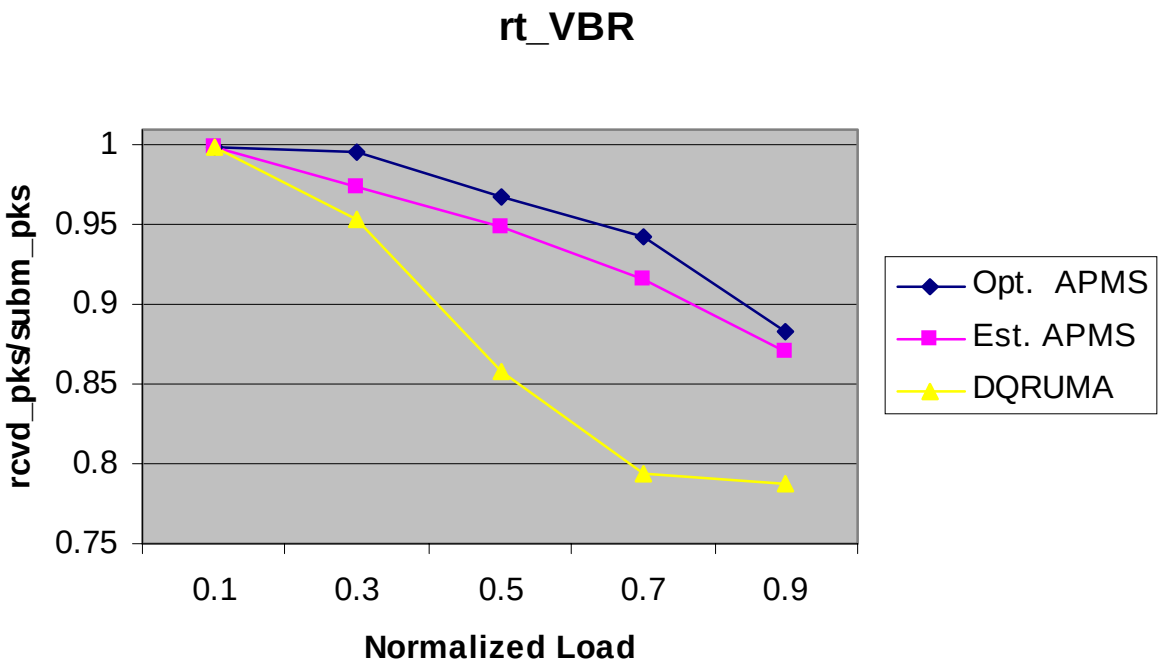


Figure 6.10: Relative Throughput for rt-VBR Traffic

For rt-VBR traffic, we can see almost identical simulation results as the CBR traffic case, unlike those in Method 1, where rt-VBR traffic performed relatively worse than CBR traffic. This reconfirms the fact that the bulk of loss in relative throughput is attributed to the inaccurate

estimation of the channel, since inaccurate channel condition assessment will lead to roughly the same portion of the data to be loss.

Overall, however, it is observed that nearly 70% of the total packets in the system that were in error using the DQRUMA protocol are converted into valid packets when the APMS protocol is used, a significant improvement in the relative throughput of the system. All of the t-Test results suggested that these results were statistically different with 95% probability. Tables for the t-Test results are listed in Appendix E.

The confidence interval values obtained for the "Est." APMS in Fig. 6.7 to 6.10 are shown in Table 6.4.

Table 6.4: The Confidence Interval for the Average Relative Throughputs

Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00064	7.3E-05	0.00011	6.3E-05	1.8E-05	1.9E-05	0.0	0.0
0.3	0.00137	0.00039	0.00045	0.00023	4.6E-05	2.4E-05	0.00103	0.00017
0.5	0.00852	0.01217	0.00715	0.01059	0.00722	0.0107	0.00853	0.01056
0.7	0.00955	0.01316	0.00918	0.00984	0.01082	0.0104	0.01255	0.00734
0.9	0.0049	0.01028	0.00291	0.00626	0.00441	0.00806	0.00459	0.00827

The numbers are generally relatively higher than those for Method 1 simulation, as was shown in Table 6.2, due to the more varying nature of the contention phase. However, the numbers are still very low, indicating that the sample means of throughput derived through 5 replications closely match the true mean. Comprehensive list of tables for each combination of the simulation factors are presented in Appendix F.

6.2.2 Average Delays

The average delays for the Method 2 simulations are now investigated. The average delays for the DQRUMA protocol are shown in Figure 6.11. Unlike the Method 1 simulation cases seen in Section 6.1, the average uplink delay results pertaining to Method 2 exhibit a much more wider range. This is to be expected since, as the number of MUs that are active increases, the likelihood that AR packets will collide increases. When a collision occurs, the AR packet must be sent out again after a backoff time period. This increases the time that a request to transmit

data for the MU reaches the BS. In addition, at the higher loading conditions, there are many MUs competing for permission to send, and further delay is possible in obtaining it. As was the case for Method 1, the delays for APMS case are relatively higher than DQRUMA as the total load of the system increases. Figure 6.12 depicts the case for the best performing APMS for Method 2, where the RL value of 4, RELEASE count of 40, and MAX_RETRAN value of 3 is employed. Once again, the APMS protocol adaptively postpones traffic that are perceived to be in the "bad" mode when the total channel capacity is not able to sustain all requests for data transfer. However, the average delay for each type of traffic still remains well below the time constraints imposed for real-time (20 mS) and non-real-time (1 second) traffic. The fact that the throughputs obtained for traffic using APMS mode are greater than those using DQRUMA, while the delays for traffic using APMS mode are worse than those using DQRUMA, illustrates that one of the tradeoffs APMS makes in gaining more throughput is time delay.

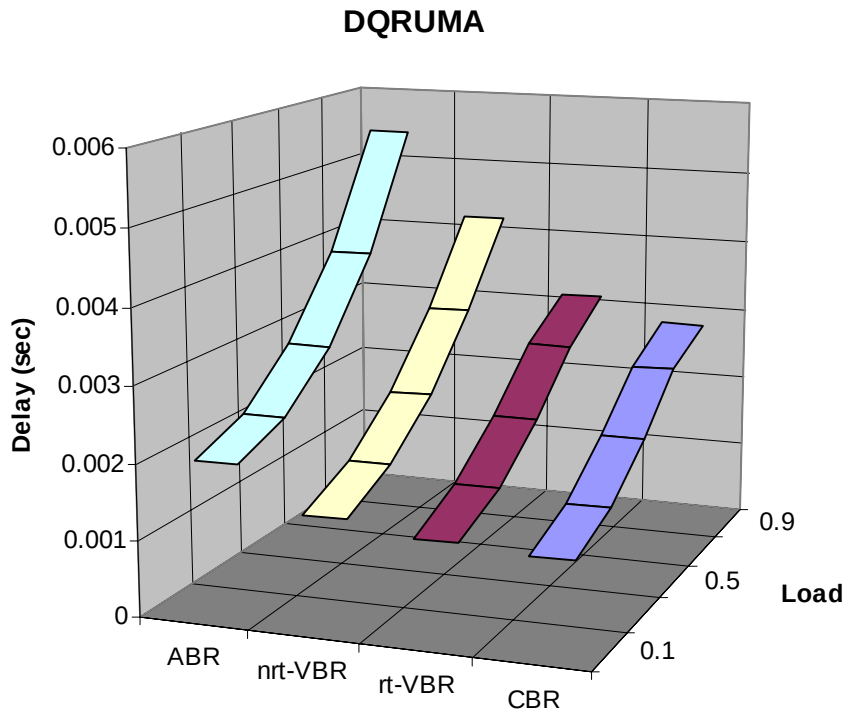


Figure 6.11: The Average Delays for the DQRUMA Protocol (in seconds)

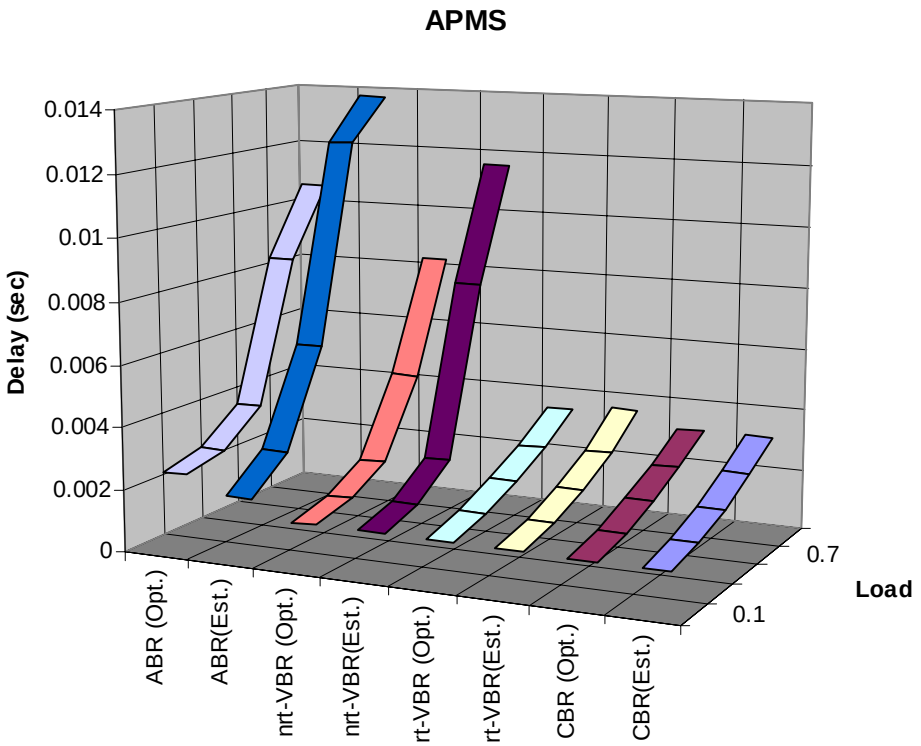


Figure 6.12: The Average Delays for the APMS Protocol (in seconds)

6.2.3 Number of Packets Dropped

In Method 2 simulation, there is an increase in the number of packets dropped due to the time constraints imposed. The results for the DQRUMA and APMS cases are shown in Figures 6.13 and 6.14, respectively. Unlike the Method 1 case, the DQRUMA experienced dropped packets due to the time constraints as low as the 0.3 load level. For the DQRUMA protocol at the 0.3 load level, the percentage of CBR, rt-VBR, nrt-VBR, and ABR packets that were dropped due to the time constraints is around 0.017, 0.032, 0.029, and 0.022%, respectively. The APMS, on average, had lesser number of packets dropped due to the time constraints at 0.3 load level. However, as the load level increased, the number of dropped packets from the APMS protocol exhibited slightly larger values for the real-time traffic but lower for the non-real-time traffic, compared to values obtained for the DQRUMA. This indicates that, unlike Method 1 simulation

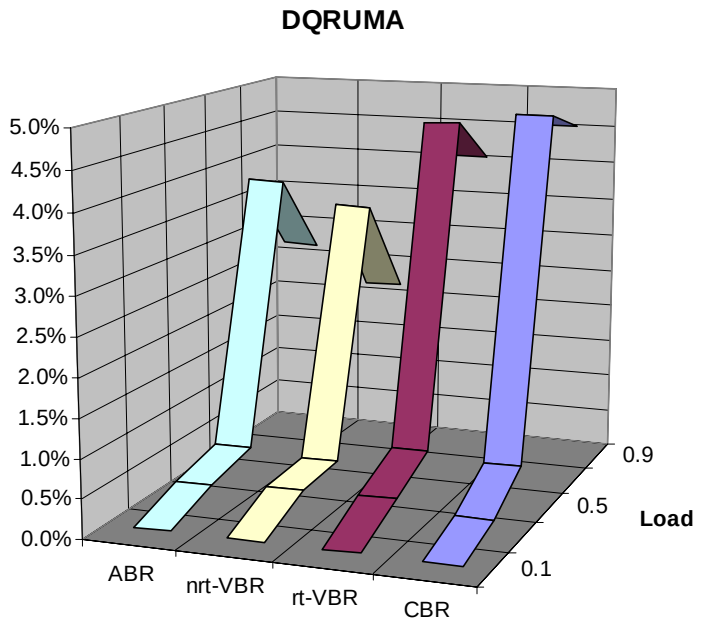


Figure 6.13: Percentage of Packets Dropped for the DQRUMA Protocol

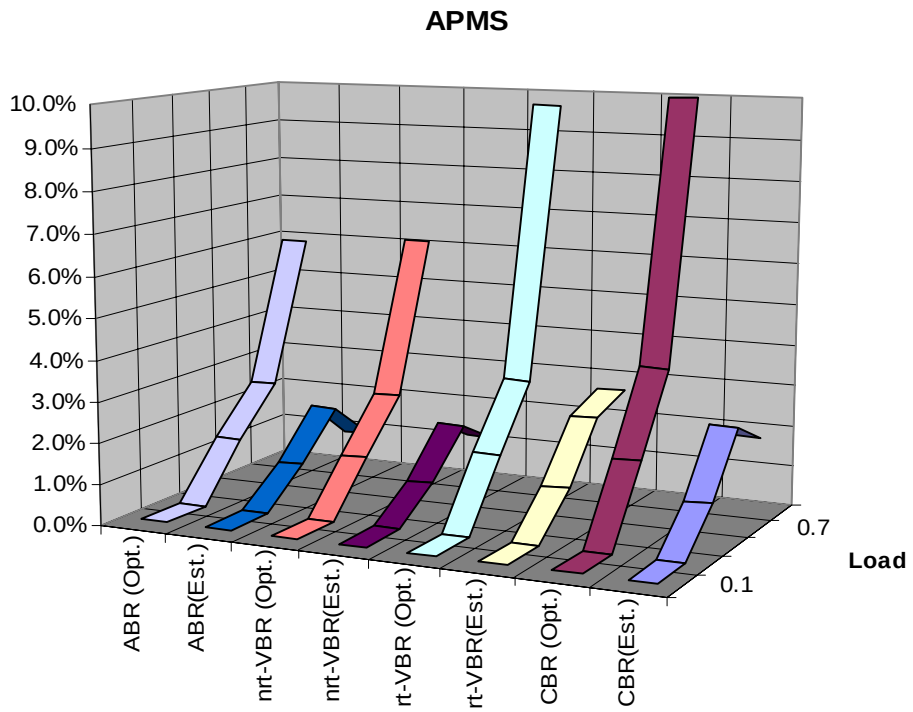


Figure 6.14: Percentage of Packets Dropped for the APMS Protocol

in Section 6.1, the larger number of MUs in the system had a greater influence on the number of packets dropped than the actual MAC protocol employed. This again justifies our dividing the simulation methods into two cases, Method 1 and Method2.

6.2.4 Number of Packets Exceeding Retransmission Limit

The third secondary response variable of interest, which only applies to the non-real-time traffic using the APMS protocol, is the number of packets exceeding the maximum retransmission limit. The best performing APMS had the MAX_RETRANS with a value of 3. It was observed that the packets lost due to excessive retransmissions were minimum; when the normalized load was at the 0.1 level, only 0.111% of the nrt-VBR packets and 0.133% of the ABR packets were lost, and even at the 0.9 level, only 0.288% and 0.425% of the packets were lost due to excessive retransmissions. For all the load levels, it is seen that the "Est." APMS simulation yielded relatively better results than "Opt." APMS. The reasoning for this follows what is stated in Section 6.2.1.

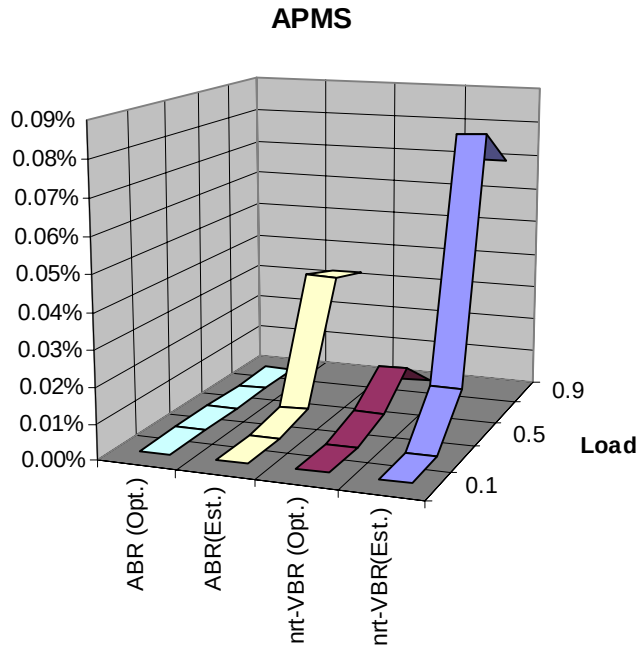


Figure 6.15: Percentage of Packets Exceeding the Retransmission Limit

6.2.5 Overall Comparison of APMS against DQRUMA in Method 2 Simulation

It was noted that as the number of MUs in the system increased, the increase in the collisions caused by AR packets in the contention mode contributed to the overall delay of the system. This, in turn, caused considerable amount of packets to be lost, for both DQRUMA and APMS protocols, due to the time constraints. Moreover, it was observed that the increase in the number of users resulted in less traffic being generated per MU, resulting some waste of capacity when using APMS protocol. In fact, it was observed that the optimal channel knowledge did not convert directly to optimal performance, but rather a less sensitive channel estimation method actually yielded better results for non-real-time traffic. Overall, however, it was observed that, even under the condition where there were considerably more MUs that were active, the APMS protocol clearly outperformed the DQRUMA protocol.

6.3 Other Simulation Studies

During the course of the study, we have been able to verify that the rake-in mechanism and retransmission scheme inherent in our APMS protocol greatly enhanced the overall throughput of the system. In this section, three special cases are studied. Subsection 6.3.1 investigates the situation where the retransmission is also employed for the DQRUMA protocol. Subsection 6.3.2 examines the case where only non-real-time traffic (nrt-VBR and ABR) are generated by the MUs. The third special case deals with the environment where only real-time traffic (CBR and rt-VBR) exist. For all comparisons, the best performing factors discovered for APMS in Sections 6.1 and 6.2 are used, and the load is varied by changing number of MUs that are active in the system.

6.3.1 Comparison of APMS against DQRUMA with Retransmission Allowed for Non-real-time Traffic

We have, until now, adhered to what was provided in [LKZ96] to describe the DQRUMA protocol. Thus, we have not allowed any retransmission for the DQRUMA protocol at the MAC layer. However, it is conceivable that between the two corner stones of the APMS protocol (the rake-in mode and the retransmission of non-real-time traffic), the retransmission of non-real-time traffic is possible for the DQRUMA protocol with minimal complexity. Here, we investigate

how much improvement the DQRUMA protocol can make against the APMS protocol with retransmission allowed for both protocols. Among the maximum retransmissions investigated, MAX_RETRANS of 7 performed the best in terms of the average relative throughput for DQRUMA. The DQRUMA plots in all the graphs assumes that maximum retransmission of 7 is allowed for non-real-time traffic. However, since the four traffic types are correlated for a given combination of factors, all four traffic types are investigated.

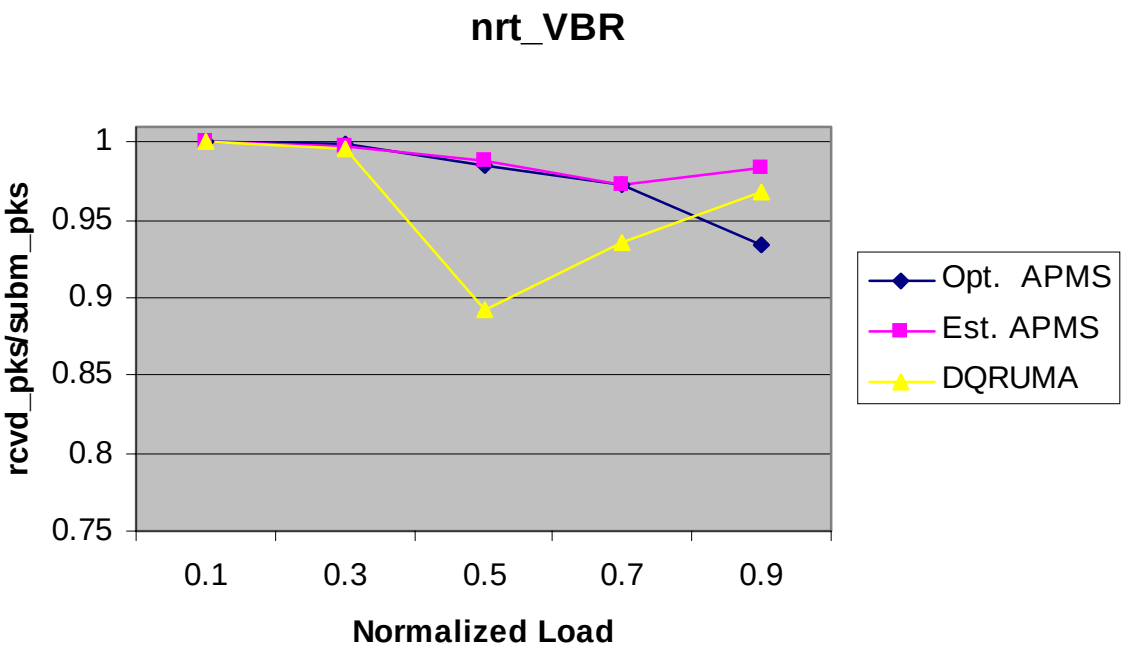


Figure 6.16: Relative Throughput for nrt-VBR Traffic

Figure 6.16 depicts the relative throughput results given for the nrt-VBR traffic. Here, the performance of the nrt-VBR traffic for the 7 retransmission allowed DQRUMA protocol is seen following a slightly improved performance, up to the 0.5 load level, compared to the case when no retransmission was allowed for DQRUMA, as was seen in Figure 6.7. However, as the load level reaches the 0.7 level, a there is a noticeable rebound in performance. This sudden improvement can be attributed to the fact that as more and more MUs are active (0.7 and 0.9 load corresponds to 70 and 90 MUs being active, respectively) there is a greater chance that the time needed to acquire transmit permission will be long enough to actually wait out the burst error period. This, however, is not without repercussions as will be discussed further.

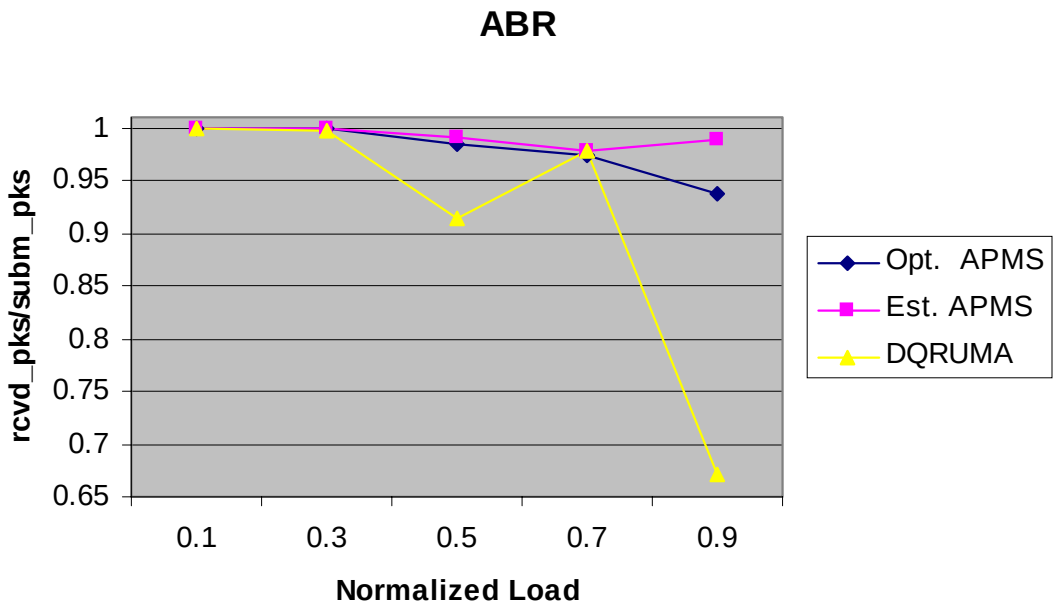


Figure 6.17: Relative Throughput for ABR Traffic

The ABR traffic on the other hand does not fully benefit from waiting out the burst error periods, as seen in Figure 6.17. Despite a temporary rebound at the 0.7 load level, the performance eventually drops below what was achieved without any retransmission being used for 0.9 load level. This is mainly due to the fact that ABR traffic is much more bursty in nature, causing many packets at the 0.9 level to be dropped while waiting for permission to send because of the time constraint imposed.

Real-time traffic is indirectly affected by the non-real-time traffic being allowed to retransmit their packets. Due to the increased activity of the retransmitting non-real-time packets, the time it takes for a packet waiting in queue to acquire transmit permission from the BS is increased, raising the probability that a real-time packet will be dropped due to time constraint. The results for the real-time traffic are given in Figures 6.18 and 6.19. It is observed that the curves for DQRUMA protocol resembles what was seen in Section 6.2, where no retransmission was allowed for DQRUMA, but degrading more rapidly toward the 0.8 relative throughput value.

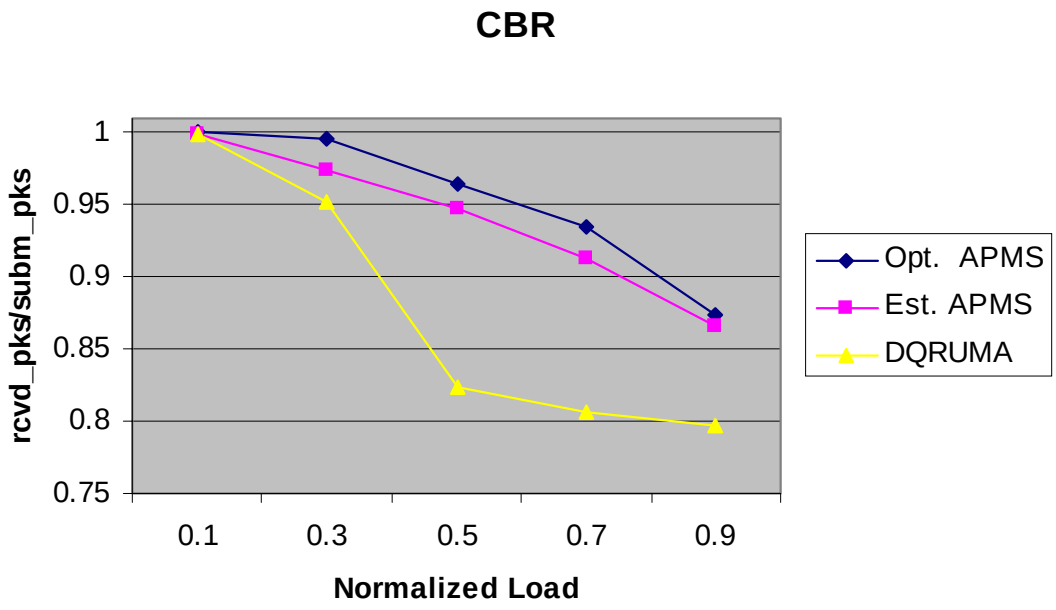


Figure 6.18: Relative Throughput for CBR Traffic

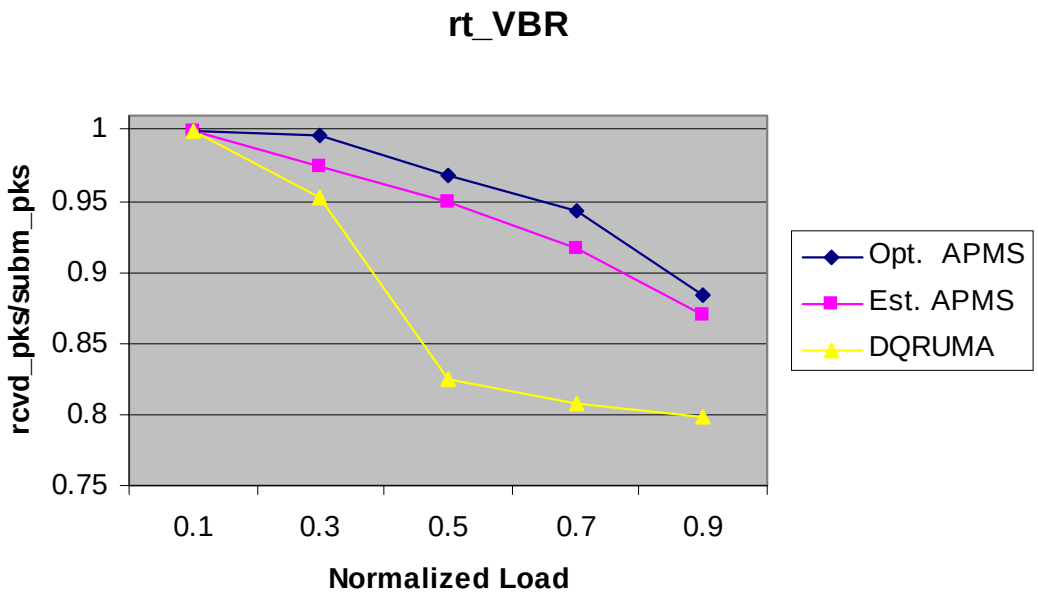


Figure 6.19: Relative Throughput for rt-VBR Traffic

In order to gain more insight into the throughput performance illustrated in Figures 6.16 to 6.19, we now describe the performance of the secondary response variables; uplink delay,

packets dropped due to time constraints, and packets discarded due to excess retransmission attempts.

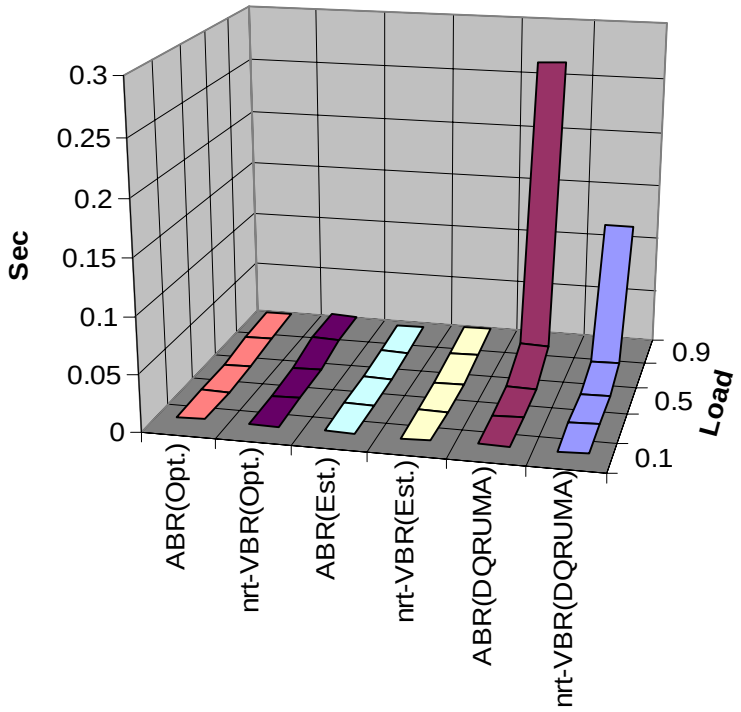


Figure 6.20: Delays for Non-real-time Traffic

First, the uplink delays for non-real-time traffic using DQRUMA and APMS are compared in Figure 6.20. Here, it is revealed that, as the load level approach 0.9 level, the average uplink delay abruptly soars for the non-real-time traffic using the DQRUMA protocol. As observed from Figure 6.17, ABR traffic suffered substantial loss in throughput at 0.9 load level due to a significant portion of the traffic exceeding the uplink delay constraint. For the nrt-VBR traffic, the loss in throughput was less affected due to the fact that nrt-VBR has higher priority in terms of being serviced by the BS. However, the performance for the nrt-VBR traffic is a delicate one, due to the rapidly increasing delay. In fact, the high level of nrt-VBR packets being kept in the queues of the MUs is part of the reasons why ABR traffic performs poorly. This is due to the fact that nrt-VBR has a higher priority in being serviced, and that nrt-VBR traffic has packets waiting in the queue of the MU most of the time.

The uplink delays for the real-time traffic using DQRUMA and APMS are presented in Figure 6.21. It is seen that, delays for real-time traffic using DQRUMA have stayed almost the

same compared to the case when no retransmission was allowed (Section 6.2). This is understandable since retransmissions only involve non-real-time traffic, and the real-time traffic are only indirectly affected by the fluctuation in the amount of non-real-time packets contending for bandwidth. Thus, as in the case in Section 6.2, the delays for DQRUMA are still less than those for APMS. This was explained that, during "bad" periods, APMS uses the rake-in mode, causing more bandwidth to be consumed for traffic using APMS, leaving some packets to be more delayed.

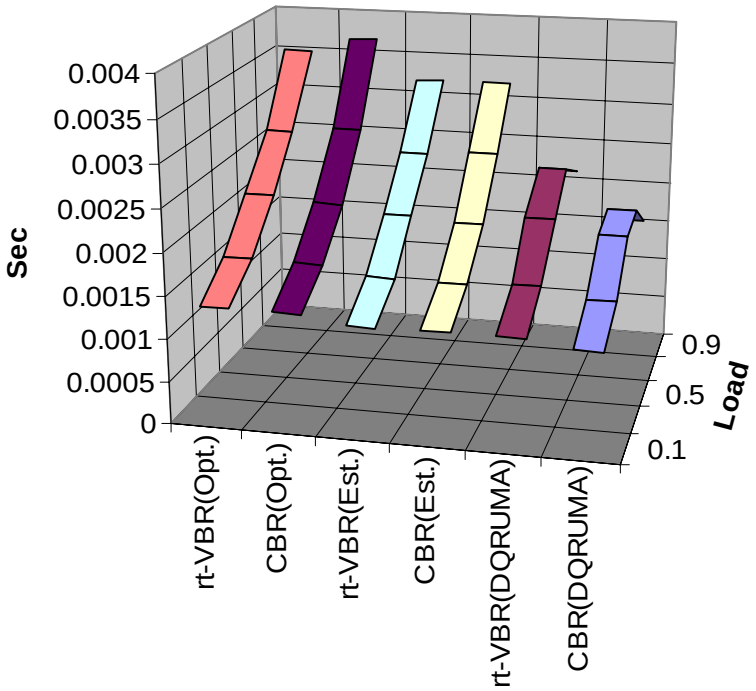


Figure 6.21: Delays for Real-time Traffic

The number of non-real-time packets dropped due to the non-real-time time constraint of 1 second is given in Figure 6.22. Here, it is seen that none of the non-real-time APMS traffic exceeded 5 % in dropped packets, while nearly 30 % of ABR traffic using DQRUMA is dropped at one point due to the time constraint. For the nrt-VBR traffic, however, there were noticeable improvements for the DQRUMA protocol having retransmissions allowed. Considering all traffic, however, allowing retransmissions alone for the DQRUMA protocol does not always improve throughput performance. As seen Figure 6.17, the relative throughput of ABR traffic

using DQRUMA with retransmission becomes much worse than when retransmission was not allowed for DQRUMA.

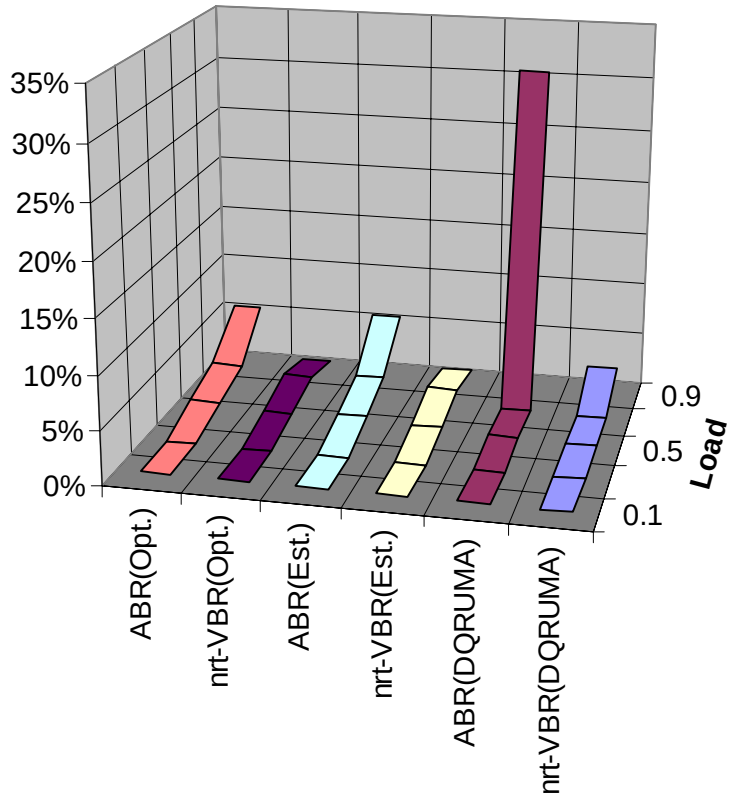


Figure 6.22: Dropped Packets Comparison for Non-real-time Traffic

The number of packets dropped due to time constraint for the real-time traffic is shown in Figure 6.23. It reveals that the time constraint imposed has a more damaging effect on the APMS than DQRUMA. This is another tradeoff that APMS is making in order to maintain a superior throughput performance than the DQRUMA protocol. That is, the real-time traffic using APMS go through periods where the rake-in mode is used. Using the rake-in mode occupies more bandwidth than when not using it, causing less source data to be transmitted during the "bad" periods where rake-in mode is employed. This increases the chance that real-time traffic using APMS will infringe upon the time constraint and be dropped. However, the traffic that are sent have a much greater chance of going through, resulting in better throughput overall, as evident in Figures 6.18 and 6.19.

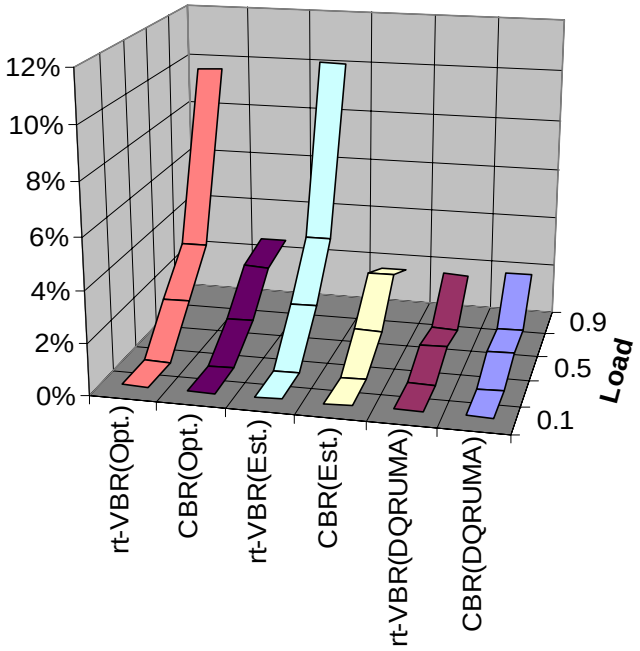


Figure 6.23: Dropped Packets Comparison for Real-time Traffic

The final secondary response variable is the number of packets exceeding the maximum retransmission limit. As explained at the start of this section, the DQRUMA was allowed to retransmit its non-real-time packets up to 7 times when needed. However, even with this many retransmissions allowed, more than 6% of the ABR traffic and 5% of nrt-VBR traffic were forced to be discarded due to excessive retransmission attempts. For APMS, less than 0.07% of nrt-VBR and 0.04% of ABR traffic were subjected to being discarded due to the maximum retransmission limit.

What is interesting about the DQRUMA case is that the peak loss came at 0.5 normalized load level for ABR traffic and at 0.7 normalized load level for nrt-VBR traffic, and not at the higher 0.9 normalized load level. At the 0.9 normalized load level, the number of packets exceeding the retransmission limit is markedly down. There are several factors contributing to this phenomenon.

The first is that at the highest level of load tested (0.9 level) the packets that are retransmitted incur more delay between retransmissions during the “bad” periods. And by the time that the retransmission attempt reaches 7, it is highly likely that the “bad” periods would have transformed into “good” periods. Second, at 0.9 load level, there is greater probability of a

non-real-time packet infringing upon the time constraint imposed before it has a chance to retransmit all 7 times.

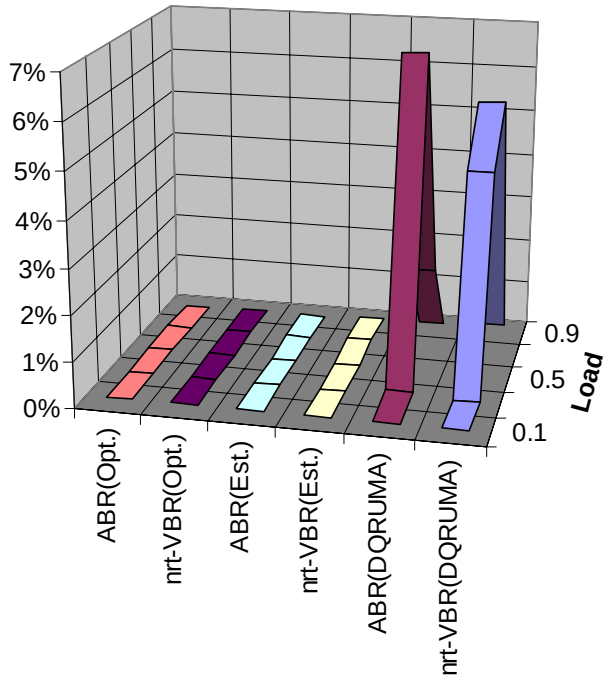


Figure 6.24: Maximum Retransmission Exceeded for Non-real-time Traffic

Overall, it is seen that some improvement in the non-real-time traffic throughput performance is achieved for the DQRUMA, but the performance is still well below the APMS protocol, which has a more sophisticated method of rendering transmit permissions and utilizing the rake-in mode during faulty channel periods.

6.3.2 Comparison of APMS against DQRUMA with Non-Real-time Traffic Only

The basic premise of this research is that there will be many types of traffic being generated from the MUs. It is, however, needed to verify that the APMS protocol will perform competently compared to the DQRUMA protocol under any skewed mix of traffic types. In this section, all traffic that are generated are non-real-time traffic (nrt-VBR and ABR). The performance of the APMS against DQRUMA protocol is investigated to see how robust the APMS is under this special circumstance and see whether the improvements in performance can

be maintained against the DQRUMA protocol. Method 2 simulation with the best combination of factors (RL value of 4, RELEASE count of 40, and MAX_RETRAN value of 3) for APMS is considered.

Compared to DQRUMA, the relative throughputs for the APMS (both "Est." and "Opt.") showed remarkable improvement in performance for the nrt-VBR traffic, as shown in Figure 6.25. The only noticeable loss of packets, however small, for the APMS protocol is from the 0.5 and 0.7 normalized load conditions, which can be attributed to heavy occurrence of AR packets during those levels of load. A closer inspection of the simulation results reveal that the small loss in the relative throughput performance for the APMS comes exclusively from packets being lost due to having exceeded the maximum retransmission allowed, as shown in Figure 23. As explained in Section 6.2, the AR packets, which are subjected to contentions, are actually reduced in their number when the load increases further, since request for access will be carried in PB_Req packets when packets in the queues of the MUs are left remaining after a Data packet is sent out to the BS. Thus, explaining the near perfect performance of the 0.9 normalized load level. The DQRUMA performance for its nrt-VBR traffic remained more or less the same as the one shown in Section 6.2 (Figure 6.7).

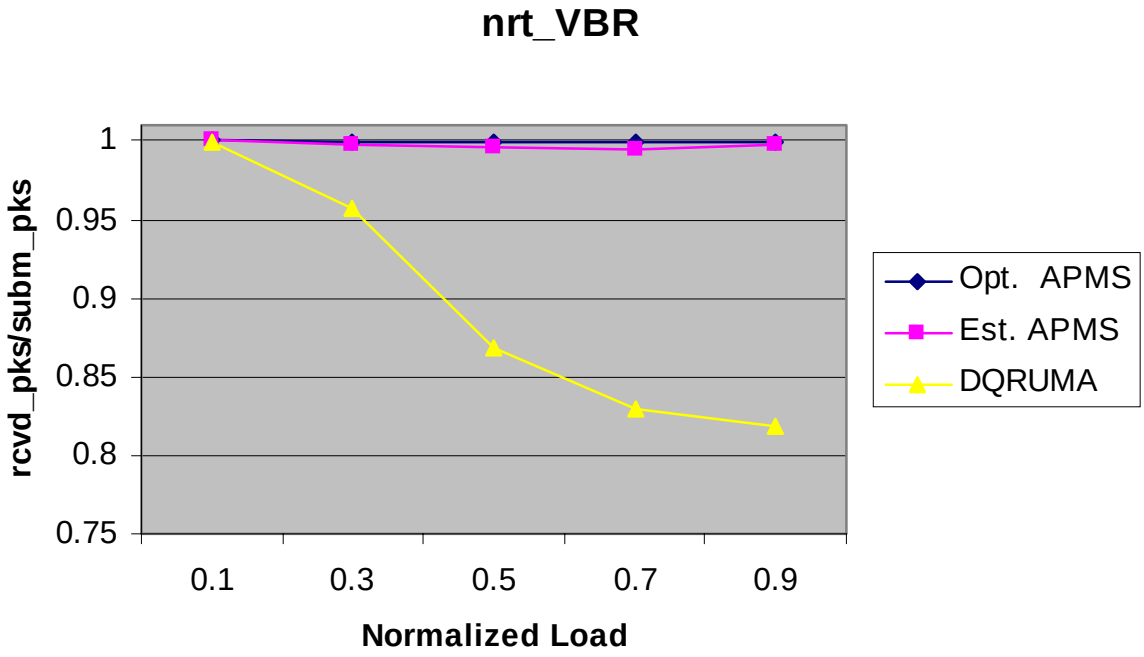


Figure 6.25: Relative Throughput for nrt-VBR Traffic

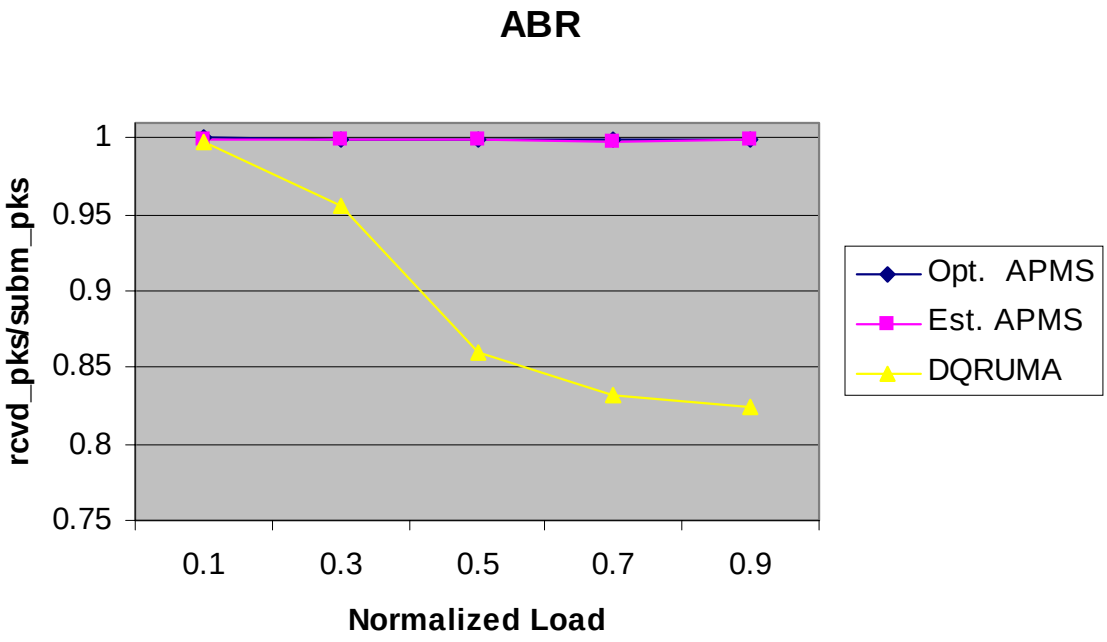


Figure 6.26: Relative Throughput for ABR Traffic

The ABR traffic showed even better performance for the APMS against DQRUMA, yielding almost perfect performance throughout all range of normalized load levels investigated. Although bursty in nature, on average the ABR traffic has less amount of traffic being generated compared to the nrt-VBR traffic and thus suffers less loss of packets due to exceeding the maximum retransmissions limit. The DQRUMA performance stayed almost the same as the one seen in Figure 6.8 in Section 6.2.

Overall, the relative throughput comparison between the APMS and DQRUMA protocols for non-real-time traffic revealed that APMS clearly had the advantage over the DQRUMA protocol under all load conditions. This is credited to the effective utilization of the channel by the APMS protocol through the use of retransmission, rake-in mode, and channel estimation. Due to the APMS protocol implementing the rake-in mode for channels going through "bad" periods, the delay of the packets for the APMS is 2 to 3 times longer than their counterparts using the DQRUMA. However, the delays are kept well within the 1 second time limit imposed on non-real-time traffic, as illustrated in Figure 6.27, and there was almost no loss of packets due to time constraints for this simulations in this section. The combination of these resulted in

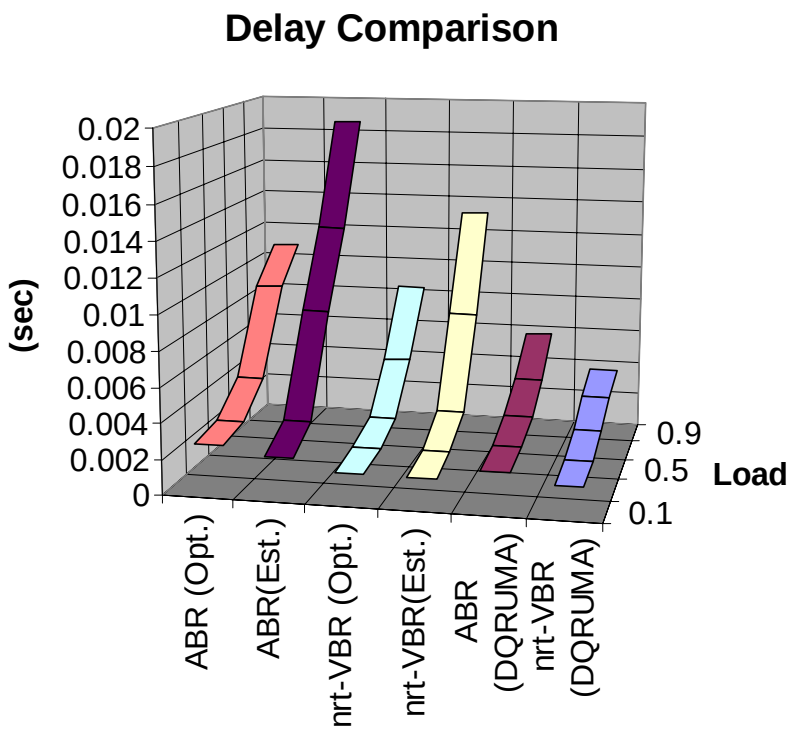


Figure 6.27: Delay Comparison for Non-real-time Traffic

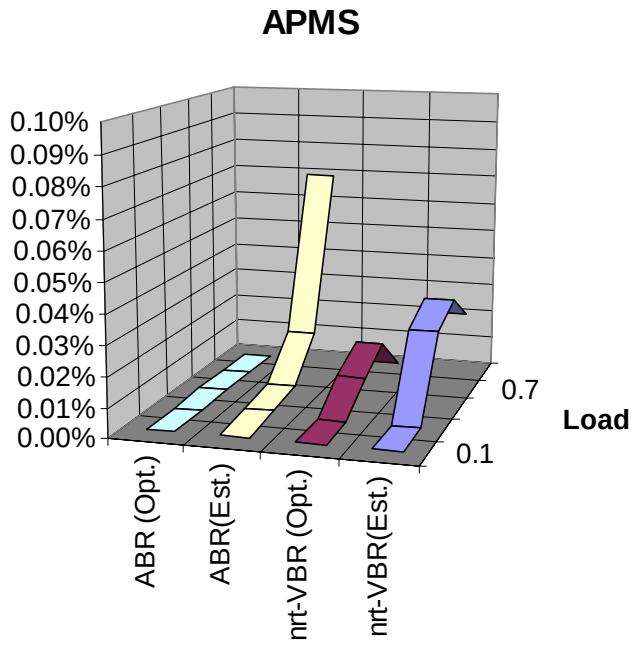


Figure 6.28: Percentage of Packets Exceeding Maximum Retransmission Limit

both the “Est.” and “Opt.” APMS protocol to perform nearly perfectly in terms of relative throughput for the non-real-time traffic, while the DQRUMA protocol suffered up to 18% loss in relative throughput.

6.3.3 Comparison of APMS against DQRUMA with Real-time Traffic Only

For this section, the distribution of traffic is reversed from what was done in Section 6.3.2, by incurring only real-time traffic, CBR and rt-VBR, from the MUs. Once again, the performance of the APMS against DQRUMA protocol is investigated to see how the APMS will perform under this special circumstance and see whether the improvements can be maintained against the DQRUMA protocol. Method 2 simulation with the best combination of factors (RL value of 4, RELEASE count of 40, and MAX_RETRAN value of 3) for APMS is considered.

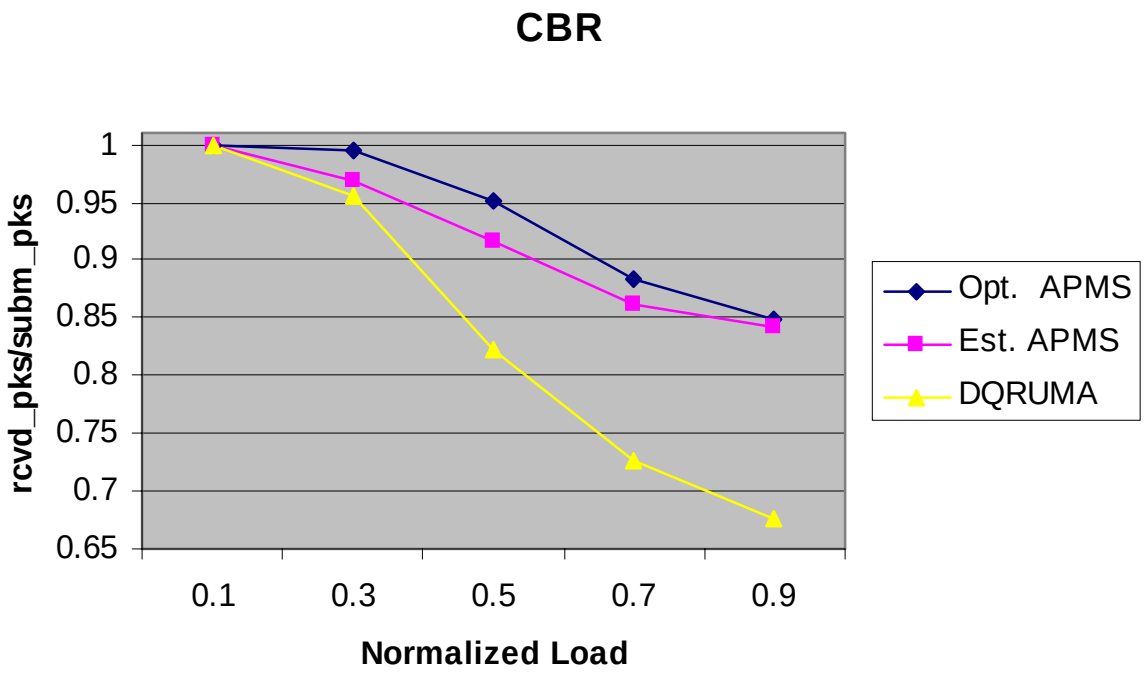


Figure 6.29: Relative Throughput for CBR Traffic

For real-time traffic, the performance for both the APMS and DQRUMA protocols were below the performance witnessed in Section 6.2. The relative throughput results for the CBR and rt-VBR traffic are shown in Figures 6.29 and 6.30. For the real-time traffic, it is seen that more than half of all the packets that were in error using the DQURMA protocol are salvaged using the APMS protocol. The proportion of the improvement of APMS protocol against the DQRUMA protocol is almost the same. What is different is that, compared to the performances given in Figures 6.9 and 6.10 of Section 6.2, Figures 6.29 and 6.30 reveal drops in performance of both APMS and DQRUMA protocols.

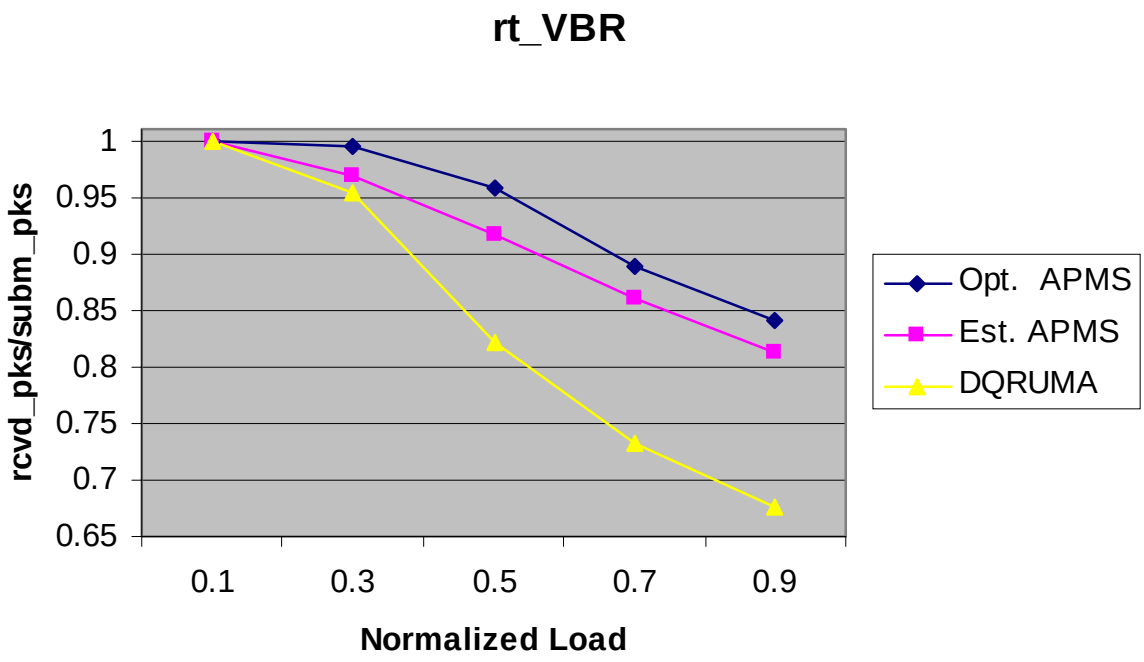


Figure 6.30: Relative Throughput for rt-VBR Traffic

This is caused by the fact that, unlike the case considered in Section 6.2, where there were both real-time and non-real-time traffic mixed, the traffic considered in this section are all real-time traffic with a stricter time constraint. Since all traffic are real-time with stricter time constraint, all the packets will be contending over the limited bandwidth with the stricter time constraint, causing many packets to be dropped as a result. It is shown in Figure 6.31 that the delay experienced by traffic using APMS are 20 to 30% higher than those using DQRUMA.

However, Figure 6.32 reveals that the number of packets dropped due to the time constraint is less for the packets using APMS than those using DQRUMA. The better performance by the APMS protocol is due to the fact that APMS give higher priority to MUs that are in the "good" period and also use the rake-in mode whenever there is redundant channel capacity left in the system, resulting in less packets being dropped, despite a slightly higher average uplink end-to-end delay.

Overall, the APMS protocol proved to be able to perform much better than the DQRUMA under the strict time constraint of 20 mS for the cases where only real-time traffic are generated.

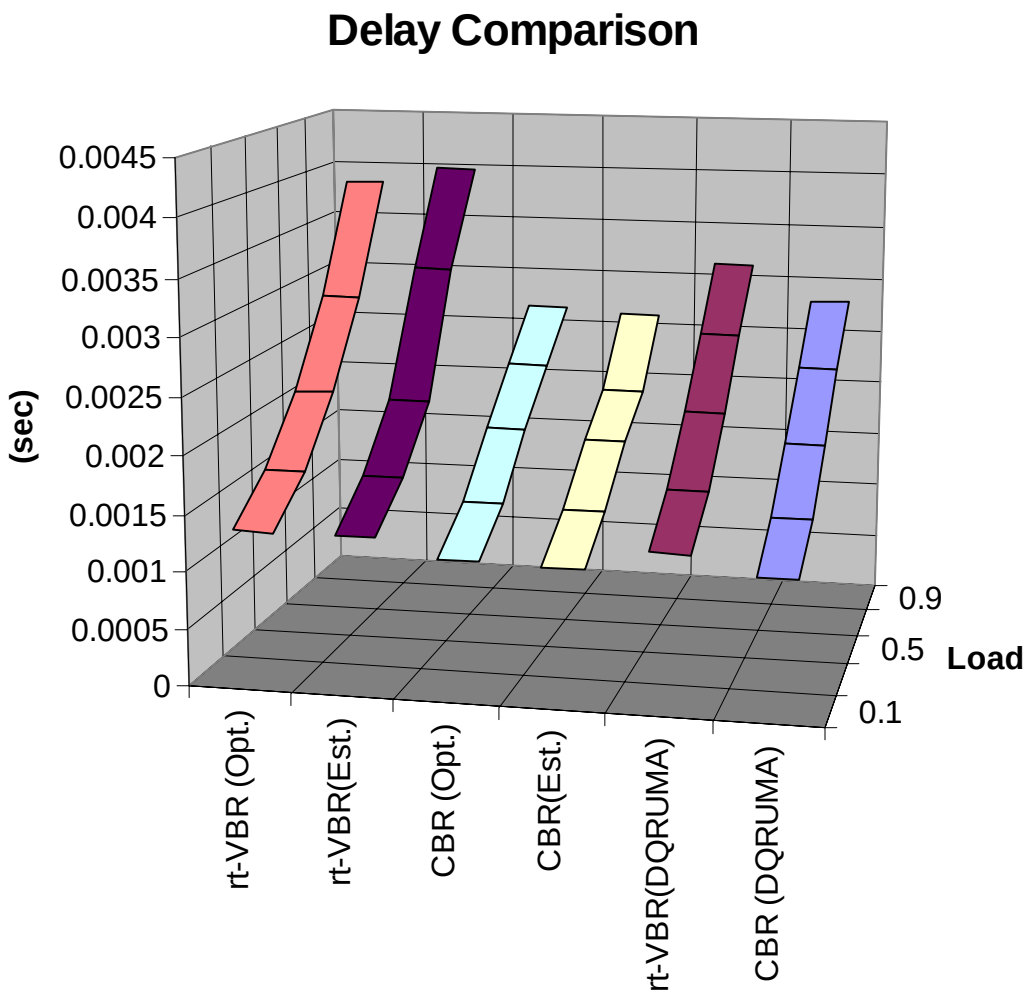


Figure 6.31: Delay Comparison for Real-time Traffic

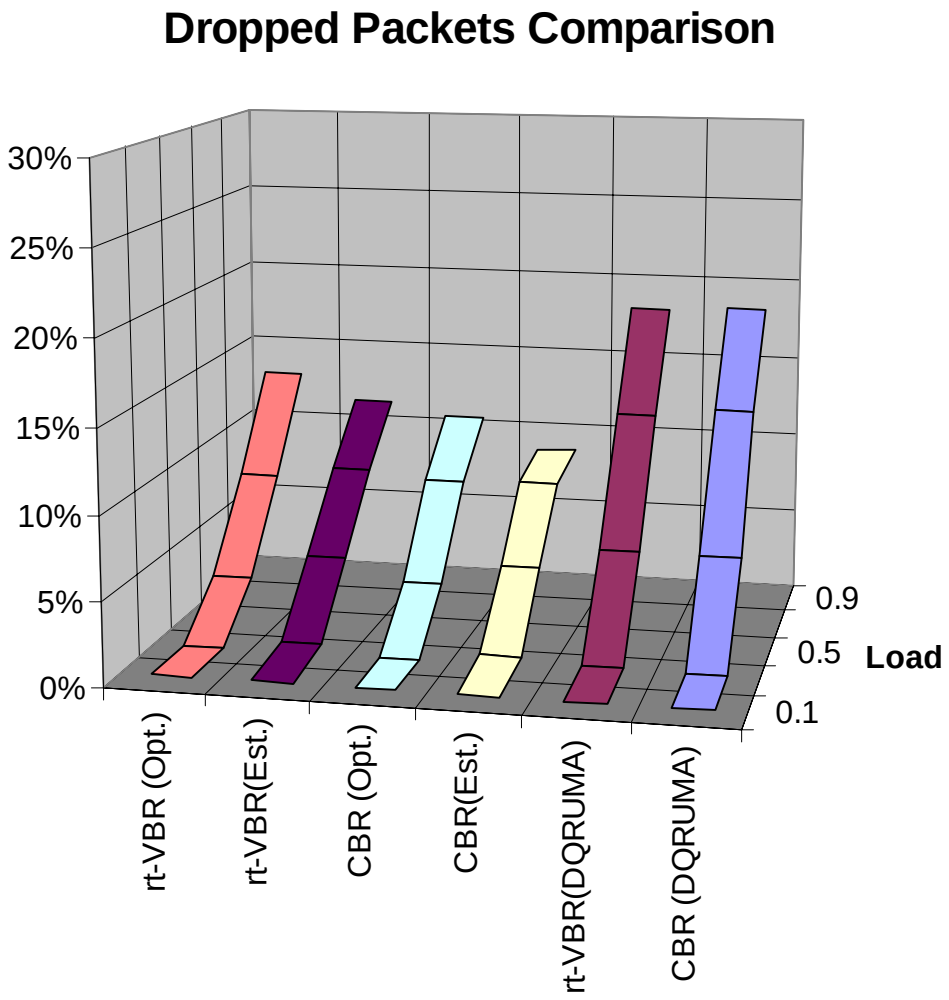


Figure 6.32: Dropped Packets Comparison for Real-time Traffic

6.4 Summary

In this chapter, various simulations were conducted to compare the APMS performance against the DQRUMA. The primary response variable with which the performance was measured was the relative throughput of the system. The supporting response variables were the uplink end-to-end delay, number of dropped packets, number of packets discarded due to excess retransmissions.

In Section 6.1, the performance of the APMS was compared against DQRUMA by fixing the total number of MUs in the system and varying the load levels within each MU (Method 1 Simulation). It was found that for non-real-time traffic (nrt-VBR and ABR), APMS approached the absolute perfect mark, while the DQRUMA suffered up to an 18% loss in relative throughput under the same faulty link conditions. For the real-time traffic, the DQRUMA lost up to 18% of its packets, while the APMS lost less than 8 % of the packets at the highest level of normalized load tested. Overall, nearly 85% of packets that were lost using the DQRUMA protocol, either by faulty channel or time constraint, were converted to valid packets using our APMS protocol.

In Section 6.2, the effect of the number of MUs varying was simulated. It was found that, compared to the results of the same load levels in Section 6.1, the relative throughputs for both APMS and DQRUMA decreased somewhat due to the extra delay introduced by the increasing number of AR packets that were subjected to contention. However, the APMS proved superior to the DQRUMA protocol, being able to convert over 60% of the packets in error using DQRUMA to error-free packets.

Three special cases were investigated in Section 6.3. The first special case was when retransmission was also allowed for the non-real-time traffic for the DQRUMA protocol. There were some improvements that the DQRUMA protocol made toward a better performing APMS for the non-real-time traffic, but overall, the APMS still maintained better results compared to DQRUMA. In the second special case, only non-real-time traffic was generated in the system. It was seen that APMS maintained almost perfect relative throughput performance, while the performance of DQRUMA essentially stayed the same as it was in Section 6.2. The third special case allowed only real-time traffic in the system. Here, both APMS and DQRUMA had around 30% worse performance than what was witnessed in Section 6.2, but APMS still managed to convert nearly half of the error packets using the DQRUMA protocol to error-free packets.

For all cases, the APMS performed markedly better than the DQRUMA protocol in terms of relative throughput under the same time constraints and channel conditions.

7. Conclusions

The primary purpose of this research effort was to explore the effectiveness that could be brought on by replicating the data being sent (rake-in mode) during periods of gross degradation in the wireless link of a wireless ATM system. Also, in order to deal with situations where the utilization of the channel was reaching the maximum channel capacity, channel estimation, selective transmit permission, and a retransmission scheme was employed to further enhance performance. It was discovered that, by implementing the sophisticated traffic management scheme at the base station (BS), there could exist a protocol that could perform better than the DQRUMA protocol in a wireless ATM system with multi-code CDMA air-interface. The resulting protocol for these purposes was named Adaptive Parallel Multiple Substream MAC (APMS) protocol. Extensive simulations were conducted to find the best combination of simulation factors that could optimize the performance of the APMS protocol against the DQRUMA protocol.

7.1 Summary of Research

Chapter 1 provided an introduction to the problem under investigation. The motivation for employing a multi-code CDMA was given. The goal of this research effort is to find a new MAC protocol that adapts to the network situation that affects the actual performance of the network as a whole. The new method that improves the overall network performance was proposed.

Chapter 2 presents the background information. Wireless ATM is a relatively new concept whose standard has not yet been mandated. ATM itself, however, has been standardized and it was looked into to appreciate the functionality of the ATM network. Various proposed

wireless ATM MAC protocols were compared against each other. It was concluded that, although TDMA-based MAC protocol has received the better part of the attentions, DS-CDMA-based MAC protocols had many advantages that could make it a better choice. The one disadvantage (the low maximum data rate) associated with the conventional DS-CDMA is overcome by structuring multiple substreams within a mobile unit, hence the use of multi-code CDMA air-interface. The effects of link degradation need to be taken into consideration to provide a more robust MAC protocol for wireless ATM.

The problem definition and objective of this research effort were provided in Chapter 3. The harsh reality faced by the wireless link was described to enlighten the need for a more robust wireless ATM MAC protocol that had the ability to adapt to the changing conditions of the channel. Assumptions made about the network under investigation were given. The primary performance metrics was identified as the throughput, and secondary response variables were identified which would support in explaining the outcome of this research.

Chapter 4 described the features of APMS-CDMA wireless ATM MAC protocol that enables the robustness against bit errors during burst bit error periods. The criterion used in deciding the various rake-in modes of the APMS protocol was addressed. The effect of the rake-in mode was analyzed and simulated to verify the effects of the rake-in mode to be used in APMS-CDMA MAC protocol. It was noted that the analysis matches the bit-level simulation results with respect to the various rake-in modes. Various traffic models (CBR, rt-VBR, nrt-VBR, and ABR) were examined. The retransmission scheme that is employed in our APMS protocol was explained.

Chapter 5 highlighted the some of the issues pertinent to the simulations. The method used to assess the channel capacity was described using previous works available in the literature related to the analysis of DS-CDMA wireless technology. Error detection and correction methods were discussed. The modeling of the bursty nature of the wireless channel was described. The simulation algorithms defining the key aspects of the APMS protocol were depicted. The simulation model validation against the conventional DQRUMA model was presented.

Various simulations results comparing the APMS performance against the DQRUMA were provided in Chapter 6. The first set of simulations obtained the performance of the APMS against DQRUMA by fixing the total number of MUs in the system and varying the load levels

within each MU (Method 1 Simulation). It was found that for non-real-time traffic (nrt-VBR and ABR), APMS approached the absolute perfect mark, while the DQRUMA suffered up to 18% loss in relative throughput under the same faulty link conditions. The APMS was able to perform better than DQRUMA also for the real-time traffic. When the number of MUs in the system was allowed to increase in size to vary the load, it was found that, compared to the results of the same load levels in the first set of simulations, the relative throughputs for both APMS and DQRUMA decreased somewhat. However, the APMS still proved superior to the DQRUMA protocol. Further investigation was conducted for three special cases. It was seen that APMS outperformed DQRUMA for essentially all cases.

7.2 Contributions

The performance of a MAC protocol is increasingly becoming an active research topic due to the ever-increasing demand for better throughput in wireless networks. However, there are few MAC protocol designs that provide a comprehensive solution to the different needs of multimedia traffic in the face of degrading channel conditions. Most adaptive MAC schemes do not consider multimedia and most MAC protocols dealing with multimedia do not consider the changing conditions of the wireless channel. This research effort addresses both the adaptive requirements and the different needs of the multimedia traffic under bursty channel conditions.

APMS, a comprehensive MAC protocol for a multi-code CDMA air-interface WATM, was developed to address both real-time and non-real-time traffic needs under faulty wireless link conditions. The real-time traffic was given higher priority to transmit data when the total access requests for transmission exceeded the maximum channel capacity to minimize the link delay. The non-real-time traffic was given permission to retransmit data at the MAC layer to meet often more stringent packet loss ratio requirements. Statistical multiplexing in both space and time was used to enhance the overall throughput of the WATM system.

The channel estimation used in this research effort did not require any additional overhead on the part of the hardware structure for the multi-code CDMA. The APMS protocol was able to use the estimation of the channel conditions to provide timely adjustments to the transmit permissions given to each of the MUs in the system. This resulted in efficient use of the

channel capacity for all the simulation factors considered in the study, as demonstrated by the results shown in Chapter 6.

7.3 Conclusions

Early in the research, it was determined that almost no packets in the "bad" period were able to cross the wireless link without error using the existing DQRUMA protocol for the multi-code CDMA WATM. The error packets accounted for almost 18% of the total packets that were attempted to be transmitted. The primary factor contributing to the heavy losses in relative throughput for the DQRUMA was that, according to [WiM99], the bit error rate suffered during the "bad" periods could be anywhere from 30 to 200 times more than during the "good" periods. The DQRUMA protocol did not have any provisions for dealing with this. By implementing the APMS protocol, under some special circumstances, such as exclusively non-real-time traffic, nearly perfect relative throughput performance was achievable, while the DQRUMA protocol still suffered nearly 18% loss under the same circumstances. For the general case where both real-time and non-real-time traffic were generated, around 50 to 60% of the real-time packets and 75% to 99% of the non-real-time packets that were in error using the DQURMA protocol converted to error-free packets by employing the APMS protocol.

For all the simulations, a simple, but effective channel estimation was assumed for the APMS protocol. However, a separate simulation assuming theoretically optimum channel knowledge ("Opt." APMS) was conducted for each simulation of regular channel estimation APMS ("Est." APMS). It was found that the regular APMS with channel estimation method performed quite closely to the "Opt." APMS, and in some isolated cases, actually surpassed the performance of "Opt." APMS.

This dissertation investigated the benefits that would be achievable by employing a novel adaptive medium access control protocol targeted to a multi-code CDMA air-interface wireless ATM. By providing a robust connectivity in face of changing conditions of the channels, the APMS protocol presented itself as an attractive alternative to an existing protocol, contributing in the on going efforts in the wireless ATM community to achieve a more reliable wireless link.

Although the context in which this research effort was conducted was a wireless ATM network environment, the protocol and techniques developed here can be applied to other

infrastructure wireless systems using multi-code CDMA as their air-interface. For example, a comparison between the SMPT mechanism, described in [FMW00] and introduced in Section 2.3.2.2, and APMS for 10 MUs in the system indicates that, whereas the best performing SMPT mechanism supports on average a relative throughput of 96.6%, the APMS protocol under similar conditions report a relative throughput of 99.9%. Also, existing CDMA systems, such as the IS-95 digital cellular system, support data and traffic at a basic rate as well as lower fractional rates [IWH96]. The "rake-in" mode proposed in this research could take advantage of the fractional rates to its advantage. In IS-95b (a revised version of the original IS-95 standard), multi-code CDMA is supported, which would once again allow the APMS protocol developed in this research effort to be implemented [NBK00]. The ongoing efforts in the third generation wireless systems also specify multi-code CDMA as one of the multirate solutions [OjP98].

Finally, independent of the air-interface technique employed, other wireless systems can benefit from the channel estimation and the traffic management techniques used in this research effort. For instance, for OFDM modulation technology being developed for IEEE 802.11a and HIPERLAN II, or the multi-carrier CDMA air-interface technology proposed for the third-generation wireless standard IMT-2000, the channel condition for those systems is not expected to be at a reliable level all the time. The channel estimation and diversified traffic management scheme for real-time and non-real-time traffic from this research effort, as discussed in Section 5.4, would be beneficial even in those circumstances as well.

7.4 Recommendations for Future Research

During the course of this research effort, there were issues that had the potential to further improve the overall throughput of the system, but have fallen outside the scope of this effort.

One of them is the backoff scheme used for managing AR packet contentions. We have followed what was used in [LKZ96], the harmonic backoff, but there are other backoff schemes, such as "exponential" backoff scheme, that could further alleviate the collision problems.

Second, we have only investigated a BCH code capable of correcting up to 3 bits in a 484-bit long packet, but continuous advancement in the VLSI technology suggest that a more ambitious assumption may be needed in the future.

Third, the effect of employing a multi-user detector would be worth investigating, since this would reduce the adverse impact of interference from other users in the system.

Fourth, the number of transmitter-receiver pairs does not have to be fixed, and it would be valuable to investigate the effects of varying the number of transmitter-receiver pairs.

Fifth, adaptive power control for different users in the system, as suggested by [WoL00], is a topic that would be helpful in exploiting transmitter adaptation to the system.

Finally, another way of controlling the data rate is to use variable spreading gain (VSG). This is an issue that has attracted the Third Generation (and beyond) Wireless communities around the world, resulting in some standardization entities around the world to employ VSG as part of their wireless MAC protocol [ZAB00].

Bibliography

[Aca96] A. Acompora, "Wireless ATM: A Perspective on Issues and Prospects," *IEEE Personal Communications*, vol. 3, pp. 8-17, August 1996.

[AcK97] A. S. Acampora and S. V. Krishnamurthy, "A New Adaptive MAC Layer Protocol for Wireless ATM Networks in Harsh Fading and Interference Environments," *IEEE International Conference on Universal Personal Communications (ICUPC'97)*, vol. 2, pp. 410-415, 1997.

[Agh94] A. H. Aghvami, "Future CDMA Cellular Mobile Systems Supporting Multi-Service Operation," *Proc. of PIMRC'94*, pp. 1276-1279, Sept. 1994.

[ATE95] C. Apostolas, R. Tafazolli, and E. Evans, "Wireless ATM LAN," *Proc. of PIMRC'95*, vol. 2, pp. 773-777, Toronto, Canada, 1995.

[ArL96] M. A. Arad and A. Leon-Garcia, "Scheduled CDMA: A Hybrid Multiple Access for Wireless ATM Networks," *Proc. of PIMRC'96*, v3, pp. 913-917, 1996.

[AzA94] H. Azad and A. H. Aghvami, "Multirate Spread Spectrum Direct Sequence CDMA Techniques," *IEE Colloquium on Spread Sectrum Techniques for Radio Communication Systems*, Digest No. 1994/098, pp. 4/1-4/5, IEE, April, 1994.

[BBF94] G. Bianchi, F. Borgonovo, *et al.*, "C-PRMA: The Centralized Packet Reservation Multiple Access for Local Wireless Communications," *Proceedings GLOBECOM'94*, San Francisco, CA, pp. 1340-1345, Nov. 1994.

[BBK97] P. Bhagwat, P. Bhattacharya, A. Krishna, et al., “Using Channel State Dependent Packet Scheduling to Improve TCP Throughput over Wireless LANs,” ACM Wireless Networks, vol. 3, pp. 91-102, 1997.

[BDT94] R. Bolla, F. Davoli, M. Taffone, and F. Reichert, “The PRMA-ISA Protocol for Multiple Access of Mixed Voice and Data Traffic,” in *Proceedings VTC'94*, vol. 2, pp. 1213-1217, June 1994.

[BeG92] D. Bertsekas and R. Gallager, *Data Networks, Second Edition*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1992.

[BLP99] F. Beritelli, A. Lombardo, S. Palazzo, and G. Schembra, “Performance Analysis of an ATM Multiplexer Loaded with VBR Traffic Generated by Multimode Speech Coders,” *IEEE J. Select. Areas Comm.*, vol. 17, no. 1, Jan. 1999, pp. 63-81.

[CaM95] J. M. Capone and L. F. Merakos, “Integrating Data Traffic into a CDMA Cellular Voice System,” *ACM Wireless Neworks*, vol. 1, no. 4, pp. 389-402, 1995.

[ChS97] S. Choi and K. G. Shin, “A Cellular Wireless Local Area Network with QoS Guarantees for Heterogeneous Traffic,” *Proceedings INFOCOM'97*, pp. 1030-1037, Kobe, Japan, May 1997.

[Cla96] Martin P. Clark, ATM Networks: *Principles and Use*; Wiley-Teubner, Chichester, England, 1996.

[Cou93] L. W. Couch II, *Digital and Analog Communication Systems, Fourth Edition*, Macmillan Publishing Company, New York, 1993.

[CuS93] L. G. Cuthbert and J-C Sapanel, *ATM: the Broadband Telecommunications Solution*, The Institute of Electrical Engineers, London, UK, 1993.

[Dix94] R. C. Dixon, *Spread Spectrum Systems with Commercial Applications*, Third Edition, John Wiley & Sons, Inc., New York, 1994.

[DRT97] R. Dube, C. Rais, and S. K. Tripathi, "Improving NFS Performance over Wireless Links," *IEEE Transactions on Computers*, vol. 46, no. 3, pp. 290-298, March 1997.

[DTS00] N. Dimitriou, R. Tafazolli, and G. Sfikas, "Quality of Service for Multimedia CDMA," *IEEE Communications Magazine*, vol. 38, no. 7, pp. 88-94, July 2000.

[EIA90] EIA/TIA Interim Standard, "Cellular System Dual Mode Mobile Station – Land Station Compatibility Specifications," IS-54, *Electronics Industries Association*, December 1992.

[Ell63] E. O. Elliott, "Estimates of Error Rates for Codes on Burst-Noise Channels," *Bell Systems Technical Journal*, Vol. 42, pp. 1977-1997, Sept. 1963.

[FGB97] K. Fu, Y. J. Guo, and S. K. Barton, "Performance of the EY-NPMA Protocol," *Wireless Personal Communications*, vol. 4, no. 1, pp.41-50, (Kluwer Academic Publishers) Jan. 1997.

[FKL86] T. Fujiwara, T. Kasami, and S. Lin, "Error Detecting Capabilities of the Shortened Hamming Codes Adopted for Error Detection in IEEE Standard 802.3," *Int. Symp. I.T.*, 1986, Ann Arbor, Michigan.

[Fla94] A. Flatman, "Wireless LANs: Developments in Technology and Standards," *Computing & Control Engineering Journal*, pp. 219-224, October 1994.

[FMW00] F. H. P. Fitzek, R. Morich, and A. Wolisz, "Comparison of Multi-Code Link-Layer Transmission Strategies in 3Gwireless CDMA," *IEEE Communications Magazine*, pp. 58-64, Oct. 2000.

[GeG91] E. Geraniotis and B. Ghaffari, "Performance of Binary and Quaternary Direct-Sequence Spread-Spectrum Multiple-Access Systems with Random Signature Sequences," *IEEE Transactions on Communications*, vol. 39, no. 5, pp. 713-724, May 1991.

[Gil60] E. N. Gilbert, "Capacity of a Burst-Noise Channel," *Bell Systems Technical Journal*, Vol. 39, pp. 1253-1265, 1960.

[GJP91] K. S. Gilhousen, I. M. Jacobs, R. Padovani et al., “On the Capacity of a Cellular CDMA System,” *IEEE Transactions on Vehicular Technology*, vol. 40, no. 2, pp. 303-311, May 1991.

[GVG89] D. J. Goodman, R. A. Valenzuela, K. T. Gayliard, and B. Ramamurthi, “Packet Reservation Multiple Access for Local Wireless Communications,” *IEEE Transactions on Communications*, vol. 37, pp. 885-890, Aug. 1989.

[HHS94] R. Handel, M. N. Huber, and Stefan Schroder, *ATM Networks: Concepts, Protocols, Applications*, Addison-Wesley, Cambridge, UK, 1994.

[Hue00] Mario R. Hueda, “Markov-based Model for Performance Evaluation in Multimedia CDMA Wireless Transmission,” *IEEE Vehicular Technology Conference*, vo. 2, no. 52, pp. 668-673, fall 2000.

[IEE85] ANSI/IEEE Std. 802.3-1985, Carrier Sense Multiple Access with Collision Detection (CSMA/CD), 1985.

[IEE99] Editors of IEEE 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Standard IEEE 802.11, Institute of Electrical and Electronics Engineers, Inc., New York, 1999.

[IEE99-a] Editors of IEEE 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High Speed Physical Layer in the 5 GHz Band, Standard IEEE 802.11a, Institute of Electrical and Electronics Engineers, Inc., New York, 1999.

[IEE99-b] Editors of IEEE 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High Speed Physical Layer Extension in the 2.4 GHz Band, Standard IEEE 802.11b, Institute of Electrical and Electronics Engineers, Inc., New York, 1999.

[IGi95] C-L I and R. D. Gitlin, “Multi-Code CDMA Wireless Personal Communications Networks,” *Proceedings ICC'95 Conference Record*, pp. 1060-1064, IEEE, Seattle, June 1995.

[INa96] C-L I and S. Nanda, “Load and Interference Based Demand Assignment (LIDA) for Integrated Services in CDMA Wireless Systems,” *Proceedings IEEE Global Telecommunications Conference*, vol. 1, pp. 235-241, 1996.

[IPO95] C-L I, G. P. Pollini, L. Ozarow, and R. D. Gitlin, “Performance of Multi-Code CDMA Wireless Personal Communications Networks,” *Proceedings VTC'95*, vol. 2, pp. 907-911, 1995.

[ITU91] ITU-T: Recommendation I.113. 'Vocabulary of Terms for Broadband Aspects of ISDN'. Rev. 1, Geneva, 1991.

[IWH96] C-L I, C. A. Webb, H. C. Huang, et al., “IS-95 enhancements for multimedia services,” Bell-Labs Technical Journal, v. 1, n. 2, pp. 60-87, Autumn, 1996.

[HaM00] Martin Haardt and Werner Mohr, “The Complete Solution for Third-Generation Wireless Communications: Two Modes on Air, One Winning Strategy,” *IEEE Personal Communications*, vol.7, no. 6, pp. 18-24, December 2000.

[Jai91] Raj Jain, *The Art of Computer Systems Performance Analysis*, John Wiley & Sons, Inc., New York, 1991.

[KiW96] J. G. Kim and I. Widjaja, “PRMA/DA: A New Media Access Control for Wireless ATM,” *Proceedings of ICC/SUPERCOMM'96*, vol. 1, pp. 240-244, June 1996.

[KrN96] J. Kruys and M. Niemi, “An Overview of Wireless ATM Standardization,” *ACTS Mobile Summit'96 Proceedings*, pp 250-255, 1996.

[LeM93] H. W. Lee and J. W. Mark, “ATM Network Traffic Characterization Using Two Types of On-Off Sources,” *IEEE INFOCOM'93*, pp. 152-159, 1993.

[LiC83] Shu Lin, Daniel J. Costello,Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 1983.

[Liu96] Zhao Liu, *Medium Access Control Schemes for DS-CDMA Wireless Packet Networks*, PhD thesis, Department of Electrical Engineering, The University of Pennsylvania, Philadelphia, PA, 1995.

[LKZ96] Z Liu, M. J. Karol, M. E. Zarki, and K. Y. Eng, "A Demand –Assignment Access Control for Multi-Code DS-CDMA Wireless Packet (ATM) Networks," *Proceedings of INFOCOM'96*, vol. 2, San Francisco, pp. 713-721, March 1996.

[MHK94] M. J. Mctiffin, A. P. Hulbert, T. J. Ketseoglou, et al., "Mobile Access to an ATM Network Using a CDMA Air Interface," *IEEE J. Select. Areas Comm.*, vol. 12, no. 5, pp. 900-908, June 1994.

[MFM96] A. S. Mahmoud, D. D. Falconer, and S. A. Mahmoud, "A Multiple Access Scheme for Wireless Access to a Broadband ATM LAN Based on Polling and Sectored Antennas," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 596-608, May 1996.

[Mil95] Scott Miller, "A Survey of Various Spreading Codes for DS/SS," *MS Project at The Mobile and Portable Radio Research Group at Virginia Tech*, Summer 1995.

[MoL89] R. K. Morrow, Jr. and J. S. Lehnert, "Bit-to-Bit Error Dependence in Slotted DS/SSMA Packet Systems With Random Signature Sequences," *IEEE Transactions on Communications*, vol. 37, no. 10, pp. 1052-1061, Oct. 1989.

[Mos96] Shimon Moshavi, "Multiuser Detection for DS-CDMA Communications", *IEEE Communications Magazine*, pp. 124-136, October 1996.

[NBK00] S. Nanda, K. Balachandran, and S. Kumar, "Adaptation Techniques in Wireless Packet Data Services," *IEEE Communications Magazine*, vol. 38, no. 1, pp. 54-64, January 2000.

[NGT91] S. Nanda, D. J. Goodman, and U. Timor, "Performance of PRMA: A Packet Voice Protocol for Cellular Systems," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 584-598, Aug. 1991.

[OjP98] Tero Ojanpera and Ramjee Prasad, “An Overview of Third-Generation Wireless Personal Communications: A European Perspective,” IEEE Personal Communications, pp. 59-65, December 1998.

[PaP95].K. Pahlavan, T.H. Probert, and M.E. Chase, “Trends in Local Wireless Networks,” *IEEE Communication Magazine*, Vol. 33, No. 3, pp. 88-95, March 1995.

[PaP97]N. Passas et al., “Quality-of-Service-Oriented Medium Access Control for Wireless ATM Networks,” *IEEE Commun. Mag.*, vol. 35, no. 11, Nov. 1997.

[PaZ97] K. Pahlavan, A Zahedi, and P. Krishnamurthy, “Wideband Local Access: Wireless LAN and Wireless ATM,” *IEEE Commun. Mag.*, vol. 35, no. 11, Nov. 1997.

[PaM98] N. Passas, L. Merakos *et al.*, “MAC Protocol and Traffic Scheduling for Wireless ATM Networks,” ACM Mobile Networks and Applications, vol. 3, no. 3, pp. 275-292, Sept. 1998.

[Pro95] J. G. Proakis, *Digital Communications.* New York: McGraw-Hill, 1995.

[PSM82] R. L. Pickholtz, D. L. Schilling and L. B. Milstein, “Theory of Spread-Spectrum Communications – a Tutorial,” *IEEE Transactions on Communications*, vol. 30, no. 5, pp. 855-884, May 1982.

[Pur77] M. B. Pursley, “Performance Evaluation of Phase-Coded Spread-Spectrum Multiple-Access Communications – Part I: System Analysis,” *IEEE Transactions on Communications*, vol. 25, no. 8, pp. 795-799, Aug. 1977.

[QiL96] X. Qiu and V. O. Li, “Dynamic Reservation Multiple Access (DRMA): A new Multiple Access Scheme for Personal Communications System (PCS)," *ACM Wireless Networks Journal*, vol. 2, pp. 86-94, August 1996.

[Rap96] T.S. Rappaport, *Wireless Communications – Principles and Practice*, Prentice-Hall Inc., NJ, 1996.

[RaW94] D. Raychaudhuri and N. Wilson, "ATM-Based Transport Architecture for Multiservices Wireless Personal Communication Networks," *IEEE Journal On Selected Areas In Communications*, vol 12, no 8, pp 1401–1413, Oct. 1994.

[Ray96] D. Raychaudhuri, "Wireless ATM Networks: Architecture, System Design and Prototyping," *IEEE Personal Communications*, vol. 3, pp. 42-49, Aug. 1996.

[Sch82] R. A. Scholtz, "The Origins of Spread-Spectrum Communications," *IEEE Transactions on Communications*, vol. 30, no. 5, pp. 822-854, May 1982.

[SiJ95] K. –Y. Siu and R. Jain, "A Brief Overview of ATM: Protocol Layers, LAN Emulation, and Traffic Management," *Computer Communications Review*, vol. 25, pp6-20, April 1995.

[SiS98] C. Sinner and R. Sigle, "Toward Wireless Multimedia Communications. Current Standards and Future Directions," *International Journal of Wireless Information Networks*, vol. 5, No. 1, pp. 61-73, 1998.

[Siu94] E. N. Singer, *Land Mobile Radio Systems*, Second Edition, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1994.

[SrW99] K. Sriram and Y.T. Wang, "Voice over ATM using AAL2 and Bit Dropping Performance and Call Admission Control," *IEEE J. Select. Areas Comm.*, vol. 17, no. 1, Jan. 1999, pp. 18-28.

[ThG97] M. Thomas and D. J. Goodman, "Multi-Rate PRMA: A Time Division Protocol for Adjustable Bit-Rate Sources," *Proceedings VTC'97*, May 1997.

[VRF99] A. Valko, A. Racz, and G. Fodor, "Voice QoS in Third-Generation Mobile Systems," *IEEE J. Select. Areas Comm.*, vol. 17, no. 1, pp. 109-123, Jan. 1999.

[Web98] W. Webb, *Introduction to Wireless Local Loop*, Artech House, Inc., Norwood, MA, 1998.

[WGJ93] N. D. Wilson, R. Ganesh, K. Joseph, and D. Raychaudhuri, "Packet CDMA versus Dynamic TDMA for Multiple Access in an Integrated Voice/Data PCN," *IEEE J. Select. Areas Commun.*, vol. 11, no. 6, pp. 870-883, 1993.

[WHT97] G. Wu, H. Harada, K. Taira, and Y. Hase, “An Integrated Transmission Protocol for Broadband Mobile Multimedia Communication Systems,” *Proceedings VTC'97*, pp. 1346-1350, June 1997.

[Wic95] Stephan B. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1995.

[WiM99] L. Wilhelmsson and L. B. Milstein, “On the Effect of Imperfect Interleaving for the Gilbert-Elliott Channel,” *IEEE Transactions on Communications*, vol. 47, no. 5, pp. 681-688, May 1999.

[WoL99] Tan F. Wong and Tat M. Lok, “Spreading Sequence Adaptation in Multicode CDMA Systems,” *IEEE International Conference on Communications*, vol. 3, pp. 1375-1379, June 2000.

[ZAB00] M. Zeng, A. Annamalai, and Vijay K. Bhargava, “Harmonization of Global Third-Generation Mobile Systems,” *IEEE Communications Magazine*, pp. 94-104, December 2000.

Appendix A: Simulation Models

This appendix describes the simulation models in OPNET Modeler used for this research effort. Discussion of the simulation tool is presented in Section A.1. The WATM network model is described in Section A.2. The node model for the MUs is depicted Section A.3. The process models used for the MUs are explained in Section A.4. BS node model is described in Section A.5. Section A.6 presents the process models that are employed in the BS. The antenna, transmitter and receiver characteristics are listed in Section A.7. The packet formats used in the simulations are summarized in Section A.8.

A.1 Simulation Tool

OPNET Modeler, version 6.0.L, OPNET Technologies, Inc. [OPN99], was chosen for this research effort in part by its established status as one of the leading products simulating communication networks. There are three levels of abstractions that describe a network with varying levels of detail. At the highest point of the hierarchy is the network domain. Here, the geometrical position of the nodes as well as their connection method is determined. Network objects are instantiated with their attributes and a node model. In this way, the same node model can be used with many objects with the same behavior, which follows the basic concept of object-oriented programming. The next level down, the node model, consists of the process modules and packet streams. Process modules are instantiated by their attributes and process model, once again enabling the object-oriented approach to programming. At the bottom of the level of abstraction is the process model. Here, the detailed behaviors and contents of the packets are determined by using state transition diagrams. In the state transition diagrams, there

are two kinds of states; the forced states, the lightly colored states (green), and the unforced states, the dark colored states (red). The forced states are states that process their executives, written in C/C++ language, and then transitions out to another state satisfying the transition conditions. The unforced state is a state where the top half portion of the executives are executed and the simulation control is relinquished back to the OPNET Simulation Kernel.

OPNET is an event driven simulator that efficiently describes the behavior of the system, forgoing idle periods that are of less interest or significance. The Simulation Kernel keeps a table of all the events that are scheduled to happen at a particular time. The simulation time progresses until the nearest event time coincides with the simulation time, in which case the event scheduled is called an "interrupt." These events and interrupts are the driving force behind the event driven OPNET simulator.

Simulation factors can be conveniently altered at the time of the simulation run by using corresponding simulation attributes. The user is allowed to collect statistics from the simulation runs by specifying probes in the simulator. A special interrupt at the end of the simulation can be generated so that the resulting global statistics collected from a simulation run can be recorded.

A.2 Network Model

The network model, as presented in Figure A.1, gives the overall structure of the WATM network studied. Shown is an infrastructure system with a BS at the center of the network and 100 MUs that are within the network cell. Among the 100 MUs, the MUs that are active are decided according to the simulation factors LOAD_METHOD and LOAD, as described in Seciton 5.5.

The network we are interested in is assumed to be within 100 meters in radius from the centrally positioned BS, as shown in Figure A.1. Since we assume perfect power control from the BS point of the, the actual locations of the MUs do not have any effect upon the overall performance of the network, and thus the distribution of the network nodes as depicted in Figure A.1 is valid without loss of generality. The absence of any wired connections implies that any communications between the BS and the MUs are done wirelessly.

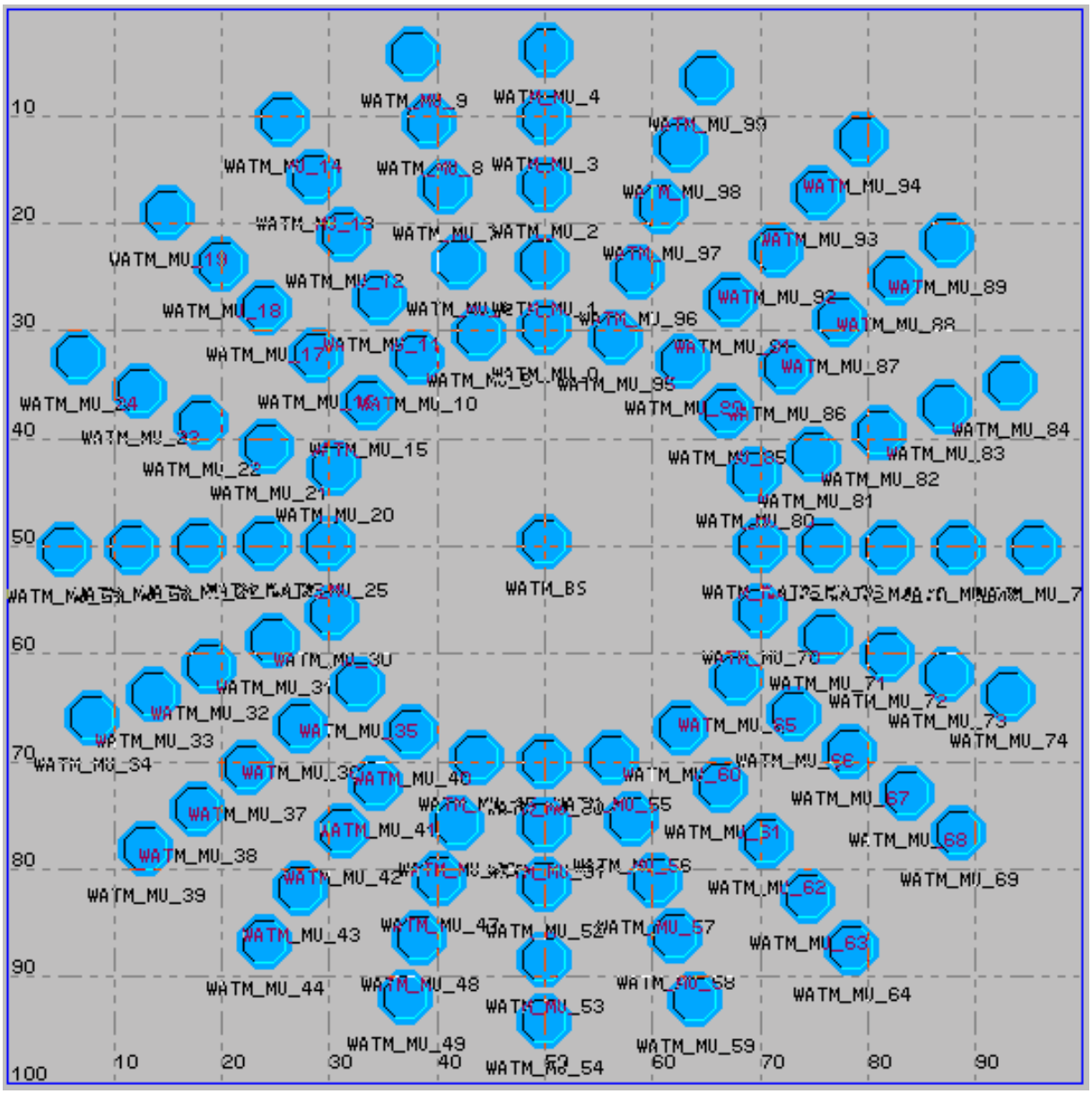


Figure A.1: Network Model

A.3 Node Model for the MUs

A single node model, WATM_MU, is used for the all the MUs in the network. Their characteristics are made unique by assigning different mobile Ids (MIDs).

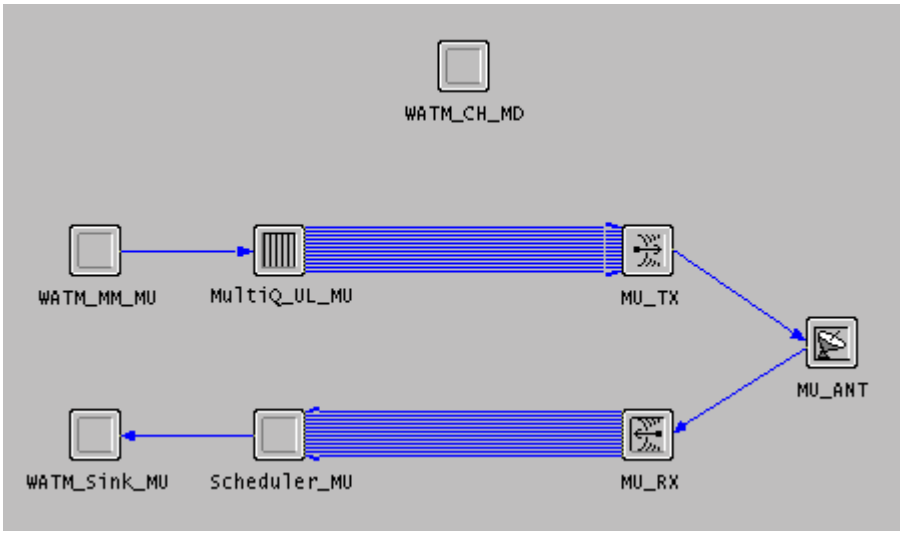


Figure A.2: MU Node Model

The MU node contains the traffic generating WATM_MM_MU process module, the multi-service MultiQ_UL_MU multi-queue module, the traffic scheduling Scheduler_MU module, the channel condition altering WATM_CH_MD module, the WATM_Sink_MU module, the MU_TX transmitter, the MU_RX receiver, and MU_ANT antenna as shown in Figure A.2. The packets are passed between the streams represented by the lines ending with pointed arrows. WATM_MM_MU process module generates all the uplink traffic according to the source generation distribution described in Section 4.6. All the traffic generated from the WATM_MM_MU process module are passed on to the MultiQ_UL_MU multi-queue module, where the traffic is stored in one of four sub-queues according to the type of traffic (CBR, rt-VBR, nrt-VBR, or ABR). When the MultiQ_UL_MU multi-queue module detects that packets are queued, it sends out an AR packet containing the number of packets that are waiting in the sub-queus to the BS using one of the 25 transmitter-receiver pair PN codes dedicated to the

access request phase by way of the MU_TX transmitter. After the BS successfully receives the AR packet, it sends a XPERM transmit permission packet back to the MU and it is received through the MU_RX receiver. The information containing the number of packets that are allowed to send is contained in this XPERM packet, and the during the next time frame, the data packets corresponding to the number permitted is sent out in parallel. WATM_CH_MD process module is used to dictate the condition of the channel and is not directly involved with traffic movement, and thus not connected physically. More will be explained of this module later in Section A.4.

A.4 Process Models for the MUs

The process models in this section dictate the creation, behavior, and change of contents of the packets in the MUs. The method used to handle events is to use state transition diagrams and programs dedicated to each state in the diagram.

A.4.1 WATM_MM_MU Process Model

All the traffic being generated by the MUs originates from WATM_MM_MU process model, as shown in Figure A.3. There are four types of traffic being generated; CBR, rt-VBR, nrt-VBR, and ABR. The distributions needed for the four types of traffic and the necessary state variables are initialized in INIT state. The number of users for each type of service within a MU is given through the simulation attribute, whose value is controllable at the simulation time (and not hard-coded within the program). The initial self-interrupts for the first “on” periods of the four traffic types (CBR, rt-VBR, nrt-VBR, ABR) are scheduled according to the distribution mentioned in Section 4.6. The ABR traffic is special in that it uses the “on” period portions of the toggle sequence to generate packets at the start of an “on” period in a burst rather than having individual packets being generated during the “on” periods as other traffic. After all the necessary initializations have been completed, the state transitions to the Waiting state, relinquishing the control back to the OPNET Simulation Kernel.

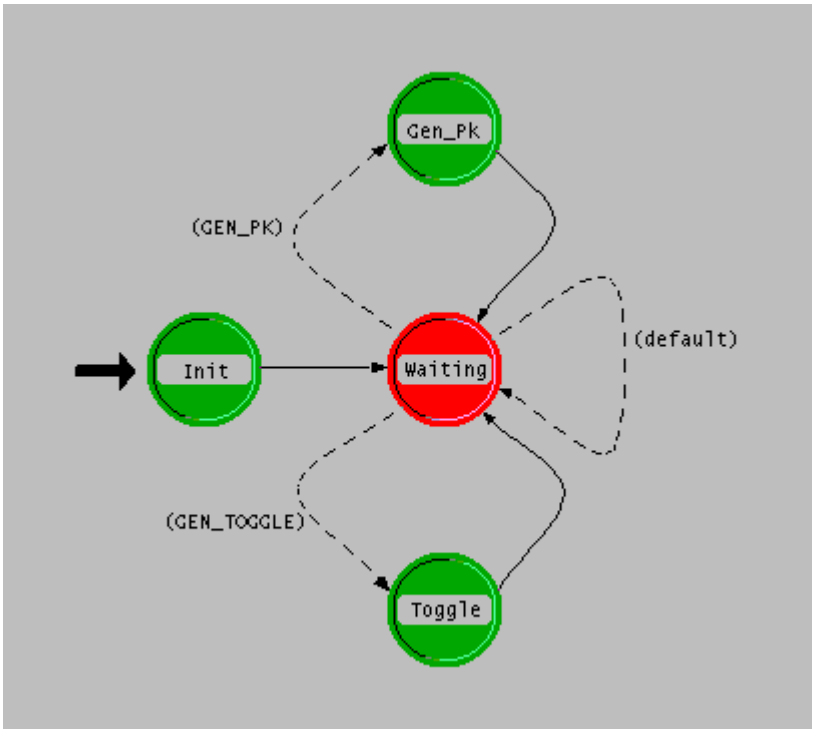


Figure A.3: WATM_MM_MU Process Model

In the Waiting state, the process remains idle until an interrupt occurs due to the scheduling done at the INIT state. The first interrupt will be from any of the four traffic types and it will indicate that an "on" period is entered and the state transitions to the Toggle state. At the Toggle state, immediate action is taken by specifying that an "on" period has been entered and scheduling for another self-interrupt for toggling from "on" to the "off" period is done. Also, a shorter time period in which the next packet will be generated is scheduled using another self-interrupt, called packet interrupt. The packet interrupt is effective as long as the source of the traffic is in the "on" period. Having completed the scheduling of both the toggling interrupt and packet interrupt, the state transitions back to the Waiting state and the control of the simulation is relinquished back to the OPNET Simulation Kernel. At this point, there could be two kinds of interrupt that can happen; one is the GEN_PK interrupt that specifies that a packet is to be generated, and the other is GEN_TOGGLE, which indicates that the toggling between "on" and "off" period is to happen. If the GEN_PK interrupt is received, a packet corresponding to the type of service it originated from is generated and another GEN_PK interrupt is scheduled.

If the GEN_TOGGLE interrupt is detected, the toggling between "on" and "off" periods occurs, and the next GEN_TOGGLE interrupt is scheduled. In either interrupts, the state transitions back to the Waiting state before relinquishing simulation control to the OPNET Simulation Kernel. This process of generating new packets and toggling between "on" and "off" periods is repeated until the end of simulation. Up till now, a general description of the state diagram has been given, but there are some differences among the four types of traffic services. For a CBR traffic, there is one source that toggles between "on" and "off" periods. However, for a (rt- or nrt-) VBR traffic, there are 10 sources that are super-positioned to represent one such traffic. Other than that, each source is tracked and the two interrupts that are generated follow the same routines. For an ABR traffic, all the traffic is generate at the start of the "on" period following a geometric distribution, as described in Section 4.6.2. Also, another "on" period is scheduled rather than toggling between "on" and "off" periods.

Table A.1 describes the simulation parameters used in this process models. The parameters control the characteristics of the network or are related to the collection of statistics used for this research effort.

Table A.1: Simulation Parameters for WATM_MM_MU Process Model

Parameter Name	Description	Data Type
CBR_Exp_Off_Mean	Mean "off" time for CBR	Double
CBR_Exp_On_Mean	Mean "on" time for CBR	Double
Num_On_Off_Sources	Number of on-off sources used for (rt and nrt) VBR	Integer
VBR_Exp_Off_Mean []	Mean "off" time for each source of (rt and nrt) VBR	Double
VBR_Exp_On_Mean []	Mean "on" time for each source of (rt and nrt) VBR	Double
ABR_Exp_Mean	Mean interarrival time for ABR bursts	Double
ABR_Burst_Par	Geometric distribution parameter for the number of packets in a burst for ABR	Double
CBR_Enable	Total number of CBR users in the MU	Integer
RT_VBR_Enable	Total number of rt-VBR users in the MU	Integer
NRT_VBR_Enable	Total number of nrt-VBR users in the MU	Integer
ABR_Enable	Total number of ABR users in the MU	Integer

A.4.2 MultQ_UL_MU Process Model

The MultiQ_UL_MU process model takes care of several functions that can be categorized into three groups; the uplink queuing group, the uplink output group, and the uplink input group. At the start of the simulation, all the state variables and simulation parameters related to the MultiQ_UL_MU process model are initialized at the Init state and state transitions to the Branch state, where the control of the simulation is relinquished back to the OPNET Simulation Kernel. All steady state simulation now occurs from the Branch state.

The first group consist only of the Ins_Tail state. This state is entered when a packet generated from the WATM_MM_MU process model causes an ARRIVAL interrupt. Once in the Ins_Tail state, the packet is sorted according to the type of packet (CBR, rt-VBR, nrt-VBR, and ABR) and stored in one of the four subqueues corresponding to the type of packet. The state transitions back to the Branch state and the simulation control is relinquished back to the OPNET Simulation Kernel.

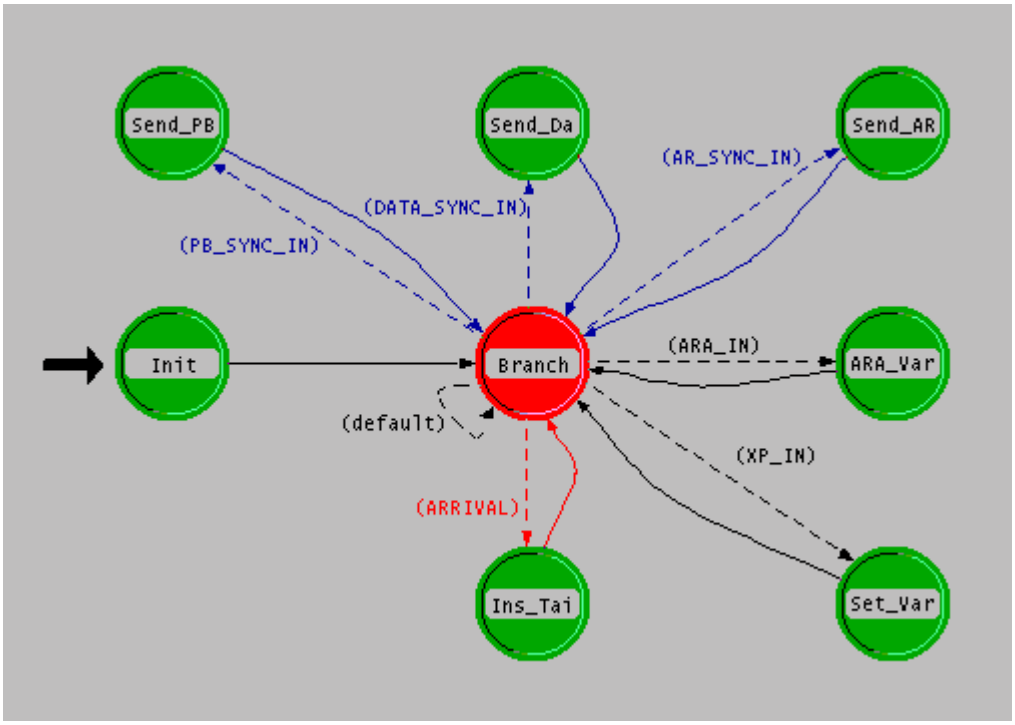


Figure A.4: MultiQ_UL_MU Process Model

The second group of states is the uplink output group, which consists of Send_AR, Send_Data, and Send_PB states. The each of theses states is entered at the corresponding synchronous times that are generated by the Scheduler_MU, which is to be described below. When an AR_SYNC_IN interrupt is received, transition is made to the Send_AR state, where it checks whether there are packets queued in one of its multi-subqueues. If there are packets waiting, the minimum of the number of packets waiting in the queue and the maximum number of parallel streams available is recorded in the AR packet and sent out to the BS. The state transitions back to the Branch state, where the simulation control is relinquished back to the OPNET Simulation Kernel. When an PB_SYNC_IN interrupt is detected, the Send_PB state is entered, where its state executives determines whether the transmit permission, XPERM, has been received from the BS and there will still be packets waiting in the queue after sending the Data packet(s). If there are no Data packets to be sent, it follows that there will be no PB_req packets that will be sent. Else, if there are packets that will be waiting in the queue after sending of the Data packet(s), then the minimum of that number of packets foreseen to be remaining and the maximum number of parallel streams available per MU will be recorded in the MSS field and the number of real-time packets among those will be recorded in the MSSRT field of the PB_req packet and sent out to the BS. If there are no packets foreseen to be remaining after the Data packet(s) are sent out for the current time frame, then a 0 will be recorded to the MSS and MSSRT fields of the PB_req packet. It is noted that the number of PB packets that are able to be sent out follow the number of parallel streams that are given permission to send out its DATA packet. After the PB_req packet is sent, the state transitions back to the Branch state and the control of the simulation is relinquished to the OPNET Simulation Kernel. Finally, the last state of the uplink output group, Send_Data state, is entered when DATA_SYNC_IN interrupt is detected. The Send_Data state executives will check whether transmit permission, XPERM, from the BS was received. If no permission was given, an immediate exit from the Send_Data state occurs and the state transitions back to the Branch state. However, if there are packets to send and XPERM was given, the number of real-time data packets according to the MSSART field, and in total the number of data packets according to the MSSA field of the XPERM packet received from the BS will be sent. After the sending of the Data packet(s), the state transitions

back to the Branch state and the simulation control is relinquished back to the OPNET Simulation Kernel.

The third group of states is the uplink input group, which consists of ARA_Var and Set_Var states. These states are entered upon receiving BS originated information by way of the Scheduler process module. The ARA_Var is entered when ARA_IN interrupt is generated when a control packet from the Scheduler process module is received. At this state, the acknowledgment of the AR request sent from the MU at the previous time frame is received. The acknowledgment information is recorded in a state variable inside the ARA_Var state. The state transitions back to the Branch state and simulation control is relinquished. The Set_Var state is entered when the information pertaining to the XPERM packet received from the Scheduler_MU is received via XP_IN interrupt. The Set_Var process will store in its state variables the BS originated information about the number of packets that will be allowed to be sent (MSSA), the number of real-time packets that will be allowed to be sent (MSSRTA), the RL to be used, the M vector describing the acknowledgment of the data last sent from the MU for each stream. This information will be used during the next cycle of the uplink time frame.

The simulation parameters used for MultiQ_UL_MU process model is listed in Table A.2. The last four parameters are used exclusively for the validating the APMS protocol against DQRUMA simulation results presented in [LKZ96], as was described in Section 5.8.

Table A.2: Simulation Parameters for MultiQ_UL_MU Process Model

Parameter Name	Description	Data Type
Rake_In_Used	Distinguishes between APMS and DQRUMA	Integer
Max_Retrans	Sets the number of maximum retransmission allowed	Integer
Max_RT_Tolerance	The time constraint imposed on real-time traffic	Double
Max_NRT_Tolerance	The time constraint imposed on non-real-time traffic	Double
Class_Traffic	Class of traffic used for validation against DQRUMA	Integer
Num_Pks_Per_Slot	Number of packets per slot used for validation against DQRUMA	Integer
Max_Backoff	Maximum number of backoff	Integer
Backoff_Moder	The backoff model used (harmonic or uniform dist)	Integer

A.4.3 Scheduler_MU Process Model

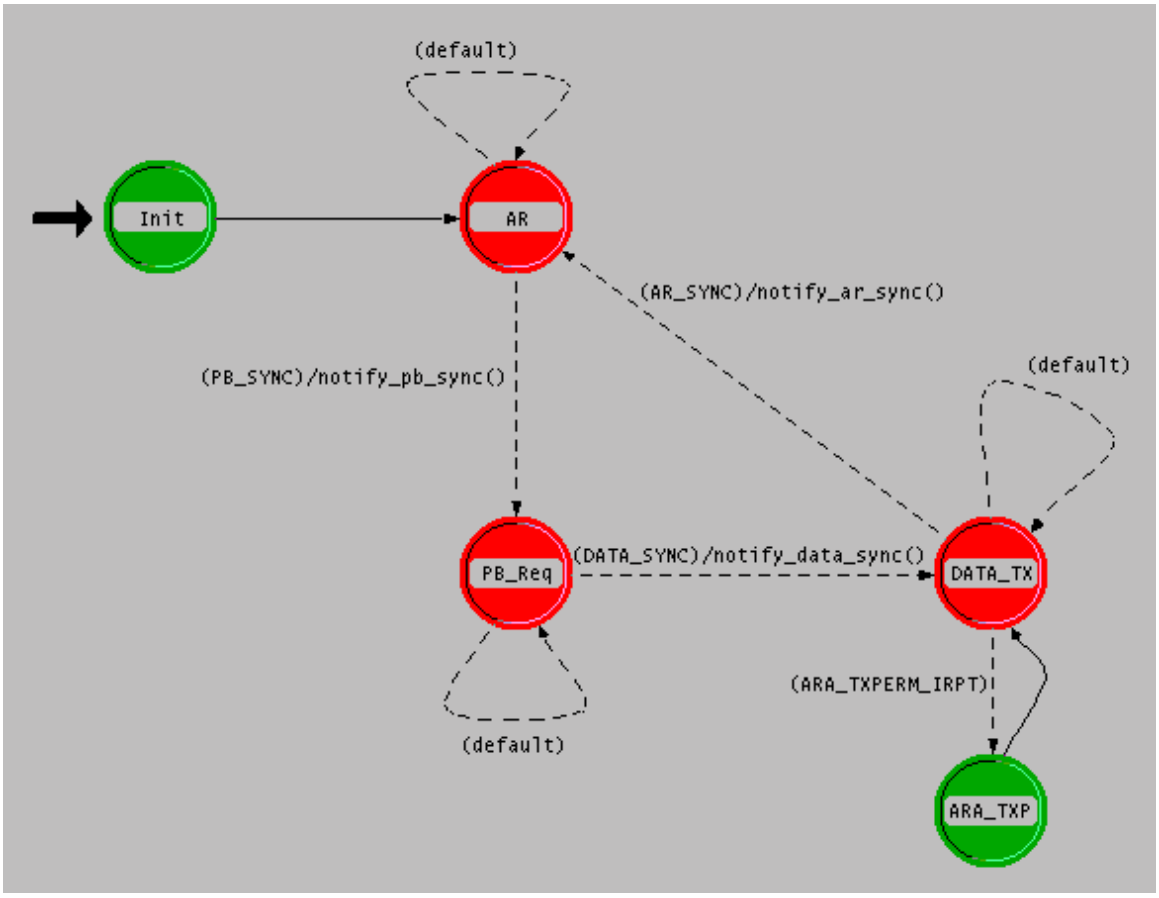


Figure A.5: Scheduler_MU Process Model

There are two main functions contained in the Scheduler_MU. One is the keeping the time synchrony in terms of the start of the AR phase, PB phase, and Data transmit phase of a time frame. The second is receiving the ARA and XPERM packet sent from the BS and relaying the information contained within the messages to the MultiQ_UL_MU process module.

At the start of the simulation, the Init state is entered, where the initialization of the state variables and duration of each of the mentioned phases is done. The first state entered from the Init state is the AR state, where preparation for the next time interrupt, which will signal the PB phase, is done. The simulation control is relinquished once this is done. The PB_SYNC interrupt is generated when the scheduled time duration for the AR phase is reached, signaling

the beginning of the PB phase by entering the PB_Req state. Here, the preparation for the next time interrupt, DATA_SYNC, which will signal the Data transmit phase, is done, and the simulation control is relinquished. Once the DATA_SYNC interrupt is detected, the DATA_TX state is entered, and a time interrupt is scheduled for the start of the next time frame, which is the next AR phase. At the DATA_TX state, there are two packet stream interrupts coming from the BS that could occur, before transitioning to the AR state. One is the ARA packet and the other is the XPERM packet. These two interrupts are taken care of in the ARA_TXPERM forced state. In the ARA_TXPERM forced state, the contents in the ARA packet or the XPERM packets are relayed to the MultiQ_UL_MU process module. The details of the contents have been discussed in Section A.4.2. Since the ARA_TXPERM state is a forced state, the state will transition back to the DATA_TX state before the simulation control is relinquished.

The simulation parameters used for Scheduler_MU process model is listed in Table A.3. Here, the lengths of the various uplink packets, along with data rate, is used in generating the appropriate synchronous times of each phases of a time frame.

Table A.3: Simulation Parameters for Scheduler_MU Process Model

Parameter Name	Description	Data Type
WATM_Len	The length of the “watm_cell” Data packet	Integer
AR_Len	The length of the “watm_ar” AR packet	Integer
PB_Len	The length of the “watm_pb” PB_Req packet	Integer
MBPS	The data rate per stream in mega bits per second	Double

A.4.4 WATM_CH_MD Process Model

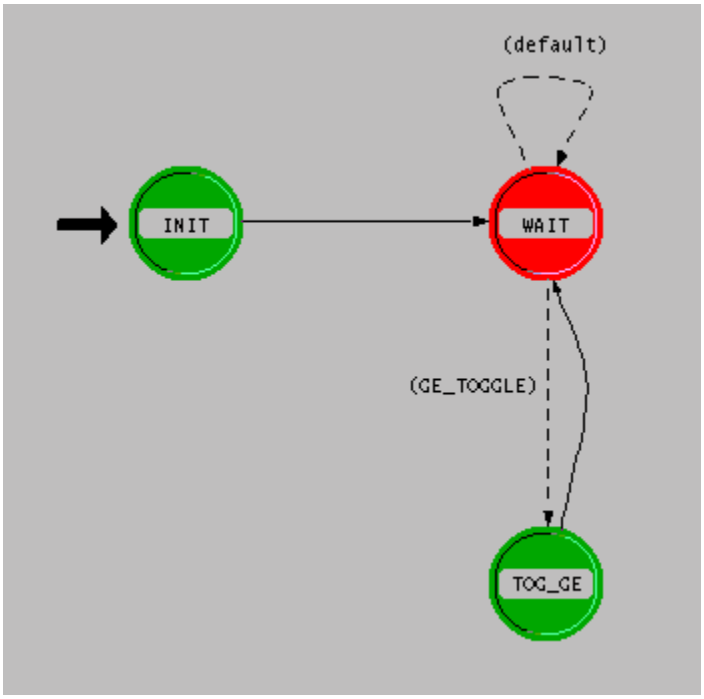


Figure A.6: WATM_CH_MD Process Model

The WATM_CH_MD process model differs from previous process models discussed in Section 4.4.1 through 4.4.3 in that this model does not directly handle packets. The main function allotted to WATM_CH_MD process model is determining whether the channel is in a "good" or "bad" period as dictated by the Gilbert-Elliott model discussed in Section 5.3. The "good" and "bad" periods toggle with exponentially distributed durations [Ell63]. At the start of the simulation, the distribution properties of the "good" and "bad" periods are initialized along with the supporting state and global variables need for this process model at the Init state and transitions to the WAIT unforced state, where the simulation control is relinquished. GE_TOGGLE interrupt indicates that there is a transition (toggling) between the "good" and "bad" periods and enters the TOG_GE forced state. Here, the global variable, g_or_b[mid], indicating the channel mode is set to the appropriate value. The "mid" is the mobile ID (MID) of the MU in question. This global variable is used to set the condition of the channel linking the

corresponding to the MU unit and the BS to either “good” or “bad” [Ell63]. It is followed by the scheduling of the next interrupt time for the toggling to occur. Once the executives in the TOG_GE forced state have been completed, the state transitions back to the WAIT state and simulation control is relinquished.

The simulation parameters used WATM_CH_MD process model is listed in Table A.4. All four parameters are involved in determining the durations of the “good” and “bad” periods for a channel between a MU and the BS.

Table A.4: Simulation Parameters for WATM_CH_MD Process Model

Parameter Name	Description	Data Type
Small_G	Probability for “bad” to “good” transition	Integer
Small_B	Probability for “good” to “bad” transition	Integer
Frame_Len	The length of a time frame	Integer
MBPS	The data rate per stream in mega bits per second	Double

A.4.5 WATM_Sink_MU Process Model

The only purpose of the WATM_Sink_MU process model is to destroy all the packets coming from the BS after the contents of them have been retrieved in order to restore memory for the OPNET Modeler simulator.

A.5 Node Model for the BS

The WATM_BS node model defines the all the functionalities of the BS. Figures A.7 and A.8 only show the first 10 sets of modules corresponding to the 10 MUs due to space limitation, but there are up to 100 sets of modules in the whole WATM_BS node model. The WATM_BS node model encompasses Proc_UL_BS_x modules based on the Proc_UL_BS process model, WATM_Sink_BS_x modules based on the WATM_Sink_BS model, the WATM_MM_BS module, WATM_DL_ARA_TXPERM_x modules based on the WATM_DL_ARA_TXPERM process model, Stat_Collect module, BS_TX transmitter, BS_RX receiver, and BS_ANT_0 and BS_ANT_1 antennas for the base station. It is noted that the lowercase “x” at the end of the modules represents one of the 100 modules corresponding to the different MUs in the system. The actual model names that those modules use are without the

lowercase “x” at the end. The uplink data from the MUs are received by one of the Proc_UL_BS_x modules using the Proc_UL_BS process model with the matching PN code. The Proc_UL_BS process model determines the type of packet that has been sent from the MU and delivers the relevant information to the WATM_MM_BS process module for uplink traffic scheduling. This delivery process is done by using an ICI (Interface Control Information) feature attached to a remote interrupt scheduling command provided by OPNET Modeler. The packets are destroyed in the WATM_Sink_BS_x process modules. The central WATM_MM_BS process module gathers all the information coming from the Proc_UL_BS_x modules and maintains a central BS traffic control table to be used for determining which AR and PB_Req packets will be granted transmit permissions (XPERM), and if so how many packets will be allowed to be sent uplink.

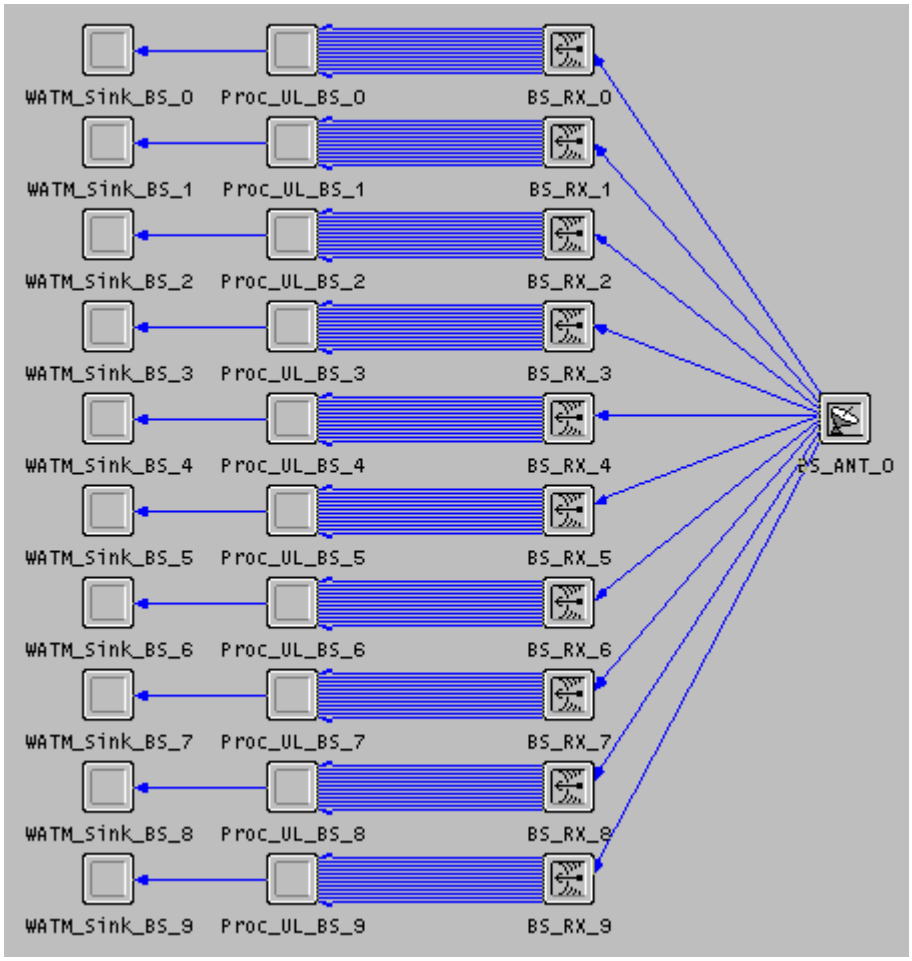


Figure A.7: BS Node Model (Part 1 of 2)

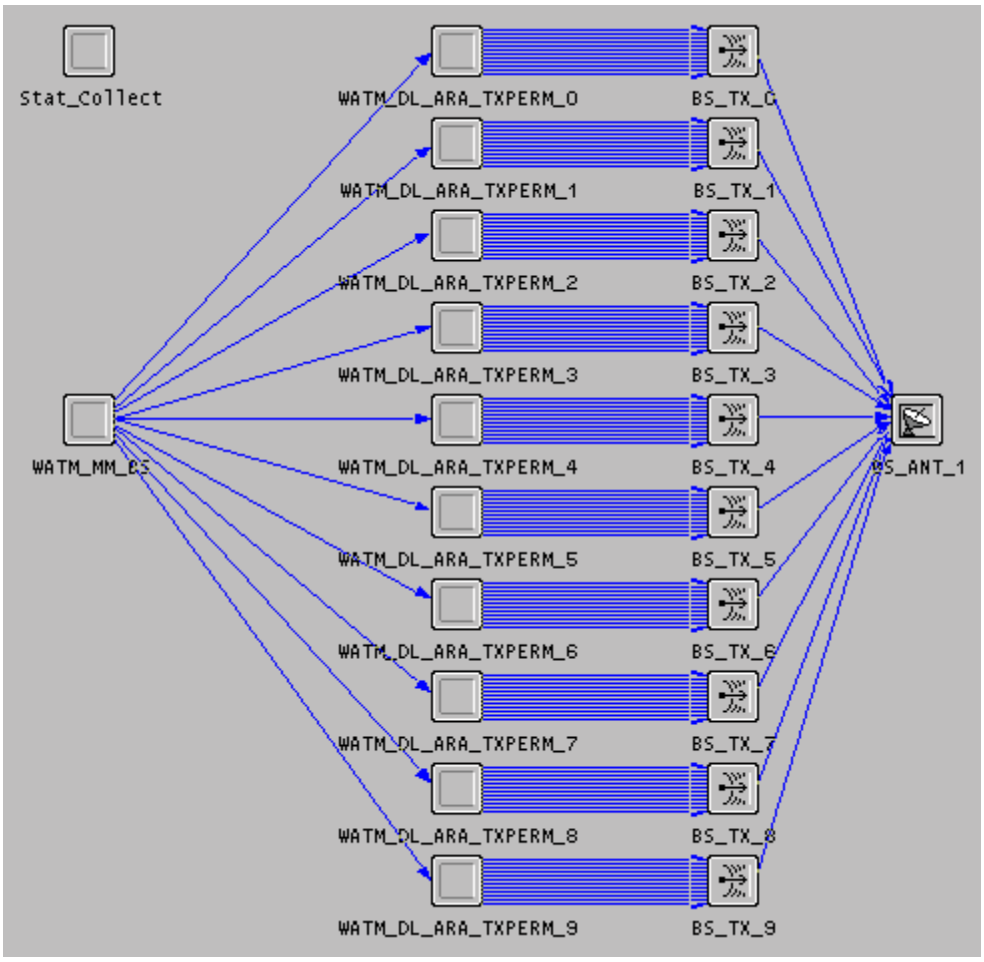


Figure A.8: BS Node Model (Part 2 of 2)

The ARA and XPERM packets will be sent back to the appropriate MUs via WATM_DL_ARA_TXPERM_x modules with the matching PN codes.

A.6 Process Models for the BS

This section encompasses the process models used for describing the creation, behavior, and change of contents of the packets in the BS. Once again, the method used to handle events is to use state transition diagrams and programs dedicated to each state in the diagram.

A.6.1 Proc_UL_BS Process Model

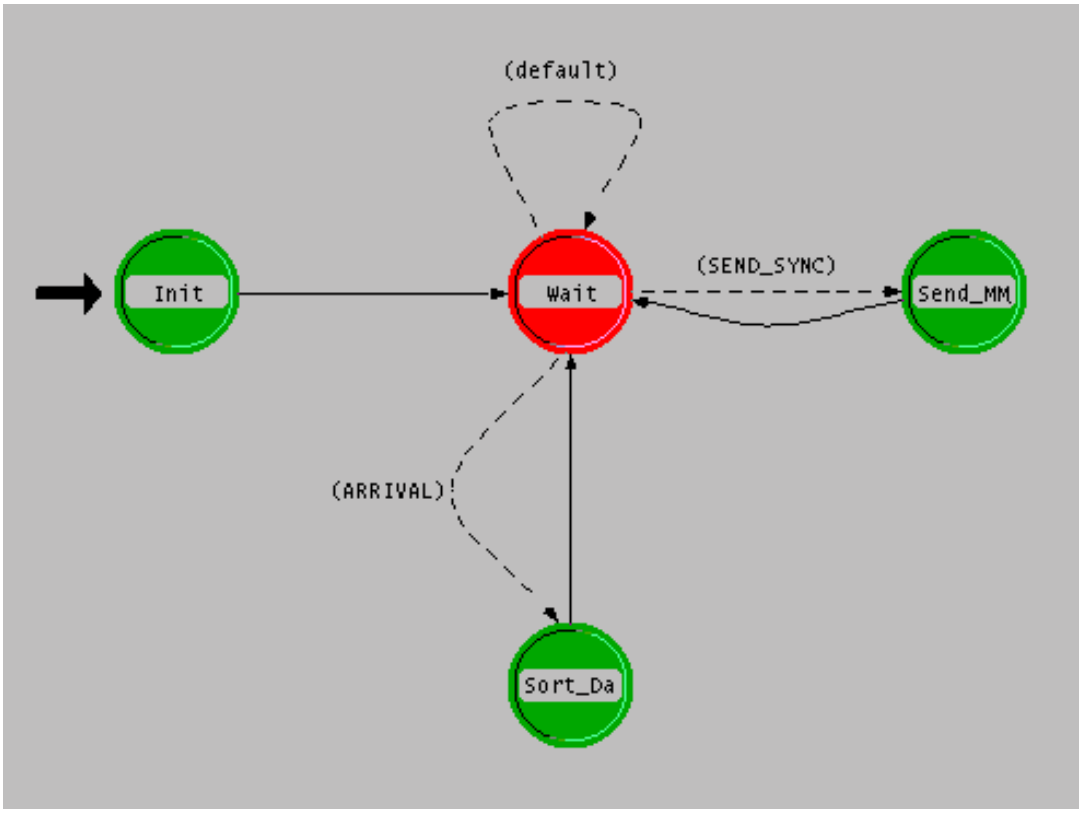


Figure A.9: Proc_UL_BS Process Model

The Proc_UL_BS process model enters the Init state at the start of the simulation, where the initialization of the state variables and ICI related initialization are done. The simulation then transitions to the Wait unforced state, where the simulation control is released to the OPNET Simulation Kernel. Figure A.9 depicts the Proc_UL_BS process model.

At the Wait state, there can be two types of interrupts. One is the ARRIVAL interrupt, which indicates the arrival of a packet from one of the MUs. In this case, the Sort_Data forced state is entered, where the type of the packet arriving is determined and appropriate action taken depending on the outcome. If the packet is determined to be a control packet (AR or PB_Req),

then the requested information is forwarded to the WATM_MM_BS process module using the ICI mechanism. If the received packet is a Data packet, then the reception of it is recorded in the "M" vector state variable that maintains the information concerning successfully transmitted packets per MUs based on the parallel stream index from which they came. After the information is processed from the Sort_Data state, the packet is forwarded to the WATM_Sink_BS_x Process Module, and the state transitions back to the Wait unforced state.

The second type of interrupt is the SEND_SYNC interrupt. This interrupt is generate for the purpose of sending out the "M" vector information once per time frame. This enables sending out the all the data reception information of a MU only once per time frame, eliminating unnecessary overhead that may be caused if interrupts were to be occur for each arriving data packet for a MU. The SEND_SYNC interrupt triggers the state to transition to the SEND_MM_BS forced state where all the data reception information of a MU per time frame is sent out to the WATM_MM_BS process module via the ICI mechanism.

The simulation parameters and their description for Proc_UL_BS process model is presented in Table A.5. Max_Dist is used to calculate the maximum propagation time in order to ensure proper handling of uplink and downlink communication sequence. The other two parameters are for handling validation mode presented in Section 5.8. This is needed here since DQRUMA protocol assumes no errors in the channel, and thus unlike in the APMS case, all DQRUMA packets must be accepted at the BS as valid.

Table A.5: Simulation Parameters for Proc_UL_BS Process Model

Parameter Name	Description	Data Type
Max_Dist	The maximum diameter of a WATM network cell	Double
Class_Traffic	Class of traffic used for validation against DQRUMA	Integer
Validate_Mode	Specifies whether or not in validation mode	Integer

A.6.2 WATM_Sink_BS Process Model

The sole purpose of the WATM_Sink_BS process model is to destroy the uplink packets that have been process by the Proc_UL_BS process model.

A.6.3 WATM_MM_BS Process Model

The WATM_MM_BS process model is the center point of the BS, where the all information regarding requests for access to the uplink channel from the MUs are gathered and processed. The main purpose of WATM_MM_BS process model is to generate downlink control packets to the MUs so that the uplink traffic could be regulated. Having a sophisticated central process module at the BS is important since it is the only place where all the information about the WATM system is known.

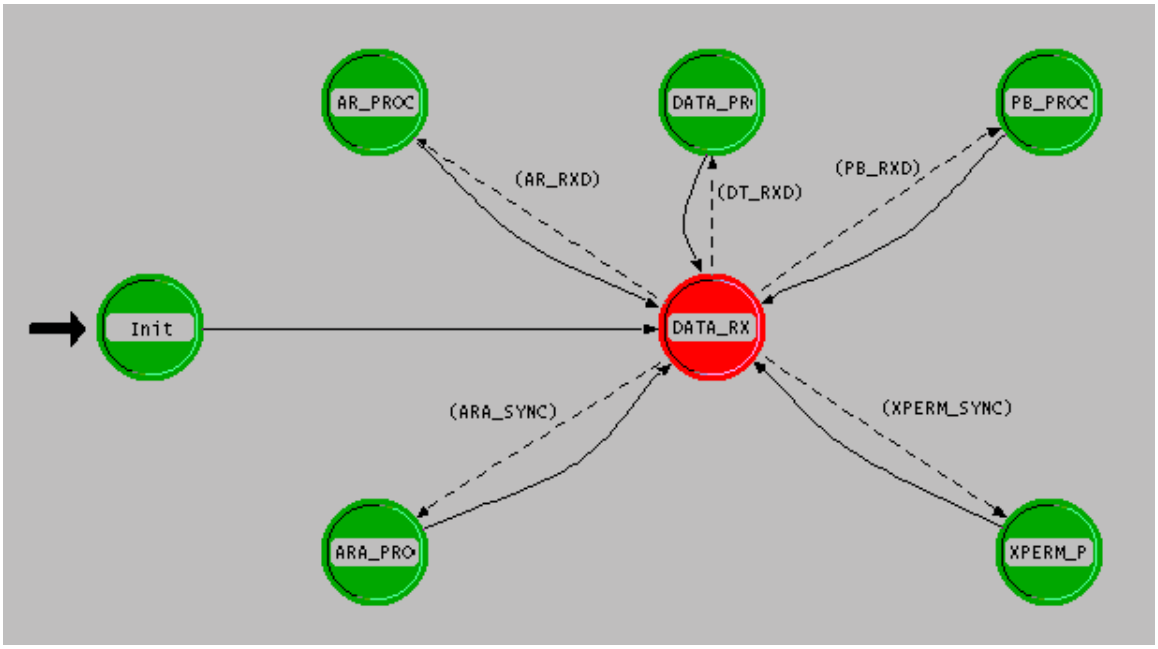


Figure A.10: WATM_MM_BS Process Model

At the start of the simulation, the Init state is entered where a central table containing information about all the MUs in the WATM system is prepared, and a timer is set to generate time interrupts for the start of the ARA phase and XPERM phase, along with the usual initialization of the supporting variables needed for the state executives. After the initializations, the state transitions to the DATA_RXD unforced state, where the simulation control is released

to the OPNET Simulation Kernel. There are six different types of interrupts that could reactivate this process model from the DATA_RXD unforced state.

The six interrupts can be categorized into two groups. The first group of interrupts originates from Proc_UL_BS_x modules, described in Section A.6.1. There are three types of interrupts for the first group; AR_RXD, PB_RXD, and DT_RXD. The AR_RXD interrupt occurs when the Proc_UL_BS_x modules send information related to the AR packets. The simulation proceeds to the AR_PROC forced state, where the access request information is stored in the input portion of the central table, and the state transitions backed to the DATA_RXD unforced state. The PB_RXD interrupt occurs when the Proc_UL_BS_x modules send information related to the PB_Req packets. The simulation proceeds to the PB_PROC forced state, where the access request information is stored in the input portion of the central table, and the state transitions backed to the DATA_RXD unforced state. The DT_RXD interrupt occurs when the Proc_UL_BS_x modules send information related to the Data packets. The simulation proceeds to the DT_PROC forced state, where the data validity information is stored in the input portion of the central table, and the state transitions backed to the DATA_RXD unforced state.

The interrupts in the second group are the ARA_SYNC and XPERM_SYNC. The ARA_SYNC is triggered when it is time for the BS to send out ARA packets to the MUs that have sent their AR packets without experiencing any collisions. This process is taken care of at the ARA_PROC forced state before returning back to the DATA_RXD unforced state. The XPERM_SYNC interrupt occurs at the start of XPERM phase of the downlink. XPERM packets are generated for each MU that has requested access for the uplink channel through either AR packets or PB_Req packets. Not all MUs may be given permission to send any of their data, if there is more demand for the channels than can be allowed due to excessive interference concerns. These calculations are done in the XPERM_PROC forced state, based on all the information collected in the BS central table. After all XPERM packets required to be sent back is taken care of, the state transitions back to the DATA_RXD unforced state and relinquishes the simulation control to wait in idle mode till the next time frame cycle.

The simulation parameters for WATM_MM_BS Process model is presented in Table A.6. Max_MSS specifies the maximum number of parallel substreams that are allowed in the system. PN_Len specifies the processing gain used for the multi-code CDMA air-interface. The

WATM_Len is used for estimating the time transmission time of a WATM Data packet. Tot_Num_MUs indicates the total number of MUs that are active. MBPS parameter is needed for calculating the transmission time the packets. Rake_In_Used is used to distinguish between the conventional DQRUMA and APMS protocols. The rake-in level (RL) used during the "bad" periods for the APMS protocol is specified in Max_Rake_In parameter. Release_Cnt determines the number of time frames that should pass before exiting a rake-in mode. The non-zero value in Ch_Est_Md indicates error packets counting channel estimation is to be used. A zero value implies that the simulation follows a perfect channel state knowledge assumption. The last three parameters are related to the validation mode described in Section 5.8.

Table A.6: Simulation Parameters for WATM_MM_BS Process Model

Parameter Name	Description	Data Type
Max_MSS	Maximum number of multiple substreams allowd	Integer
PN_Len	Length of the PN codes	Integer
WATM_Len	Length of the WATM Data packet	Integer
Tot_Num_MUs	Total number of mobile units that are active	Integer
MBPS	The data rate per stream in mega bits per second	Double
Rake_In_Used	Distinguishes between APMS and DQRUMA	Integer
Max_Rake_In	Maximum level of rake-in used	Integer
Release_Cnt	Count of time frames for before RELEASE is done from a rake-in mode	Integer
Ch_Est_Md	Specifies whether or not in channel estimation mode	
Validate_Mode	Specifies whether or not in validation mode	Integer
Class_Traffic	Class of traffic used for validation against DQRUMA	Integer
Num_Pks_Per_Slot	Number of packets per slot used for validation against DQRUMA	Integer

A.6.4 WATM_DL_ARA_TXPERM Process Model

The main function of the WATM_DL_ARA_TXPERM Process Model, as shown in Figure A.11, is to relay the packet coming in from the WATM_MM_BS module to the appropriate substreams of the BS_TX transmitter. For this research effort, only control packets (ARA and XPERM) are sent downlink.

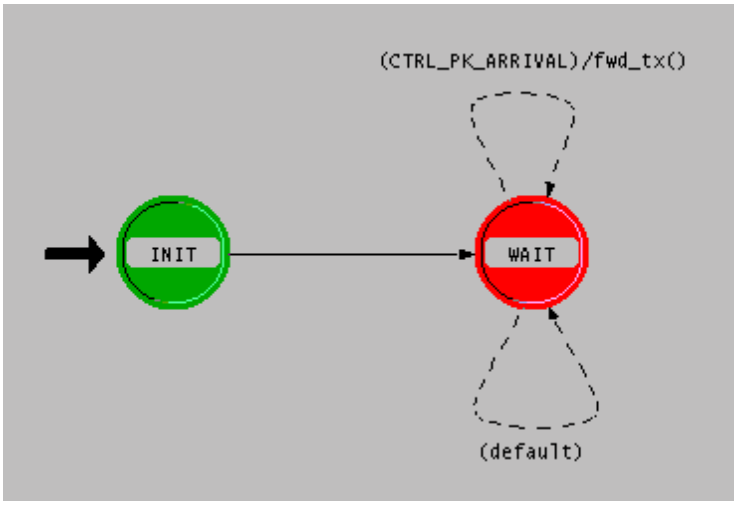


Figure A.11: WATM_DL_ARA_TXPERM Process Model

A.6.5 Stat_Collect Process Model

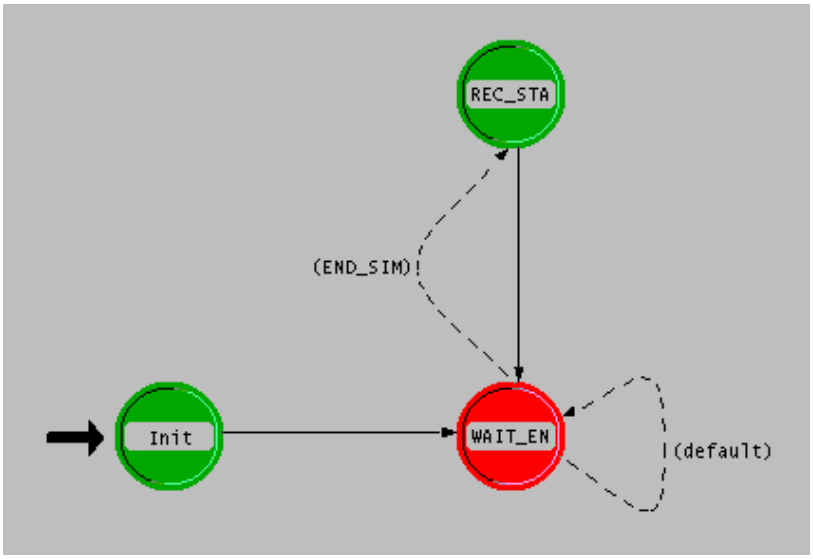


Figure A.12: Stat_Collect Process Model

The Stat_Collect process model, as described in Figure A.12, is not directly involved in processing packets, but is used to initialize the global variables used in the whole system and to collect the scalar data information generated per simulation run, such as throughput, delay, the number of packets dropped due to time constraints, and the number of packets exceeding maximum retransmission limit.

All of the parameters specified in Stat_Collect process model are used for collecting statistics per simulation run. Their names and descriptions are given in Table A.7. The last four parameters are used exclusively for validation purposes, as described in Section 5.8.

Table A.7: Simulation Parameters for Stat_Collect Process Model

Parameter Name	Description	Data Type
Throughput _CBR	Relative throughput for CBR traffic	Double
Throughput _rt_VBR	Relative throughput for rt-VBR traffic	Double
Throughput _nrt_VBR	Relative throughput for CBR traffic	Double
Throughput _ABR	Relative throughput for ABR traffic	Double
Dropped_Pks_CBR	Percentage of packets dropped for CBR traffic	Double
Dropped_Pks_rt_VBR	Percentage of packets dropped for rt-VBR traffic	Double
Dropped_Pks_nrt_VBR	Percentage of packets dropped for CBR traffic	Double
Dropped_Pks_ABR	Percentage of packets dropped for ABR traffic	Double
Delay_CBR	Average uplink delay for CBR traffic	Double
Delay_rt_VBR	Average uplink delay for rt-VBR traffic	Double
Delay_nrt_VBR	Average uplink delay for CBR traffic	Double
Delay_ABR	Average uplink delay for ABR traffic	Double
Retran_Exceeded_nrt_VBR	Percentage of nrt-VBR packets exceeding maximum retransmission limit	Double
Retran_Exceeded_ABR	Percentage of ABR packets exceeding maximum retransmission limit	Double
Slot_Cnt_Class1	Average number of slots consumed for Class 1 traffic in validation mode	Double
Slot_Cnt_Class2	Average number of slots consumed for Class 2 traffic in validation mode	Double
Slot_Cnt_Class3_1	Average number of slots consumed for Class 1 traffic in mixed traffic(Class3) in validation mode	Double
Delay_RT_VBR	Average number of slots consumed for Class 1 traffic in mixed traffic(Class3) in validation mode	Double
Delay_NRT_VBR	Average uplink delay for CBR traffic	Double
Delay_ABR	Average uplink delay for ABR traffic	Double

A.7 Antenna, Transmitter and Receiver

This section describes the various parameters associated with the physical models.

A.7.1 Antenna

All the simulations assume a standard OPNET provided omnidirectional isotropic antenna.

A.7.2 Transmitter

A radio transmitter with 12 channels is used. All channels are equal in characteristics except the spread-spectrum spreading code. All pipelines used are OPNET provided standard pipeline models.

The following are the default values.

1. Data rate – 1.185 Mbits/sec
2. Packet format – all formatted, unformatted
3. Bandwidth – 152 MHz
4. Minimum Frequency – 2 GHz
5. Transmit Power – 1 Watt
6. Modulation - BPSK

A.7.3 Receiver

A radio receiver with 12 channels is used. All channels are equal in characteristics except the spread-spectrum spreading code. All pipelines used are OPNET provided standard pipeline models, except the error correction (ecc) pipeline stage, as explained in Section A.7.4.

The following are the default values.

1 Data rate – 1.185 Mbits/sec

2 Packet format – all formatted, unformatted

3 Bandwidth – 152 MHz

4 Minimum Frequency – 2 GHz

5 Processing Gain – 21 dB

6 Modulation – BPSK

7 Error correction model - WATM_dra_ecc (correctable up to 3 error bits)

A.7.4 Error Correction Model – WATM_dra_ecc pipeline stage

The last pipeline stage is the WATM_dra_ecc error correction model. The Gilbert-Elliott bursty channel and the 3-bit error correction functionalities are implemented in this model.

The Gilbert-Elliott bursty channel is modeled based on the values presented in [WiM99]. A curve-fitting method is employed on the signal-to-noise-interference from the “good” mode to the “bad” mode. The bit error rate is derived from Equation 4.11. When a rake-in mode is employed, Equation 4.12 is used, as explained in Section 4.5.

Once a bit error rate is determined, the packet error rate is calculated assuming up to 3 bit errors in the WATM Data packets are correctable. The final validity of a WATM Data packet is decided based on the packet error rate.

A.8 Packet Formats

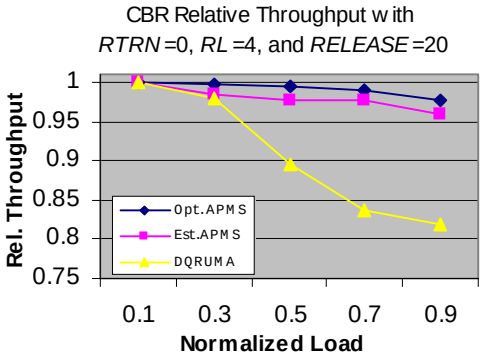
There are five packet formats used for this research; watm_cell, watm_ar, watm_pb, watm_ara and watm_xperm. The first three of them are for use in the uplink and the other two are for use in the downlink. Their formats and usages are presented in Table A.8.

Table A.8: Simulation Model Packet Formats

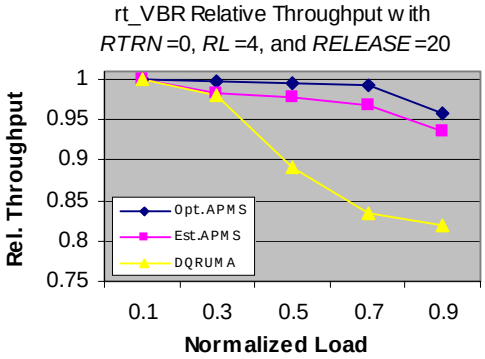
Packet Format	Field Name	Description	Size (bits)
watm_cell (Used for UL)	PK_TYPE	Packet type: WATM_DAT (0)	3
	MU_ID	Range of the MU identification: 0 – 99	7
	R_Mode	The actual rake-in mode used: 1 - 6	3
	Service_Type	CBR, rt_VBR, nrt_VBR, ABR	3
	ATM	ATM packet to be used	424
	CRC	Cyclic Redundancy Check code	12
	BCH	BCH forward error correction code	27
	(Reserved)	Reserved for expanded BCH and/or CRC code capability	32
watm_ar (Used for UL)	PK_TYPE	Packet type: ACCESS_REQ (1)	3
	MU_ID	Range of the MU identification: 0 – 99	7
	MSS	Multiple substreams requested: 0-12	4
	MSSRT	Multiple substreams for real-time requested: 0-12	4
	BCH	BCH forward error correction code	10
Watm_pb (Used for UL)	PK_TYPE	Packet type: PB_REQ(2)	3
	MSS	Multiple substreams requested: 0-12	4
	MSSRT	Multiple substreams for real-time requested: 0-12	4
	BCH	BCH forward error correction code	4
watm_ara (Used for DL)	PK_TYPE	Packet type: ACCESS_REQ_ACK (10)	3
	MU_ID	Range of the MU identification: 0 – 99	7
	ARA	0 or 1	1
	BCH	BCH forward error correction code	4
watm_xperm (Used for DL)	PK_TYPE	Packet type: XMT_PERM(11)	3
	MU_ID	Range of the MU identification: 0 – 99	7
	MSSA	Number of streams permitted: 0-12	4
	MSSRT	Real-time Multiple substreams permitted: 0-12	4
	RLA	Rake-in level to be used: 0-6	3
	M	Acknowledgment of the UL data received: Bit vector (bit0 to bit11)	12
	BCH	BCH forward error correction code	30

Appendix B: Relative Throughput Plots and t-Test Results for Method 1 (Number of MUs fixed to 5)

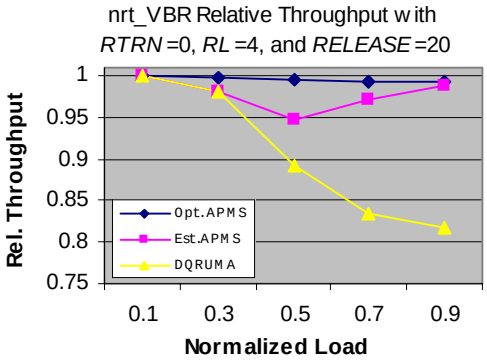
This appendix contains the relative throughput plots and their corresponding t-test results for method 1 case of varying the normalized load in the channel.



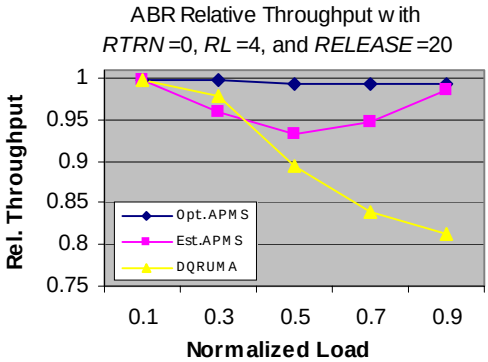
	Est. APMS	*DQRUMA*
Mean	0.97892784	0.90525756
Variance	0.000219142	0.006690417
Observations	5	5
Pearson Correlation	0.884307729	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.385654883	
P(T<=t) one-tail	0.03776174	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97245592	0.90476884
Variance	0.000565837	0.006786921
Observations	5	5
Pearson Correlation	0.861669431	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.40043565	
P(T<=t) one-tail	0.037160601	
t Critical one-tail	2.131846486	

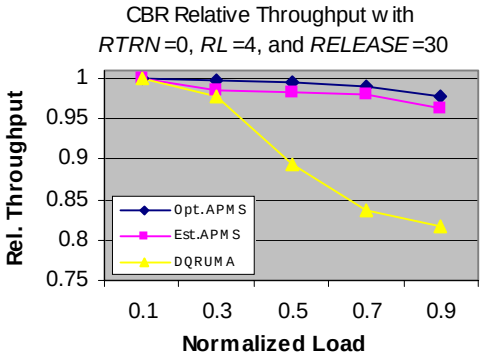


	Est. APMS	*DQRUMA*
Mean	0.97722996	0.90475668
Variance	0.000400466	0.006803505
Observations	5	5
Pearson Correlation	0.342189356	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.079285957	
P(T<=t) one-tail	0.053054322	
t Critical one-tail	2.131846486	

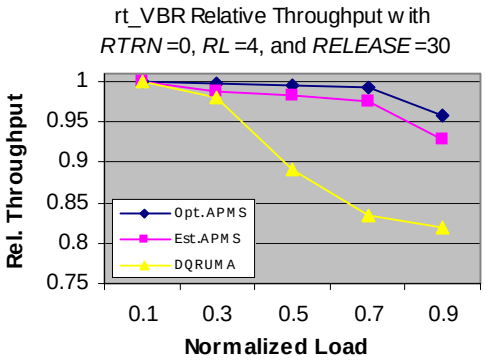


	Est. APMS	*DQRUMA*
Mean	0.96501324	0.90508996
Variance	0.00074582	0.006727657
Observations	5	5
Pearson Correlation	0.233021913	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.671054963	
P(T<=t) one-tail	0.085015239	
t Critical one-tail	2.131846486	

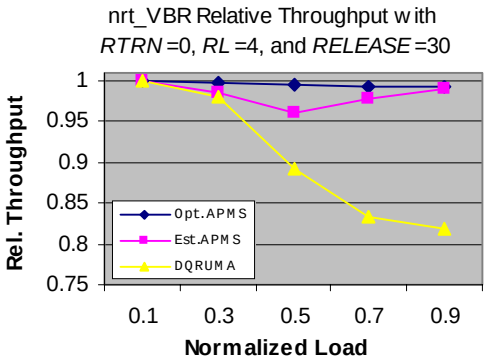
Figure B.1: Relative Throughput with RTRN=0, RL=4, and RELEASE=20



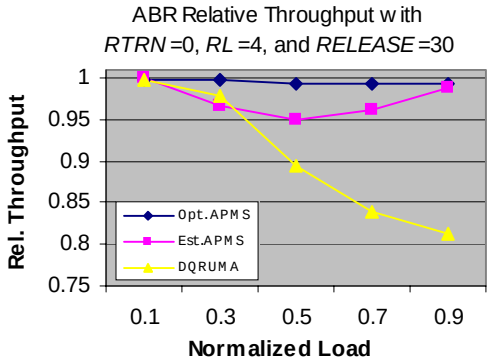
	Est. APMS	*DQRUMA*
Mean	0.98218028	0.90525756
Variance	0.000163944	0.006690417
Observations	5	5
Pearson Correlation	0.865103338	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.422284465	
P(T<=t) one-tail	0.036291771	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.9743592	0.90476884
Variance	0.000729858	0.006786921
Observations	5	5
Pearson Correlation	0.791521191	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.462395058	
P(T<=t) one-tail	0.034756125	
t Critical one-tail	2.131846486	

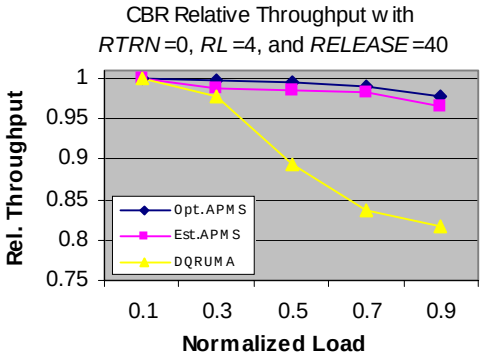


	Est. APMS	*DQRUMA*
Mean	0.9743592	0.90476884
Variance	0.000729858	0.006786921
Observations	5	5
Pearson Correlation	0.791521191	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.462395058	
P(T<=t) one-tail	0.034756125	
t Critical one-tail	2.131846486	

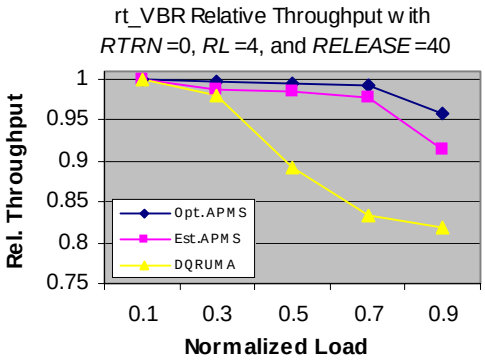


	Est. APMS	*DQRUMA*
Mean	0.97348296	0.90508996
Variance	0.000391873	0.006727657
Observations	5	5
Pearson Correlation	0.248310057	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.924743697	
P(T<=t) one-tail	0.063289251	
t Critical one-tail	2.131846486	

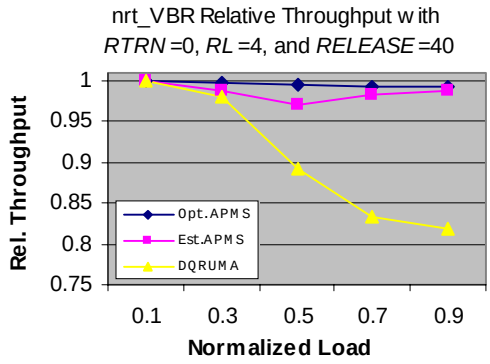
Figure B.2: Relative Throughput with RTRN=0, RL=4, and RELEASE=30



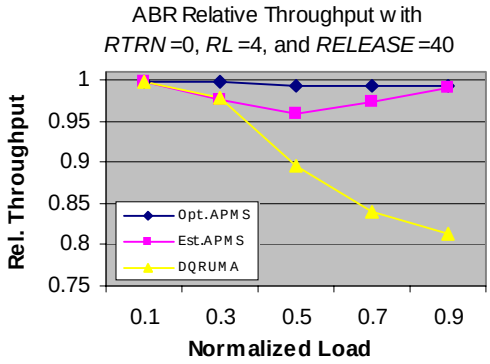
	Est. APMS	*DQRUMA*
Mean	0.9840204	0.90525756
Variance	0.000160344	0.006690417
Observations	5	5
Pearson Correlation	0.852031894	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.469604434	
P(T<=t) one-tail	0.03448802	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97306324	0.90476884
Variance	0.001160654	0.006786921
Observations	5	5
Pearson Correlation	0.7271215	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.456052107	
P(T<=t) one-tail	0.034993972	
t Critical one-tail	2.131846486	

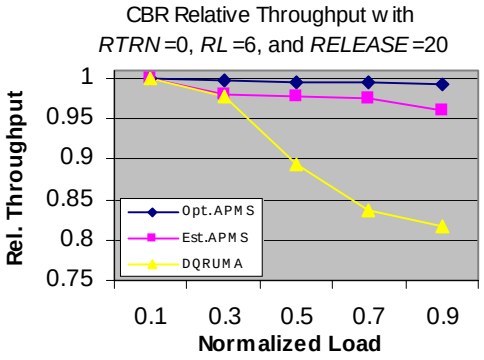


	Est. APMS	*DQRUMA*
Mean	0.98577356	0.90475668
Variance	0.000113606	0.006803505
Observations	5	5
Pearson Correlation	0.417632277	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.303926171	
P(T<=t) one-tail	0.041290847	
t Critical one-tail	2.131846486	

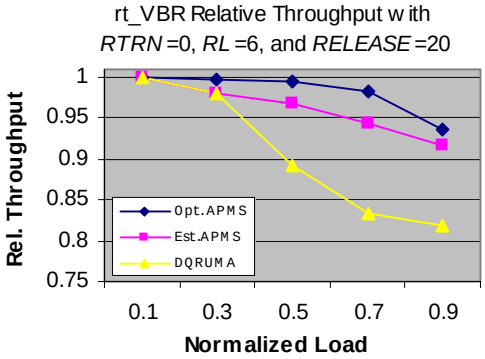


	Est. APMS	*DQRUMA*
Mean	0.97866712	0.90508996
Variance	0.000229726	0.006727657
Observations	5	5
Pearson Correlation	0.221342143	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.055413491	
P(T<=t) one-tail	0.05450916	
t Critical one-tail	2.131846486	

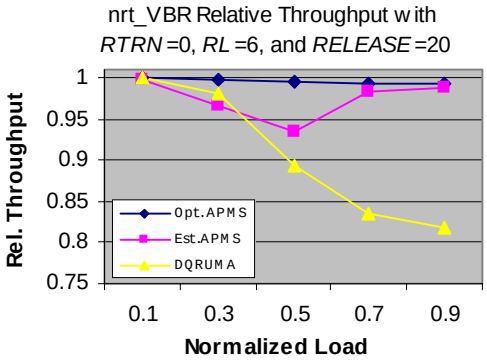
Figure B.3: Relative Throughput with RTRN=0, RL=4, and RELEASE=40



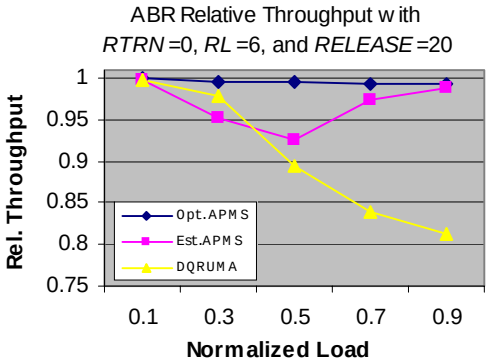
	Est. APMS	*DQRUMA*
Mean	0.9788164	0.90525756
Variance	0.000183073	0.006690417
Observations	5	5
Pearson Correlation	0.867944585	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.337300139	
P(T<=t) one-tail	0.039806595	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96155624	0.90476884
Variance	0.001059082	0.006786921
Observations	5	5
Pearson Correlation	0.945719604	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.410489504	
P(T<=t) one-tail	0.036757907	
t Critical one-tail	2.131846486	

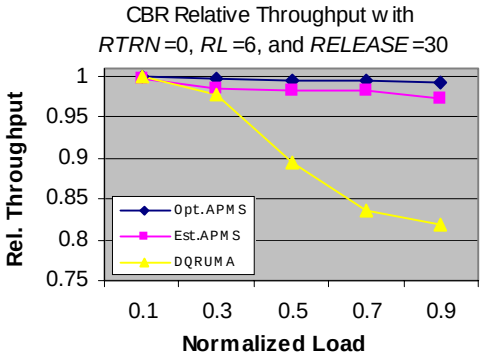


	Est. APMS	*DQRUMA*
Mean	0.97479704	0.90475668
Variance	0.000618999	0.006803505
Observations	5	5
Pearson Correlation	0.049189175	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.843088805	
P(T<=t) one-tail	0.069549661	
t Critical one-tail	2.131846486	

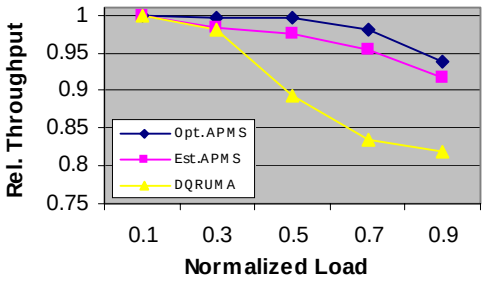


	Est. APMS	*DQRUMA*
Mean	0.96753692	0.90508996
Variance	0.000855051	0.006727657
Observations	5	5
Pearson Correlation	-0.027237612	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.589916886	
P(T<=t) one-tail	0.093529128	
t Critical one-tail	2.131846486	

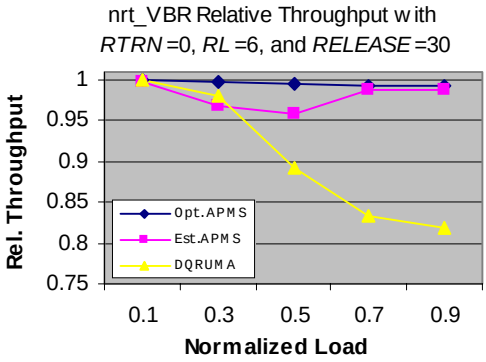
Figure B.4: Relative Throughput with RTRN=0, RL=6, and RELEASE=20



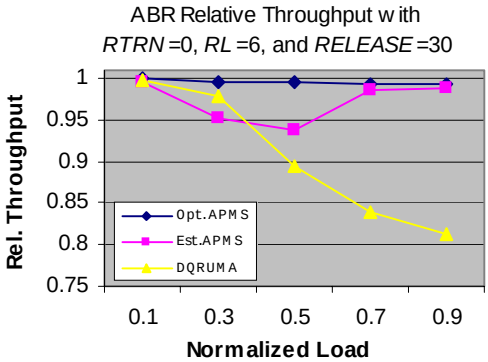
	Est. APMS	*DQRUMA*
Mean	0.9843986	0.90525756
Variance	9.04216E-05	0.006690417
Observations	5	5
Pearson Correlation	0.860296699	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.398746987	
P(T<=t) one-tail	0.037228728	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96607344	0.90476884
Variance	0.000990737	0.006786921
Observations	5	5
Pearson Correlation	0.900874545	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.459838336	
P(T<=t) one-tail	0.034851774	
t Critical one-tail	2.131846486	

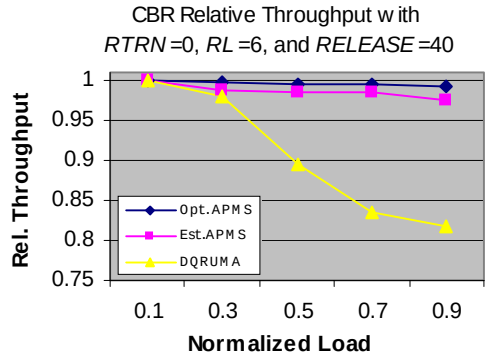


	Est. APMS	*DQRUMA*
Mean	0.98076396	0.90475668
Variance	0.000272599	0.006803505
Observations	5	5
Pearson Correlation	-0.031332248	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.00835248	
P(T<=t) one-tail	0.057507446	
t Critical one-tail	2.131846486	

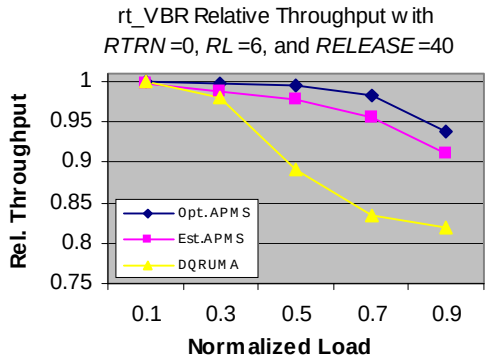


	Est. APMS	*DQRUMA*
Mean	0.97217696	0.90508996
Variance	0.000648644	0.006727657
Observations	5	5
Pearson Correlation	-0.144551128	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.679251047	
P(T<=t) one-tail	0.084201402	
t Critical one-tail	2.131846486	

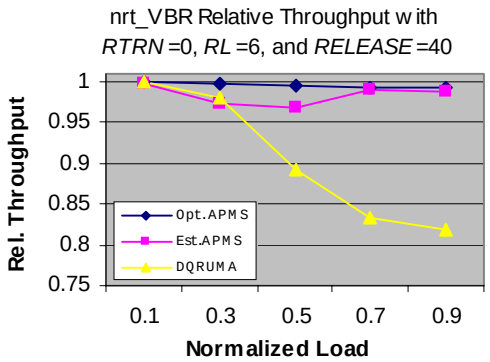
Figure B.5: Relative Throughput with RTRN=0, RL=6, and RELEASE=30



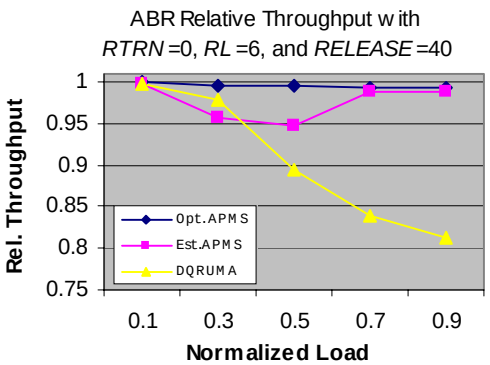
	Est. APMS	*DQRUMA*
Mean	0.98637688	0.90525756
Variance	6.9533E-05	0.006690417
Observations	5	5
Pearson Correlation	0.860469003	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.42690222	
P(T<=t) one-tail	0.036111108	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96597812	0.90476884
Variance	0.001215671	0.006786921
Observations	5	5
Pearson Correlation	0.871582026	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.500720944	
P(T<=t) one-tail	0.033357564	
t Critical one-tail	2.131846486	

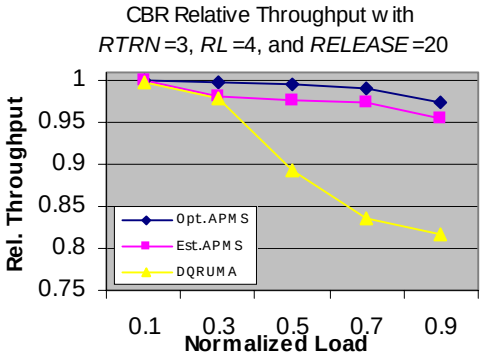


	Est. APMS	*DQRUMA*
Mean	0.98332952	0.90475668
Variance	0.000169292	0.006803505
Observations	5	5
Pearson Correlation	-0.010583429	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.10062089	
P(T<=t) one-tail	0.051790332	
t Critical one-tail	2.131846486	

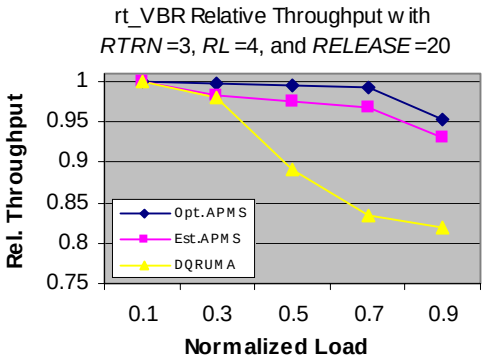


	Est. APMS	*DQRUMA*
Mean	0.97594376	0.90508996
Variance	0.000497938	0.006727657
Observations	5	5
Pearson Correlation	-0.127489036	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.806427886	
P(T<=t) one-tail	0.072574476	
t Critical one-tail	2.131846486	

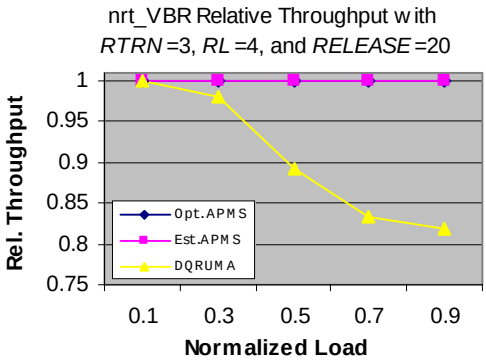
Figure B.6: Relative Throughput with RTRN=0, RL=6, and RELEASE=40



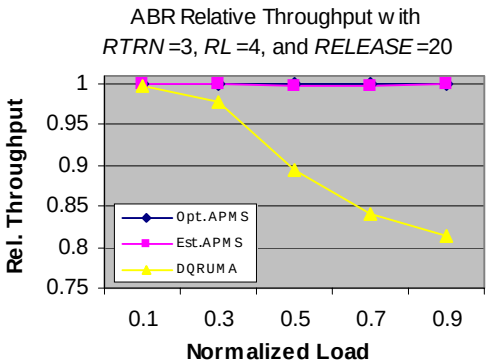
	Est. APMS	*DQRUMA*
Mean	0.9778262	0.90525756
Variance	0.00024788	0.006690417
Observations	5	5
Pearson Correlation	0.880832125	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.374611968	
P(T<=t) one-tail	0.03821805	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97115764	0.90476884
Variance	0.000656618	0.006786921
Observations	5	5
Pearson Correlation	0.865123627	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.41104795	
P(T<=t) one-tail	0.036735684	
t Critical one-tail	2.131846486	

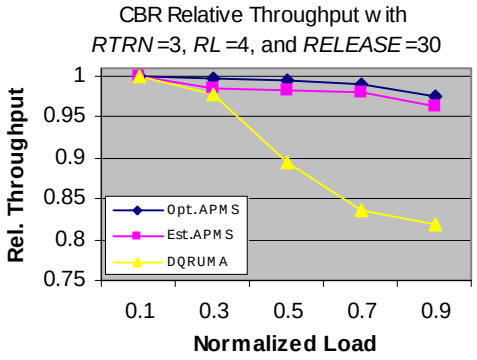


	Est. APMS	*DQRUMA*
Mean	0.99967804	0.90475668
Variance	8.35351E-08	0.006803505
Observations	5	5
Pearson Correlation	0.747603943	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.580005397	
P(T<=t) one-tail	0.03066354	
t Critical one-tail	2.131846486	

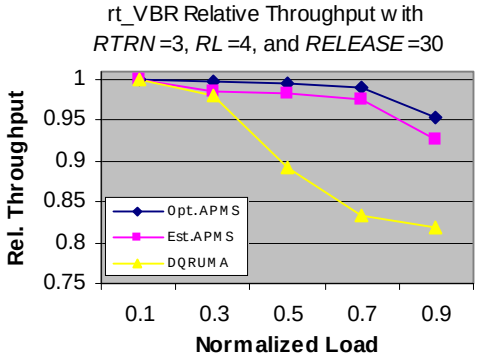


	Est. APMS	*DQRUMA*
Mean	0.99931884	0.90508996
Variance	6.56047E-07	0.006727657
Observations	5	5
Pearson Correlation	0.581350034	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.583588323	
P(T<=t) one-tail	0.030547809	
t Critical one-tail	2.131846486	

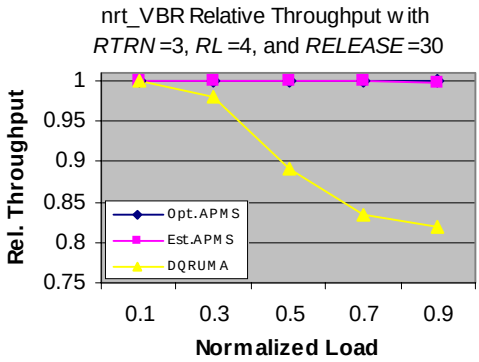
Figure B.7: Relative Throughput with RTRN=3, RL=4, and RELEASE=20



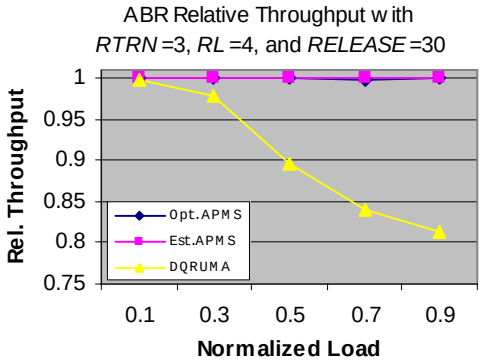
	Est. APMS	*DQRUMA*
Mean	0.98222292	0.90525756
Variance	0.000173393	0.006690417
Observations	5	5
Pearson Correlation	0.855607036	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.42883149	
P(T<=t) one-tail	0.036035931	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97397176	0.90476884
Variance	0.000761557	0.006786921
Observations	5	5
Pearson Correlation	0.783298001	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.450714343	
P(T<=t) one-tail	0.035195563	
t Critical one-tail	2.131846486	

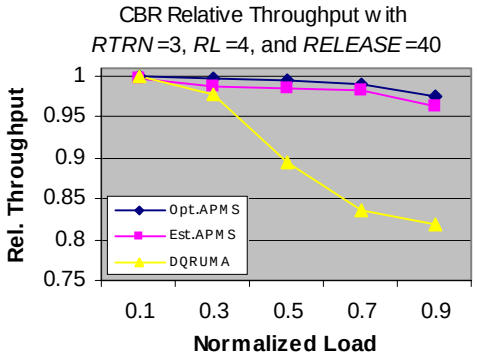


	Est. APMS	*DQRUMA*
Mean	0.99953916	0.90475668
Variance	4.9048E-07	0.006803505
Observations	5	5
Pearson Correlation	0.726319629	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585388521	
P(T<=t) one-tail	0.030489851	
t Critical one-tail	2.131846486	

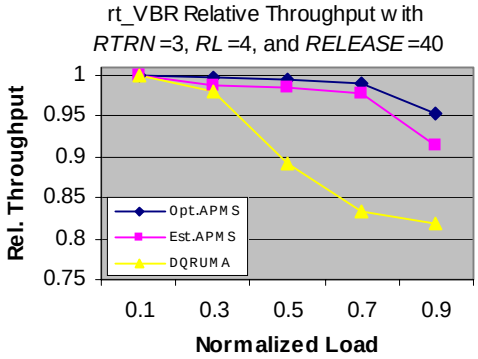


	Est. APMS	*DQRUMA*
Mean	0.99980452	0.90508996
Variance	4.04679E-08	0.006727657
Observations	5	5
Pearson Correlation	0.766227232	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.586939021	
P(T<=t) one-tail	0.030440033	
t Critical one-tail	2.131846486	

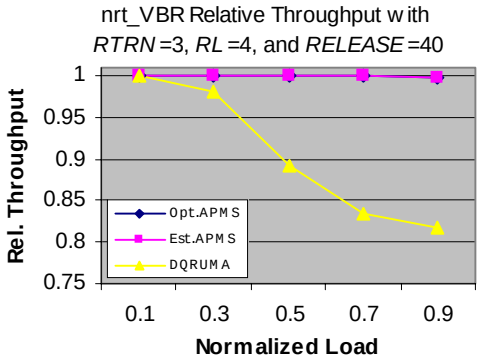
Figure B.8: Relative Throughput with RTRN=3, RL=4, and RELEASE=30



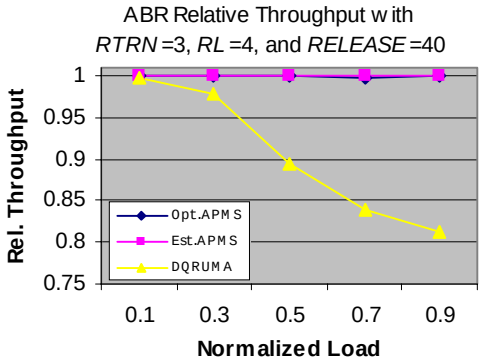
	Est. APMS	*DQRUMA*
Mean	0.98359412	0.90525756
Variance	0.000170005	0.006690417
Observations	5	5
Pearson Correlation	0.852725428	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.466984291	
P(T<=t) one-tail	0.034585185	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97304508	0.90476884
Variance	0.00115503	0.006786921
Observations	5	5
Pearson Correlation	0.730351263	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.459796017	
P(T<=t) one-tail	0.03485336	
t Critical one-tail	2.131846486	

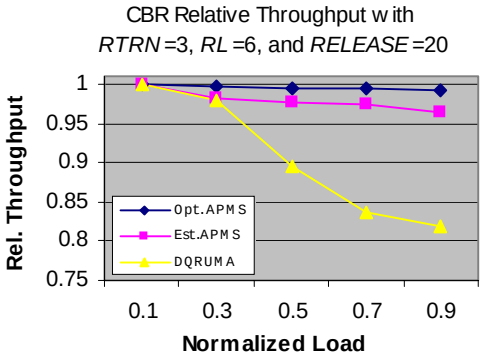


	Est. APMS	*DQRUMA*
Mean	0.99953916	0.90475668
Variance	3.98765E-07	0.006803505
Observations	5	5
Pearson Correlation	0.786263342	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585020228	
P(T<=t) one-tail	0.030501698	
t Critical one-tail	2.131846486	

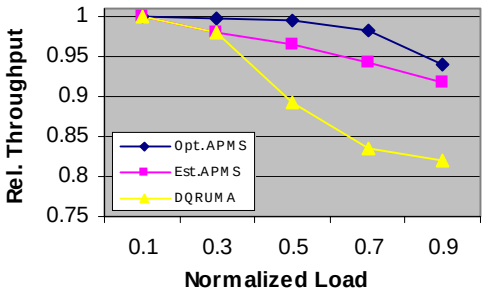


	Est. APMS	*DQRUMA*
Mean	0.99984104	0.90508996
Variance	5.04195E-08	0.006727657
Observations	5	5
Pearson Correlation	0.824748329	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.588918569	
P(T<=t) one-tail	0.030376565	
t Critical one-tail	2.131846486	

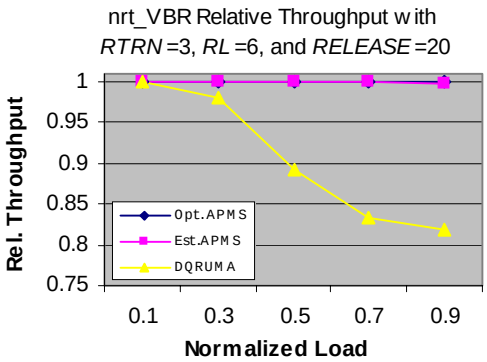
Figure B.9: Relative Throughput with RTRN=3, RL=4, and RELEASE=40



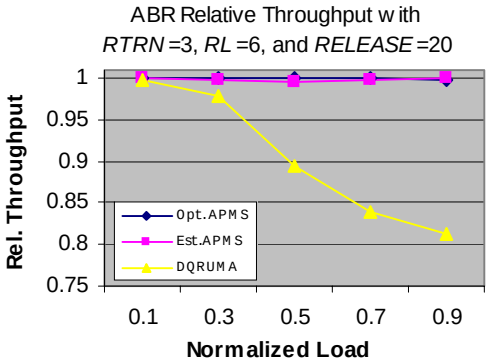
	Est. APMS	*DQRUMA*
Mean	0.97884972	0.90525756
Variance	0.000169149	0.006690417
Observations	5	5
Pearson Correlation	0.885263955	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.332776345	
P(T<=t) one-tail	0.040004215	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96115696	0.90476884
Variance	0.001003849	0.006786921
Observations	5	5
Pearson Correlation	0.954241721	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.378892902	
P(T<=t) one-tail	0.038040418	
t Critical one-tail	2.131846486	

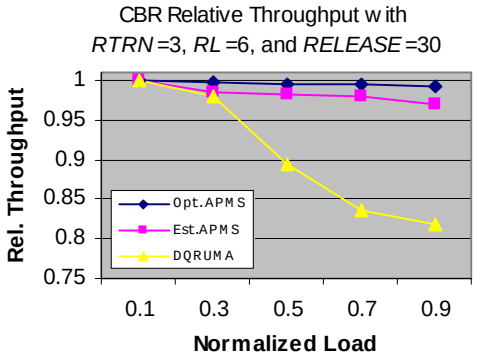


	Est. APMS	*DQRUMA*
Mean	0.99940208	0.90475668
Variance	3.69935E-07	0.006803505
Observations	5	5
Pearson Correlation	0.631381802	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.577731561	
P(T<=t) one-tail	0.030737247	
t Critical one-tail	2.131846486	

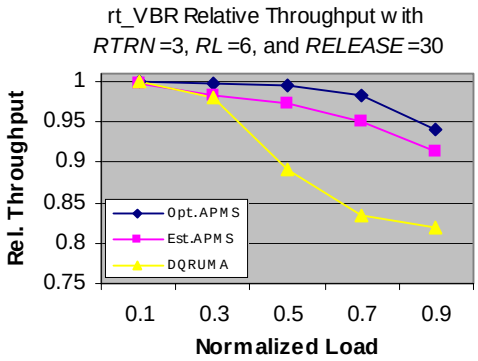


	Est. APMS	*DQRUMA*
Mean	0.99857932	0.90508996
Variance	2.26389E-06	0.006727657
Observations	5	5
Pearson Correlation	0.170520862	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.55625695	
P(T<=t) one-tail	0.031443459	
t Critical one-tail	2.131846486	

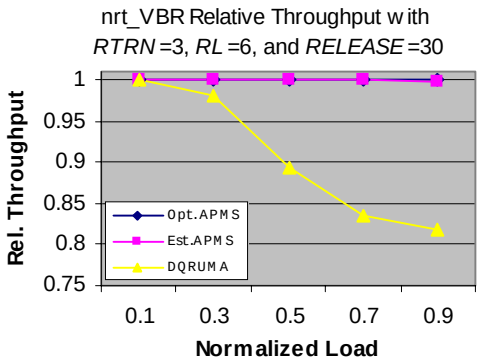
Figure B.10: Relative Throughput with RTRN=3, RL=6, and RELEASE=20



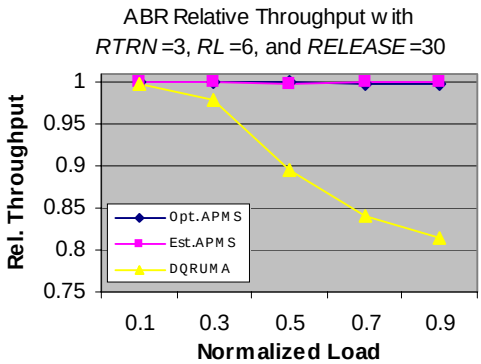
	Est. APMS	*DQRUMA*
Mean	0.98322988	0.90525756
Variance	0.000104874	0.006690417
Observations	5	5
Pearson Correlation	0.857531407	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.381752524	
P(T<=t) one-tail	0.037922283	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96405052	0.90476884
Variance	0.001105945	0.006786921
Observations	5	5
Pearson Correlation	0.908648354	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.455610776	
P(T<=t) one-tail	0.03501059	
t Critical one-tail	2.131846486	

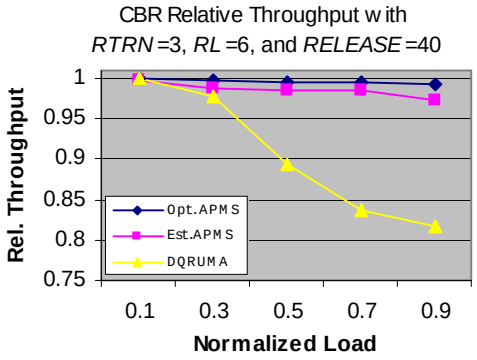


	Est. APMS	*DQRUMA*
Mean	0.9994474	0.90475668
Variance	5.27257E-07	0.006803505
Observations	5	5
Pearson Correlation	0.695908092	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.582772095	
P(T<=t) one-tail	0.03057413	
t Critical one-tail	2.131846486	

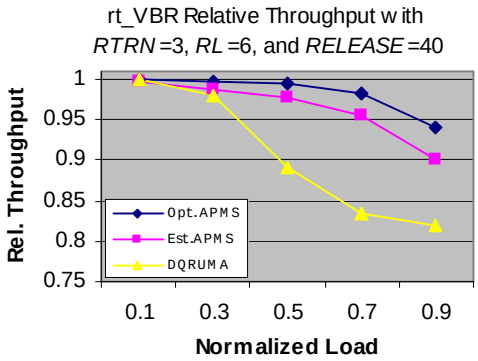


	Est. APMS	*DQRUMA*
Mean	0.99950952	0.90508996
Variance	2.98187E-07	0.006727657
Observations	5	5
Pearson Correlation	0.164676906	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.57680801	
P(T<=t) one-tail	0.030767242	
t Critical one-tail	2.131846486	

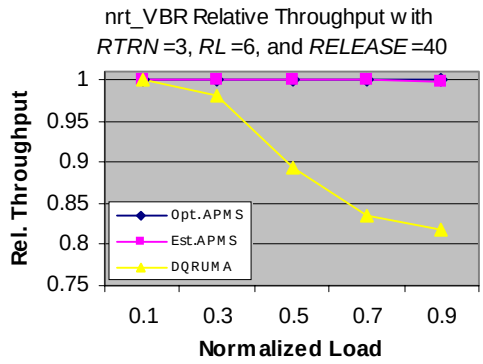
Figure B.11: Relative Throughput with RTRN=3, RL=6, and RELEASE=30



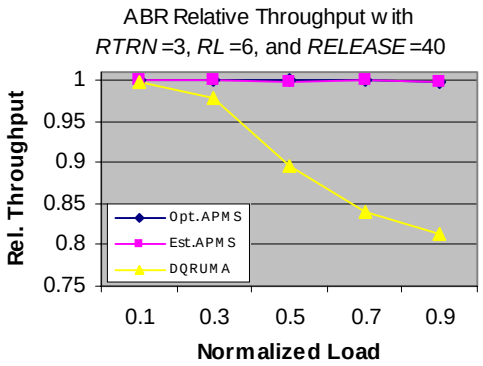
	Est. APMS	*DQRUMA*
Mean	0.9855796	0.90525756
Variance	8.0508E-05	0.006690417
Observations	5	5
Pearson Correlation	0.838101894	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.412882181	
P(T<=t) one-tail	0.036662801	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96443884	0.90476884
Variance	0.001470141	0.006786921
Observations	5	5
Pearson Correlation	0.850546922	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.484644166	
P(T<=t) one-tail	0.033936265	
t Critical one-tail	2.131846486	

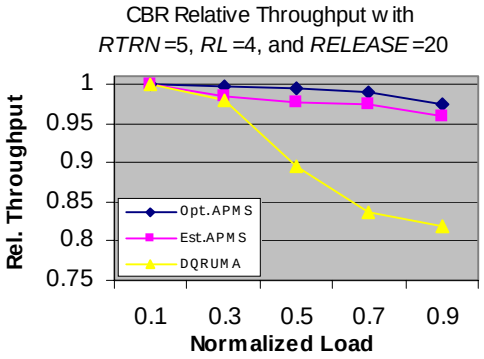


	Est. APMS	*DQRUMA*
Mean	0.9994364	0.90475668
Variance	4.36251E-07	0.006803505
Observations	5	5
Pearson Correlation	0.770995022	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.58261384	
P(T<=t) one-tail	0.030579236	
t Critical one-tail	2.131846486	

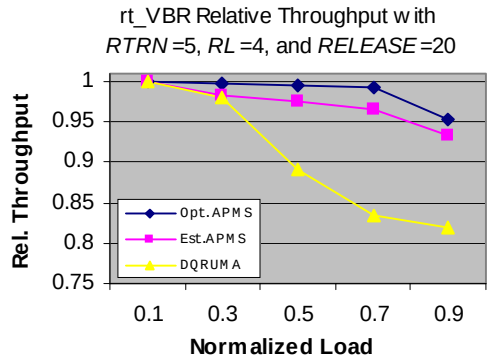


	Est. APMS	*DQRUMA*
Mean	0.99931356	0.90508996
Variance	3.77379E-07	0.006727657
Observations	5	5
Pearson Correlation	0.891943952	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585956344	
P(T<=t) one-tail	0.030471596	
t Critical one-tail	2.131846486	

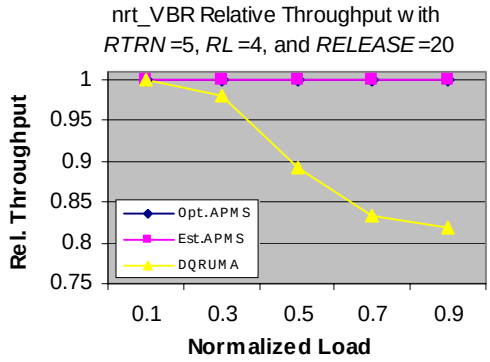
Figure B.12: Relative Throughput with RTRN=3, RL=6, and RELEASE=40



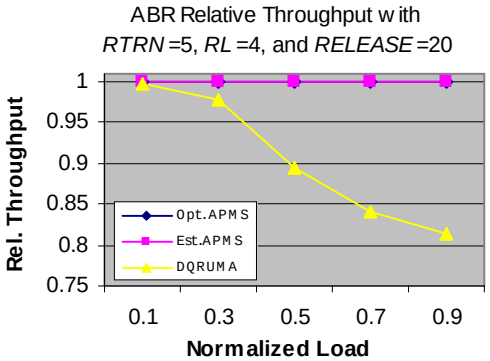
	Est. APMS	*DQRUMA*
Mean	0.97805768	0.90525756
Variance	0.000228943	0.006690417
Observations	5	5
Pearson Correlation	0.906451714	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.380650277	
P(T<=t) one-tail	0.037967769	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97164576	0.90476884
Variance	0.000618908	0.006786921
Observations	5	5
Pearson Correlation	0.871790733	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.415621845	
P(T<=t) one-tail	0.036554247	
t Critical one-tail	2.131846486	

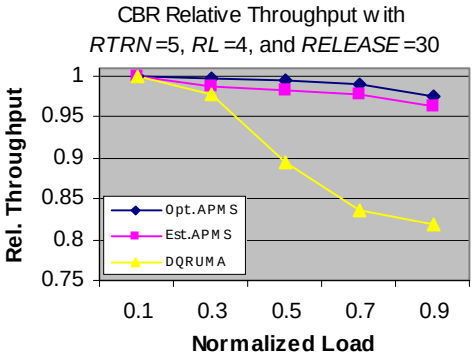


	Est. APMS	*DQRUMA*
Mean	0.99964288	0.90475668
Variance	1.5427E-07	0.006803505
Observations	5	5
Pearson Correlation	0.905265756	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.583431722	
P(T<=t) one-tail	0.030552857	
t Critical one-tail	2.131846486	

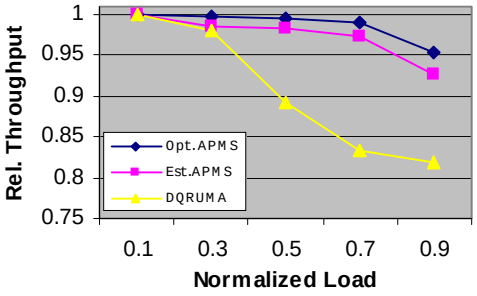


	Est. APMS	*DQRUMA*
Mean	0.99976796	0.90508996
Variance	6.5945E-08	0.006727657
Observations	5	5
Pearson Correlation	0.944211614	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.588735431	
P(T<=t) one-tail	0.03038243	
t Critical one-tail	2.131846486	

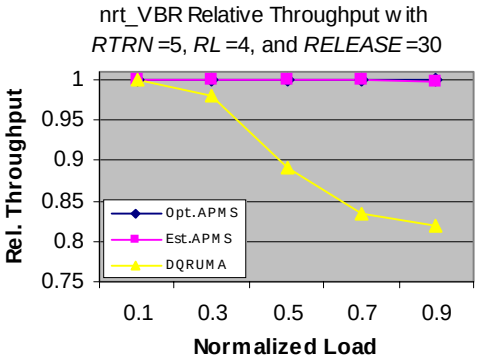
Figure B.13: Relative Throughput with RTRN=5, RL=4, and RELEASE=20



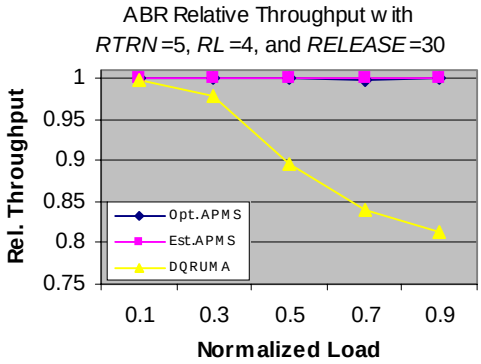
	Est. APMS	*DQRUMA*
Mean	0.98202704	0.90525756
Variance	0.000177243	0.006690417
Observations	5	5
Pearson Correlation	0.889538617	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.444759248	
P(T<=t) one-tail	0.035422029	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97349848	0.90476884
Variance	0.000761005	0.006786921
Observations	5	5
Pearson Correlation	0.802246451	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.460444118	
P(T<=t) one-tail	0.034829084	
t Critical one-tail	2.131846486	

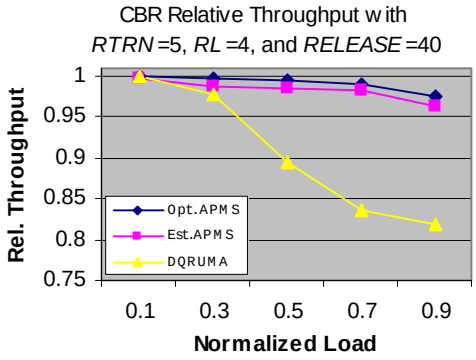


	Est. APMS	*DQRUMA*
Mean	0.99946364	0.90475668
Variance	5.01124E-07	0.006803505
Observations	5	5
Pearson Correlation	0.832705744	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585892367	
P(T<=t) one-tail	0.030473652	
t Critical one-tail	2.131846486	

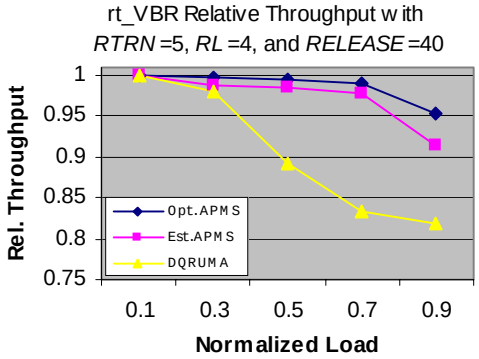


	Est. APMS	*DQRUMA*
Mean	0.99966316	0.90508996
Variance	9.16207E-08	0.006727657
Observations	5	5
Pearson Correlation	0.639688961	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.584317338	
P(T<=t) one-tail	0.030524323	
t Critical one-tail	2.131846486	

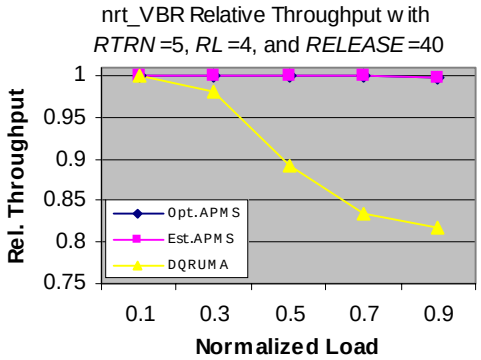
Figure B.14: Relative Throughput with RTRN=5, RL=4, and RELEASE=30



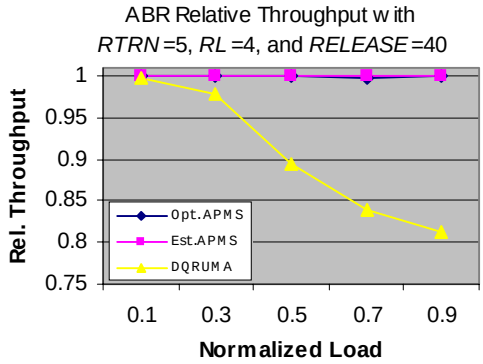
	Est. APMS	*DQRUMA*
Mean	0.98361192	0.90525756
Variance	0.000169883	0.006690417
Observations	5	5
Pearson Correlation	0.850091746	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.4660424	
P(T<=t) one-tail	0.034620191	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97304208	0.90476884
Variance	0.001155368	0.006786921
Observations	5	5
Pearson Correlation	0.731554296	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.461925059	
P(T<=t) one-tail	0.034773685	
t Critical one-tail	2.131846486	

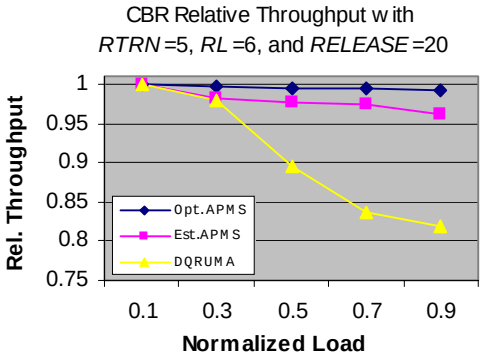


	Est. APMS	*DQRUMA*
Mean	0.9995394	0.90475668
Variance	3.9738E-07	0.006803505
Observations	5	5
Pearson Correlation	0.78521415	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.584978676	
P(T<=t) one-tail	0.030503035	
t Critical one-tail	2.131846486	

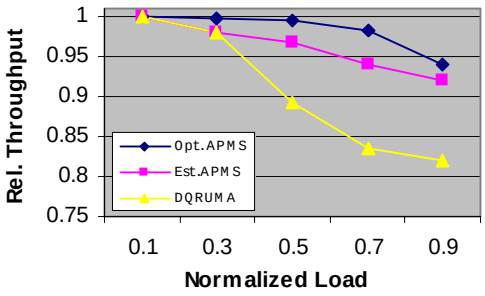


	Est. APMS	*DQRUMA*
Mean	0.9996324	0.90508996
Variance	1.25904E-07	0.006727657
Observations	5	5
Pearson Correlation	0.773258672	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.586029336	
P(T<=t) one-tail	0.03046925	
t Critical one-tail	2.131846486	

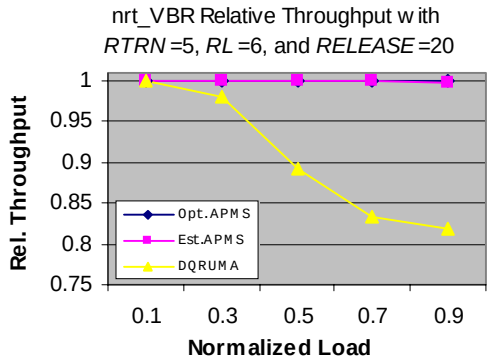
Figure B.15: Relative Throughput with RTRN=5, RL=4, and RELEASE=40



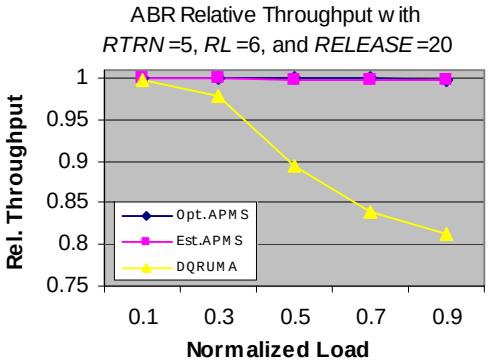
	Est. APMS	*DQRUMA*
Mean	0.97896136	0.90525756
Variance	0.000173969	0.006690417
Observations	5	5
Pearson Correlation	0.898190653	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.348075881	
P(T<=t) one-tail	0.039340287	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96138536	0.90476884
Variance	0.001007185	0.006786921
Observations	5	5
Pearson Correlation	0.957540503	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.398015218	
P(T<=t) one-tail	0.037258294	
t Critical one-tail	2.131846486	

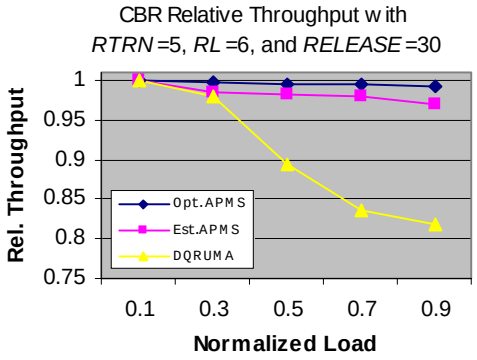


	Est. APMS	*DQRUMA*
Mean	0.9994212	0.90475668
Variance	6.26664E-07	0.006803505
Observations	5	5
Pearson Correlation	0.782622792	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585665688	
P(T<=t) one-tail	0.030480939	
t Critical one-tail	2.131846486	

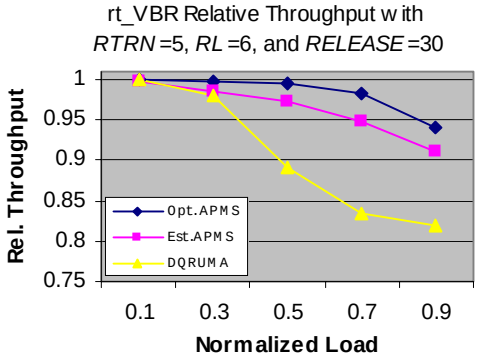


	Est. APMS	*DQRUMA*
Mean	0.99935088	0.90508996
Variance	3.43244E-07	0.006727657
Observations	5	5
Pearson Correlation	0.949989473	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.587263443	
P(T<=t) one-tail	0.030429621	
t Critical one-tail	2.131846486	

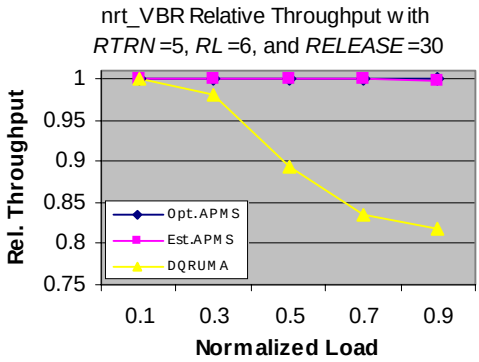
Figure B.16: Relative Throughput with RTRN=5, RL=6, and RELEASE=20



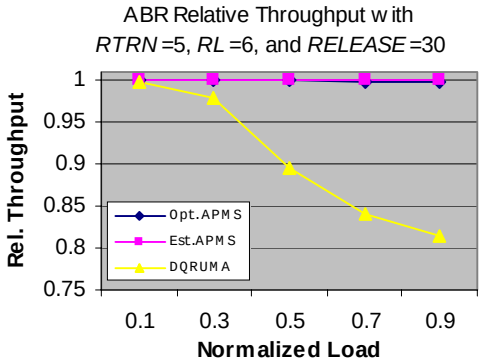
	Est. APMS	*DQRUMA*
Mean	0.98305288	0.90525756
Variance	0.00011103	0.006690417
Observations	5	5
Pearson Correlation	0.865622481	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.387365945	
P(T<=t) one-tail	0.03769159	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96339476	0.90476884
Variance	0.001183631	0.006786921
Observations	5	5
Pearson Correlation	0.91325759	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.480200113	
P(T<=t) one-tail	0.034098248	
t Critical one-tail	2.131846486	

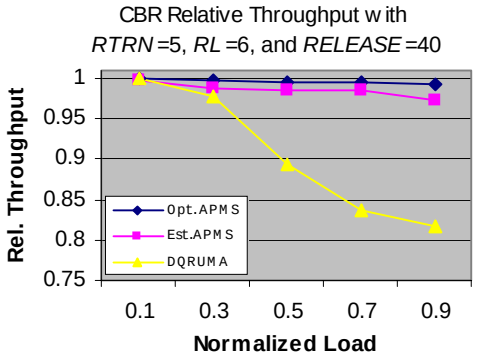


	Est. APMS	*DQRUMA*
Mean	0.99935772	0.90475668
Variance	1.10718E-06	0.006803505
Observations	5	5
Pearson Correlation	0.65268587	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585979707	
P(T<=t) one-tail	0.030470845	
t Critical one-tail	2.131846486	

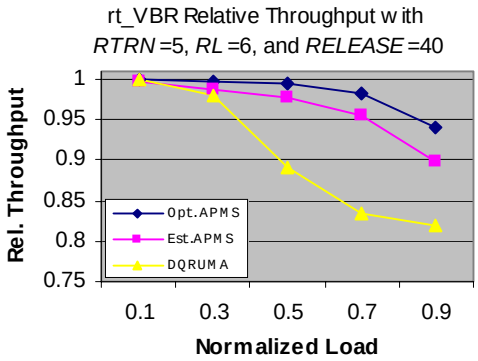


	Est. APMS	*DQRUMA*
Mean	0.99977832	0.90508996
Variance	5.07546E-08	0.006727657
Observations	5	5
Pearson Correlation	0.846889263	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.587382319	
P(T<=t) one-tail	0.030425807	
t Critical one-tail	2.131846486	

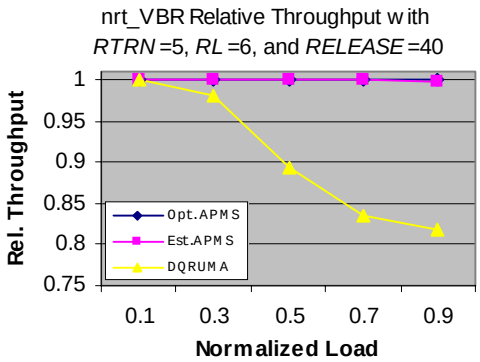
Figure B.17: Relative Throughput with RTRN=5, RL=6, and RELEASE=30



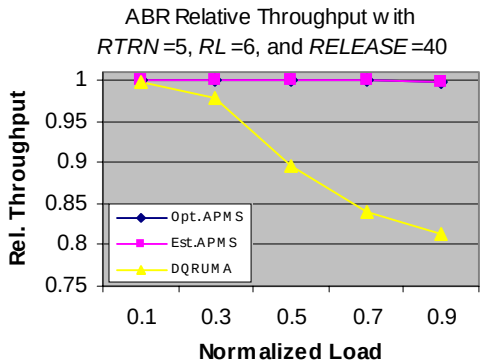
	Est. APMS	*DQRUMA*
Mean	0.98557484	0.90525756
Variance	8.81642E-05	0.006690417
Observations	5	5
Pearson Correlation	0.846365148	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.42639761	
P(T<=t) one-tail	0.0361308	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96394076	0.90476884
Variance	0.001544994	0.006786921
Observations	5	5
Pearson Correlation	0.841798875	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.46543803	
P(T<=t) one-tail	0.034642673	
t Critical one-tail	2.131846486	

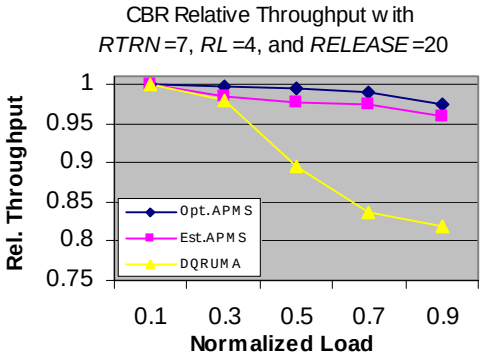


	Est. APMS	*DQRUMA*
Mean	0.99956996	0.90475668
Variance	3.97445E-07	0.006803505
Observations	5	5
Pearson Correlation	0.70406355	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.584191705	
P(T<=t) one-tail	0.030528369	
t Critical one-tail	2.131846486	

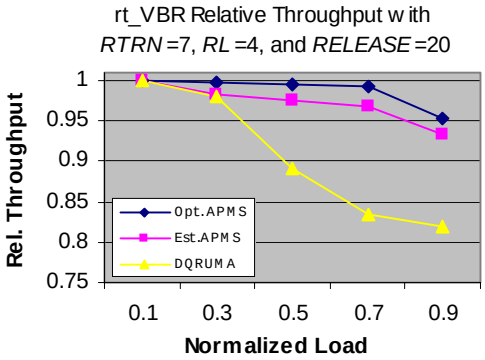


	Est. APMS	*DQRUMA*
Mean	0.99957508	0.90508996
Variance	5.73642E-07	0.006727657
Observations	5	5
Pearson Correlation	0.694399289	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.592390825	
P(T<=t) one-tail	0.030265604	
t Critical one-tail	2.131846486	

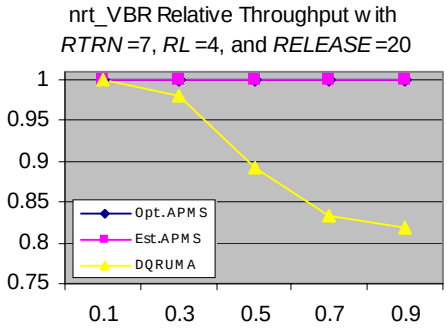
Figure B.18: Relative Throughput with RTRN=5, RL=6, and RELEASE=40



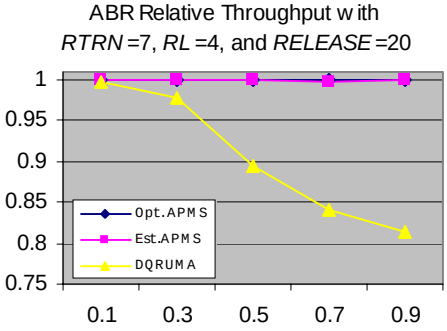
	Est. APMS	*DQRUMA*
Mean	0.97851124	0.90525756
Variance	0.000225812	0.006690417
Observations	5	5
Pearson Correlation	0.894671217	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.384991304	
P(T<=t) one-tail	0.037788985	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97172712	0.90476884
Variance	0.000618459	0.006786921
Observations	5	5
Pearson Correlation	0.869396087	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.415203932	
P(T<=t) one-tail	0.036570782	
t Critical one-tail	2.131846486	

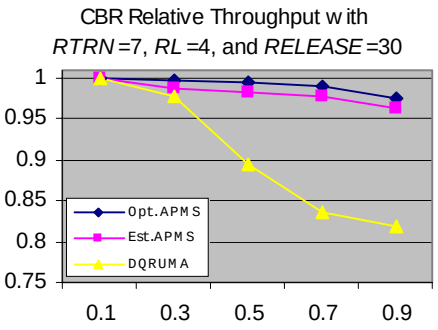


	Est. APMS	*DQRUMA*
Mean	0.99960604	0.90475668
Variance	2.05736E-07	0.006803505
Observations	5	5
Pearson Correlation	0.880579882	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.583804769	
P(T<=t) one-tail	0.030540834	
t Critical one-tail	2.131846486	

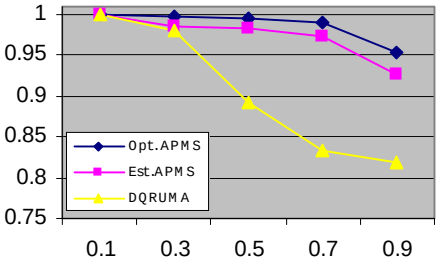


	Est. APMS	*DQRUMA*
Mean	0.99968776	0.90508996
Variance	2.32919E-07	0.006727657
Observations	5	5
Pearson Correlation	0.72263035	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.589888218	
P(T<=t) one-tail	0.030345532	
t Critical one-tail	2.131846486	

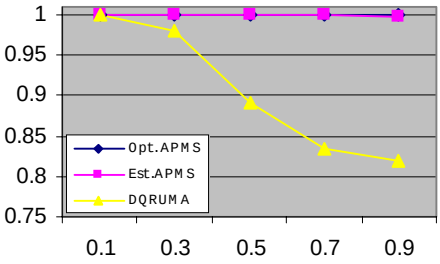
Figure B.19: Relative Throughput with RTRN=7, RL=4, and RELEASE=20



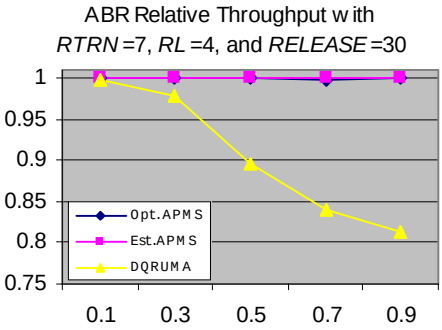
	Est. APMS	*DQRUMA*
Mean	0.98203264	0.90525756
Variance	0.000177252	0.006690417
Observations	5	5
Pearson Correlation	0.889444199	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.444896812	
P(T<=t) one-tail	0.035416779	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.97352292	0.90476884
Variance	0.000761536	0.006786921
Observations	5	5
Pearson Correlation	0.801796886	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.460908014	
P(T<=t) one-tail	0.03481172	
t Critical one-tail	2.131846486	

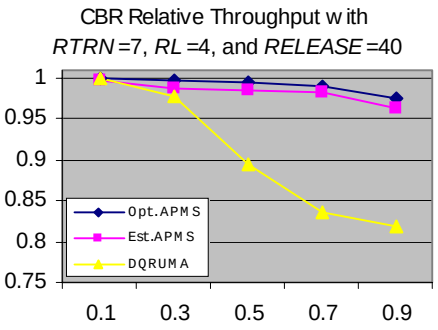


	Est. APMS	*DQRUMA*
Mean	0.99946048	0.90475668
Variance	4.98226E-07	0.006803505
Observations	5	5
Pearson Correlation	0.835977245	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585825797	
P(T<=t) one-tail	0.030475792	
t Critical one-tail	2.131846486	

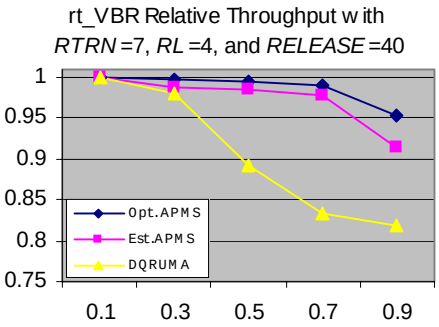


	Est. APMS	*DQRUMA*
Mean	0.99973872	0.90508996
Variance	7.48388E-08	0.006727657
Observations	5	5
Pearson Correlation	0.666983566	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.586031769	
P(T<=t) one-tail	0.030469172	
t Critical one-tail	2.131846486	

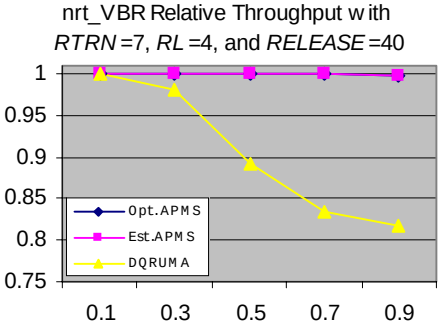
Figure B.20: Relative Throughput with RTRN=7, RL=4, and RELEASE=30



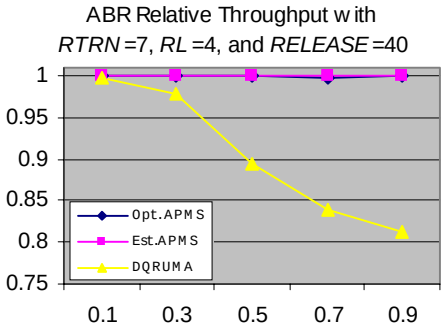
	Est. APMS	*DQRUMA*
Mean	0.98367356	0.90525756
Variance	0.000170236	0.006690417
Observations	5	5
Pearson Correlation	0.848410126	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.467478671	
P(T<=t) one-tail	0.034566828	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.973064	0.90476884
Variance	0.001156071	0.006786921
Observations	5	5
Pearson Correlation	0.731208155	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.462268608	
P(T<=t) one-tail	0.034760848	
t Critical one-tail	2.131846486	

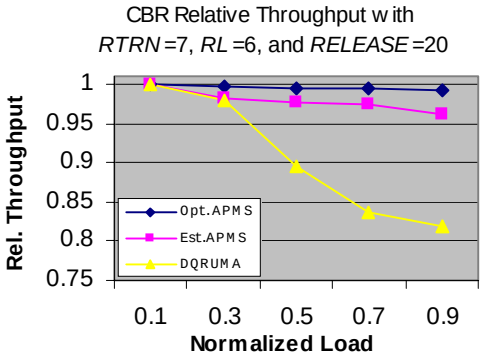


	Est. APMS	*DQRUMA*
Mean	0.99954548	0.90475668
Variance	4.01632E-07	0.006803505
Observations	5	5
Pearson Correlation	0.779217887	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585106921	
P(T<=t) one-tail	0.030498909	
t Critical one-tail	2.131846486	

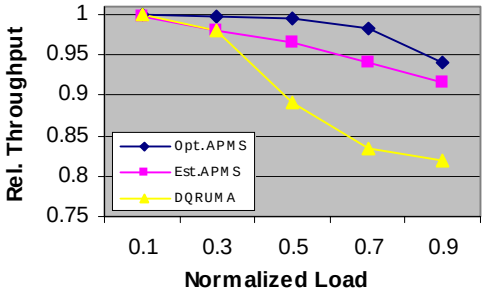


	Est. APMS	*DQRUMA*
Mean	0.99968848	0.90508996
Variance	8.11469E-08	0.006727657
Observations	5	5
Pearson Correlation	0.934101341	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.587308891	
P(T<=t) one-tail	0.030428163	
t Critical one-tail	2.131846486	

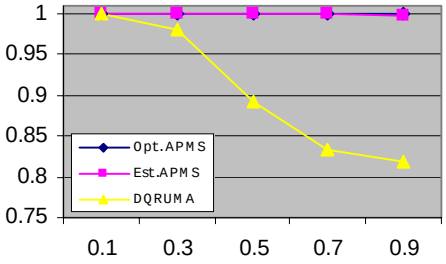
Figure B.21: Relative Throughput with RTRN=7, RL=4, and RELEASE=40



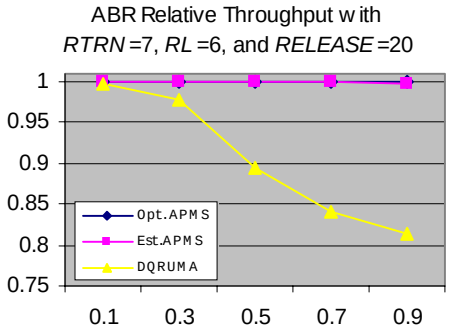
	Est. APMS	*DQRUMA*
Mean	0.97848784	0.90525756
Variance	0.00019126	0.006690417
Observations	5	5
Pearson Correlation	0.893228162	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.348659047	
P(T<=t) one-tail	0.039315227	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96046648	0.90476884
Variance	0.00106473	0.006786921
Observations	5	5
Pearson Correlation	0.96028369	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.401810599	
P(T<=t) one-tail	0.037105235	
t Critical one-tail	2.131846486	

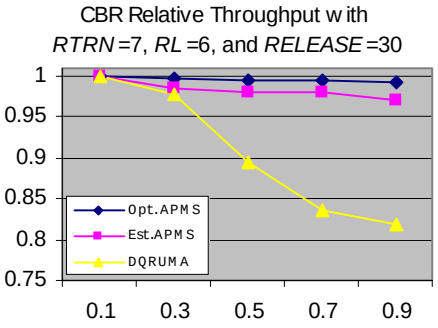


	Est. APMS	*DQRUMA*
Mean	0.9993044	0.90475668
Variance	6.07875E-07	0.006803505
Observations	5	5
Pearson Correlation	0.891753331	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.584889282	
P(T<=t) one-tail	0.030505912	
t Critical one-tail	2.131846486	

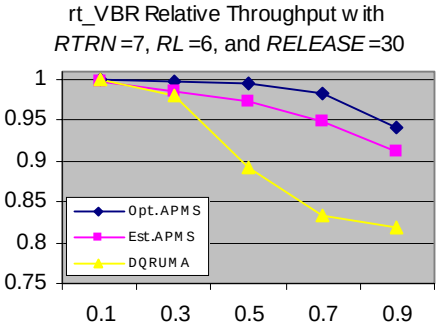


	Est. APMS	*DQRUMA*
Mean	0.9994966	0.90508996
Variance	2.35655E-07	0.006727657
Observations	5	5
Pearson Correlation	0.900049263	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.587460777	
P(T<=t) one-tail	0.03042329	
t Critical one-tail	2.131846486	

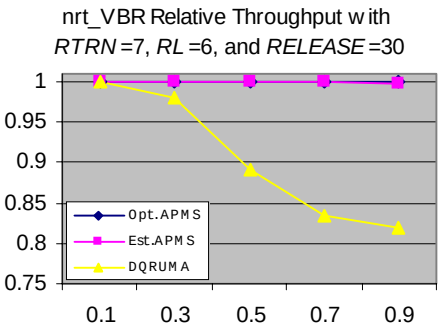
Figure B.22: Relative Throughput with RTRN=7, RL=6, and RELEASE=20



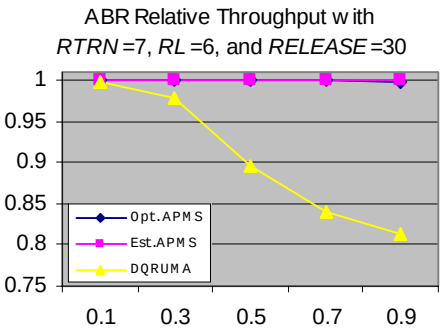
	Est. APMS	*DQRUMA*
Mean	0.98299636	0.90525756
Variance	0.000111435	0.006690417
Observations	5	5
Pearson Correlation	0.869822163	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.387782679	
P(T<=t) one-tail	0.037674527	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.9636586	0.90476884
Variance	0.001173998	0.006786921
Observations	5	5
Pearson Correlation	0.908623834	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.47474812	
P(T<=t) one-tail	0.034298174	
t Critical one-tail	2.131846486	

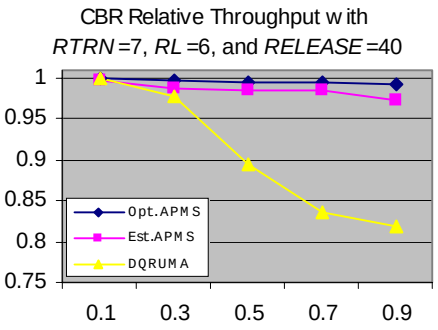


	Est. APMS	*DQRUMA*
Mean	0.9993168	0.90475668
Variance	1.06953E-06	0.006803505
Observations	5	5
Pearson Correlation	0.690167372	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585728653	
P(T<=t) one-tail	0.030478914	
t Critical one-tail	2.131846486	

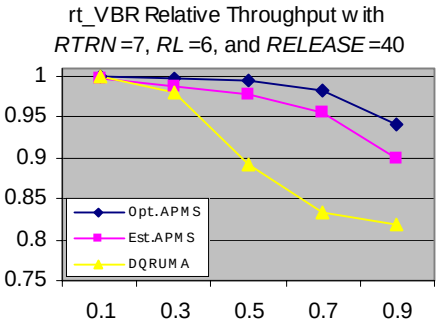


	Est. APMS	*DQRUMA*
Mean	0.99982712	0.90508996
Variance	4.27991E-08	0.006727657
Observations	5	5
Pearson Correlation	0.896288481	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.588547063	
P(T<=t) one-tail	0.030388465	
t Critical one-tail	2.131846486	

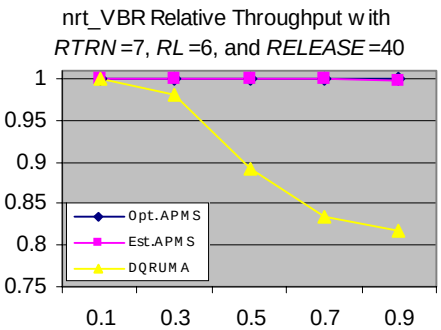
Figure B.23: Relative Throughput with RTRN=7, RL=6, and RELEASE=30



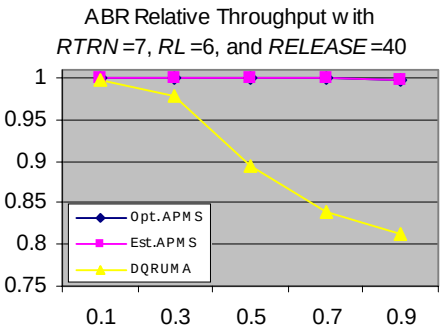
	Est. APMS	*DQRUMA*
Mean	0.98561628	0.90525756
Variance	8.81128E-05	0.006690417
Observations	5	5
Pearson Correlation	0.845863947	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.427406472	
P(T<=t) one-tail	0.036091442	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.96389172	0.90476884
Variance	0.00154325	0.006786921
Observations	5	5
Pearson Correlation	0.842513892	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.464804722	
P(T<=t) one-tail	0.03466625	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.99957464	0.90475668
Variance	3.9971E-07	0.006803505
Observations	5	5
Pearson Correlation	0.700655017	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.584290579	
P(T<=t) one-tail	0.030525185	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.99959044	0.90508996
Variance	5.86177E-07	0.006727657
Observations	5	5
Pearson Correlation	0.683970636	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.592737161	
P(T<=t) one-tail	0.030254562	
t Critical one-tail	2.131846486	

Figure B.24: Relative Throughput with RTRN=7, RL=6, and RELEASE=40

Appendix C: Confidence Interval for Method 1 (Number of MUs fixed to 5)

This appendix contains the confidence intervals of the relative throughputs for traffic following Method 1, where the number of MUs in the system is fixed to 5, and the load is varied within each MU. Table C.1 to C.24 list the confidence intervals for various combinations of factors pertaining the APMS protocol. Table C.25 shows the confidence interval for the relative throughput resulting from the DQRUMA protocol.

Confidence Interval for RTRN=0 and RL=4

Table C.1: Confidence Interval for RTRN=0, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00033	0.00023	0.00013	5.4E-05	9E-05	5.7E-05	0.00061	0.00164
0.3	0.00063	0.00044	0.00083	0.00016	0.0006	0.00019	0.00363	0.00065
0.5	0.00095	0.00044	0.00036	0.00037	0.00079	0.0003	0.00334	0.00176
0.7	0.00068	0.00012	0.0016	0.00029	0.00155	0.00026	0.00293	0.00094
0.9	0.00294	0.00141	0.00375	0.00333	0.00088	0.0004	0.00168	0.00043

Table C.2: Confidence Interval for RTRN=0, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00029	0.00023	0.00012	5.4E-05	0.00016	5.7E-05	0.00039	0.00164
0.3	0.00072	0.00044	0.00036	0.00016	0.00056	0.00019	0.00401	0.00065
0.5	0.00049	0.00044	0.00045	0.00037	0.00097	0.0003	0.00439	0.00176
0.7	0.00085	0.00012	0.00119	0.00029	0.0003	0.00026	0.00199	0.00094
0.9	0.00249	0.00141	0.0062	0.00333	0.00071	0.0004	0.00108	0.00043

Table C.3: Confidence Interval for RTRN=0, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00029	0.00023	0.00012	5.4E-05	0.00013	5.7E-05	0.00177	0.00164
0.3	0.00056	0.00044	0.00063	0.00016	0.00076	0.00019	0.00549	0.00065
0.5	0.00054	0.00044	0.00048	0.00037	0.00122	0.0003	0.00357	0.00176
0.7	0.00058	0.00012	0.00095	0.00029	0.00071	0.00026	0.00148	0.00094
0.9	0.00123	0.00141	0.00307	0.00333	0.00053	0.0004	0.00139	0.00043

Confidence Interval for RTRN=0 and RL=6

Table C.4: Confidence Interval for RTRN=0, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00038	0.0001	0.00015	5.3E-05	0.00017	3.5E-05	0.00141	0.00022
0.3	0.00119	0.00012	0.00064	0.00019	0.0018	0.00016	0.00507	0.00151
0.5	0.00093	0.00031	0.00081	0.0002	0.00165	0.00062	0.00173	0.00102
0.7	9.6E-05	0.00018	0.00064	0.00119	0.00069	0.00045	0.0003	0.00079
0.9	0.00219	0.00041	0.00389	0.00232	0.00031	0.00032	0.00134	0.00031

Table C.5: Confidence Interval for RTRN=0, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00024	0.0001	0.00023	5.3E-05	0.00018	3.5E-05	0.00246	0.00022
0.3	0.00047	0.00012	0.00052	0.00019	0.00174	0.00016	0.00138	0.00151
0.5	0.00075	0.00031	0.00057	0.0002	0.00125	0.00062	0.00424	0.00102
0.7	0.00032	0.00018	0.00136	0.00119	0.00033	0.00045	0.00097	0.00079
0.9	0.00199	0.00041	0.00342	0.00232	0.00079	0.00032	0.0006	0.00031

Table C.6: Confidence Interval for RTRN=0, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00014	0.0001	9.2E-05	5.3E-05	0.00011	3.5E-05	0.00067	0.00022
0.3	0.00058	0.00012	0.00033	0.00019	0.00089	0.00016	0.00347	0.00151
0.5	9.7E-05	0.00031	0.00139	0.0002	0.00156	0.00062	0.00194	0.00102
0.7	0.00081	0.00018	0.00127	0.00119	0.00044	0.00045	0.00122	0.00079
0.9	0.00109	0.00041	0.00314	0.00232	0.00047	0.00032	0.00116	0.00031

Confidence Interval for RTRN=3 and RL=4

Table C.7: Confidence Interval for RTRN=3, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00045	0.00022	0.00012	5E-05	5.4E-06	3.7E-06	0.0	0.0
0.3	0.00054	0.00039	0.001	0.00016	1.1E-05	1.2E-05	0.0	0.0
0.5	0.00065	0.00027	0.00052	0.00021	0.00022	5.4E-05	0.00036	0.0
0.7	0.00078	0.00083	0.00101	0.00093	0.00019	0.00036	0.0002	0.00089
0.9	0.00249	0.00079	0.00418	0.00322	0.0004	0.00035	0.00024	0.00038

Table C.8: Confidence Interval for RTRN=3, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00042	0.00022	8.7E-05	5E-05	8E-06	3.7E-06	0.0	0.0
0.3	0.00122	0.00039	0.00043	0.00016	9E-06	1.2E-05	0.00012	0.0
0.5	0.00123	0.00027	0.00045	0.00021	2.9E-05	5.4E-05	4.9E-05	0.0
0.7	0.00064	0.00083	0.00129	0.00093	0.00027	0.00036	0.00052	0.00089
0.9	0.00316	0.00079	0.00659	0.00322	0.00045	0.00035	0.00019	0.00038

Table C.9: Confidence Interval for RTRN=3, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00015	0.00022	0.00016	5E-05	1.1E-05	3.7E-06	0.0	0.0
0.3	0.00075	0.00039	0.00055	0.00016	4.5E-06	1.2E-05	0.0	0.0
0.5	0.00041	0.00027	0.00022	0.00021	0.0001	5.4E-05	8.5E-05	0.0
0.7	0.00051	0.00083	0.00079	0.00093	0.00028	0.00036	0.00015	0.00089
0.9	0.00115	0.00079	0.0028	0.00322	0.00077	0.00035	0.00079	0.00038

Confidence Interval for RTRN=3 and RL=6

Table C.10: Confidence Interval for RTRN=3, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.0002	0.00021	7.7E-05	6E-05	2.1E-05	1.2E-05	0.0	0.0
0.3	0.00186	0.00015	0.00106	0.00011	0.0001	3.7E-05	0.00079	0.0
0.5	0.00059	0.00052	0.00094	0.00024	9.4E-05	6.6E-05	0.00064	0.00092
0.7	0.00077	0.00026	0.00209	0.00087	0.00018	0.00016	0.00078	0.00075
0.9	0.00204	0.00029	0.00462	0.00419	0.00056	0.00028	0.00048	0.00075

Table C.11: Confidence Interval for RTRN=3, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00015	0.00021	0.00012	6E-05	1.1E-05	1.2E-05	0.0	0.0
0.3	0.00094	0.00015	0.00032	0.00011	6E-05	3.7E-05	0.00024	0.0
0.5	0.00065	0.00052	0.00032	0.00024	0.00019	6.6E-05	0.00061	0.00092
0.7	0.00106	0.00026	0.00319	0.00087	0.0003	0.00016	0.00032	0.00075
0.9	0.0019	0.00029	0.00393	0.00419	0.00111	0.00028	0.00043	0.00075

Table C.12: Confidence Interval for RTRN=3, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00062	0.00021	0.00012	6E-05	1.2E-05	1.2E-05	6E-05	0.0
0.3	0.00074	0.00015	0.0006	0.00011	7.3E-05	3.7E-05	0.00013	0.0
0.5	0.00042	0.00052	0.00082	0.00024	0.0004	6.6E-05	0.00034	0.00092
0.7	0.00033	0.00026	0.00199	0.00087	0.00027	0.00016	0.00076	0.00075
0.9	0.00115	0.00029	0.00173	0.00419	0.00094	0.00028	0.00123	0.00075

Confidence Interval for RTRN=5 and RL=4

Table C.13: Confidence Interval for RTRN=5, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00045	0.00022	0.00012	5E-05	5.4E-06	3.7E-06	0.0	0.0
0.3	0.00048	0.00039	0.00064	0.00016	1E-05	1.2E-05	0.0	0.0
0.5	0.00068	0.00027	0.00041	0.00021	9.3E-05	5.4E-05	0.00013	#NUM!
0.7	0.0016	0.00083	0.002	0.00093	0.00052	0.00036	0.00059	0.00089
0.9	0.00285	0.00079	0.00502	0.00322	0.00045	0.00035	0.00027	0.00038

Table C.14: Confidence Interval for RTRN=5, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00042	0.00022	8.7E-05	5E-05	8E-06	3.7E-06	0.0	0.0
0.3	0.00118	0.00039	0.00057	0.00016	5.8E-06	1.2E-05	0.00025	0.0
0.5	0.00138	0.00027	0.00051	0.00021	0.00018	5.4E-05	0.0006	0.0
0.7	0.00108	0.00083	0.00079	0.00093	0.00068	0.00036	0.00076	0.00089
0.9	0.00316	0.00079	0.00659	0.00322	0.00045	0.00035	0.00019	0.00038

Table C.15: Confidence Interval for RTRN=5, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00015	0.00022	0.00016	5E-05	1.1E-05	3.7E-06	0.0	0.0
0.3	0.00056	0.00039	0.00049	0.00016	2.8E-06	1.2E-05	0.0	0.0
0.5	0.00095	0.00027	0.00029	0.00021	0.00011	5.4E-05	0.00086	0.0
0.7	0.00079	0.00083	0.00071	0.00093	0.00033	0.00036	0.0004	0.00089
0.9	0.00115	0.00079	0.0028	0.00322	0.00077	0.00035	0.00079	0.00038

Confidence Interval for RTRN=5 and RL=6

Table C.16: Confidence Interval for RTRN=5, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.0002	0.00021	7.7E-05	6E-05	2.1E-05	1.2E-05	0.0	0.0
0.3	0.00086	0.00015	0.00055	0.00011	1.5E-05	3.7E-05	6.9E-05	0.0
0.5	0.00114	0.00052	0.00071	0.00024	5.5E-05	6.6E-05	0.00026	0.00092
0.7	0.00077	0.00026	0.00199	0.00087	0.00034	0.00016	0.00108	0.00075
0.9	0.00306	0.00029	0.00554	0.00419	0.0005	0.00028	0.00102	0.00075

Table C.17: Confidence Interval for RTRN=5, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00015	0.00021	0.00012	6E-05	1.1E-05	1.2E-05	0.0	0.0
0.3	0.00068	0.00015	0.00043	0.00011	6.8E-05	3.7E-05	1.7E-05	0.0
0.5	0.00126	0.00052	0.00107	0.00024	6.3E-05	6.6E-05	0.00035	0.00092
0.7	0.0007	0.00026	0.00211	0.00087	0.00032	0.00016	0.00019	0.00075
0.9	0.00207	0.00029	0.00488	0.00419	0.00088	0.00028	0.00044	0.00075

Table C.18: Confidence Interval for RTRN=5, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00062	0.00021	0.00012	6E-05	1.2E-05	1.2E-05	6E-05	0.0
0.3	0.00093	0.00015	0.00058	0.00011	1.4E-05	3.7E-05	0.0	0.0
0.5	0.00085	0.00052	0.00089	0.00024	0.00013	6.6E-05	0.00015	0.00092
0.7	0.00063	0.00026	0.00238	0.00087	0.00015	0.00016	0.0003	0.00075
0.9	0.00172	0.00029	0.00369	0.00419	0.00082	0.00028	0.00112	0.00075

Confidence Interval for RTRN=7 and RL=4

Table C.19: Confidence Interval for RTRN=7, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00045	0.00022	0.00012	5E-05	5.4E-06	3.7E-06	0.0	0.0
0.3	0.00048	0.00039	0.00064	0.00016	1E-05	1.2E-05	0.0	0.0
0.5	0.00151	0.00027	0.00091	0.00021	4.9E-05	5.4E-05	0.0	0.0
0.7	0.00099	0.00083	0.00087	0.00093	0.00032	0.00036	0.00073	0.00089
0.9	0.00285	0.00079	0.00502	0.00322	0.00045	0.00035	0.00027	0.00038

Table C.20: Confidence Interval for RTRN=7, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00042	0.00022	8.7E-05	5E-05	8E-06	3.7E-06	0.0	0.0
0.3	0.00118	0.00039	0.00057	0.00016	5.8E-06	1.2E-05	0.00025	0.0
0.5	0.00142	0.00027	0.00063	0.00021	0.0002	5.4E-05	0.00025	0.0
0.7	0.00108	0.00083	0.00079	0.00093	0.00068	0.00036	0.00076	0.00089
0.9	0.00316	0.00079	0.00659	0.00322	0.00045	0.00035	0.00019	0.00038

Table C.21: Confidence Interval for RTRN=7, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00015	0.00022	0.00016	5E-05	1.1E-05	3.7E-06	0.0	0.0
0.3	0.00056	0.00039	0.00049	0.00016	2.8E-06	1.2E-05	0.0	0.0
0.5	0.00091	0.00027	0.00026	0.00021	0.00012	5.4E-05	0.00085	0.0
0.7	0.00079	0.00083	0.00071	0.00093	0.00033	0.00036	0.0004	0.00089
0.9	0.00115	0.00079	0.0028	0.00322	0.00077	0.00035	0.00079	0.00038

Confidence Interval for RTRN=7 and RL=6

Table C.22: Confidence Interval for RTRN=7, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.0002	0.00021	7.7E-05	6E-05	2.1E-05	1.2E-05	0.0	0.0
0.3	0.00088	0.00015	0.0003	0.00011	5.3E-05	3.7E-05	0.0	0.0
0.5	0.00148	0.00052	0.00175	0.00024	0.00028	6.6E-05	0.00074	0.00092
0.7	0.00091	0.00026	0.00183	0.00087	0.00066	0.00016	0.00059	0.00075
0.9	0.00181	0.00029	0.00324	0.00419	0.00049	0.00028	0.00102	0.00075

Table C.23: Confidence Interval for RTRN=7, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00015	0.00021	0.00012	6E-05	1.1E-05	1.2E-05	0.0	0.0
0.3	0.00062	0.00015	0.00043	0.00011	6.8E-05	3.7E-05	0.0	0.0
0.5	0.00024	0.00052	0.00091	0.00024	0.00012	6.6E-05	0.00017	0.00092
0.7	0.0007	0.00026	0.00213	0.00087	0.00032	0.00016	0.00019	0.00075
0.9	0.00207	0.00029	0.00488	0.00419	0.00088	0.00028	0.00044	0.00075

Table C.24: Confidence Interval for RTRN=7, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00062	0.00021	0.00012	6E-05	1.2E-05	1.2E-05	6E-05	0.0
0.3	0.00093	0.00015	0.00058	0.00011	1.4E-05	3.7E-05	0.0	0.0
0.5	0.00095	0.00052	0.00077	0.00024	0.00013	6.6E-05	6.3E-05	0.00092
0.7	0.00063	0.00026	0.00238	0.00087	0.00015	0.00016	0.0003	0.00075
0.9	0.00172	0.00029	0.00369	0.00419	0.00082	0.00028	0.00112	0.00075

Confidence Interval for DQRUMA (RTRN=0 and RL=1)

Table C.25: Confidence Interval for DQRUMA (RTRN=0 and RL=1)

	Traffic Type			
Load	CBR	rt-VBR	nrt-VBR	ABR
0.1	0.00042	7.6E-05	9.5E-05	0.00138
0.3	0.00218	0.0009	0.001	0.00398
0.5	0.00476	0.003	0.00274	0.00315
0.7	0.00482	0.00493	0.0042	0.00717
0.9	0.00551	0.00519	0.00543	0.00695

Appendix D: Average Delay for Method 1 (Number of MUs fixed to 5)

This appendix contains the average delay of for the uplink using Method 1, where the number of MUs in the system is fixed to 5, and the load is varied within each MU. Table D.1 to D.24 shows the average delay for each type of traffic using the APMS protocol. The average delay for the DQRUMA protocol is given in Table D.25.

Average Delay for RTRN=0 and RL=4

Table D.1: Average Delay for RTRN=0, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00107	0.00121	0.00121	0.0012	0.0012	0.00186	0.00226
0.3	0.00085	0.00086	0.00118	0.00118	0.00119	0.0012	0.00247	0.00301
0.5	0.00076	0.00076	0.00119	0.00119	0.00186	0.00365	0.00373	0.00508
0.7	0.00072	0.00072	0.00148	0.00164	0.00817	0.00963	0.00644	0.00862
0.9	0.00078	0.00081	0.00298	0.00283	0.02788	0.0222	0.01422	0.01233

Table D.2: Average Delay for RTRN=0, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00107	0.00121	0.00121	0.0012	0.0012	0.00179	0.00226
0.3	0.00085	0.00086	0.00118	0.00118	0.00119	0.0012	0.00277	0.00301
0.5	0.00076	0.00076	0.00119	0.00119	0.00278	0.00365	0.00439	0.00508
0.7	0.00073	0.00072	0.00175	0.00164	0.01177	0.00963	0.00925	0.00862
0.9	0.00083	0.00081	0.00331	0.00283	0.03439	0.0222	0.01595	0.01233

Table D.3: Average Delay for RTRN=0, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00107	0.00121	0.00121	0.0012	0.0012	0.00185	0.00226
0.3	0.00085	0.00086	0.00118	0.00118	0.00121	0.0012	0.00301	0.00301
0.5	0.00076	0.00076	0.0012	0.00119	0.00366	0.00365	0.00511	0.00508
0.7	0.00073	0.00072	0.00203	0.00164	0.01438	0.00963	0.01092	0.00862
0.9	0.00091	0.00081	0.00369	0.00283	0.04228	0.0222	0.01823	0.01233

Average Delay for RTRN=0 and RL=6

Table D.4: Average Delay for RTRN=0, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00107	0.00105	0.00121	0.00121	0.0012	0.00121	0.00186	0.0027
0.3	0.00085	0.00085	0.00119	0.00119	0.00133	0.00231	0.00263	0.00513
0.5	0.00075	0.00076	0.00127	0.00139	0.00388	0.00748	0.00431	0.00755
0.7	0.00072	0.00073	0.00182	0.00261	0.01099	0.01147	0.00874	0.00999
0.9	0.00079	0.00087	0.00287	0.00291	0.02952	0.02447	0.01476	0.01326

Table D.5: Average Delay for RTRN=0, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00185	0.0027
0.3	0.00085	0.00085	0.00119	0.00119	0.00148	0.00231	0.0032	0.00513
0.5	0.00075	0.00076	0.00132	0.00139	0.00572	0.00748	0.00558	0.00755
0.7	0.00073	0.00073	0.00226	0.00261	0.01343	0.01147	0.01009	0.00999
0.9	0.00084	0.00087	0.00308	0.00291	0.03145	0.02447	0.01462	0.01326

Table D.6: Average Delay for RTRN=0, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.0019	0.0027
0.3	0.00085	0.00085	0.00119	0.00119	0.00166	0.00231	0.00399	0.00513
0.5	0.00076	0.00076	0.00139	0.00139	0.00802	0.00748	0.00677	0.00755
0.7	0.00073	0.00073	0.00276	0.00261	0.01619	0.01147	0.01296	0.00999
0.9	0.0009	0.00087	0.00331	0.00291	0.03595	0.02447	0.01684	0.01326

Average Delay for RTRN=3 and RL=4

Table D.7: Average Delay for RTRN=3, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00106	0.00121	0.00121	0.0012	0.0012	0.00182	0.00225
0.3	0.00086	0.00085	0.00118	0.00118	0.00123	0.00121	0.00304	0.00332
0.5	0.00076	0.00076	0.00119	0.00119	0.00449	0.00373	0.00575	0.00539
0.7	0.00072	0.00073	0.00152	0.00167	0.01114	0.00973	0.00936	0.00842
0.9	0.00078	0.00084	0.00303	0.00284	0.03154	0.02498	0.01482	0.01301

Table D.8: Average Delay for RTRN=3, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00106	0.00121	0.00121	0.0012	0.0012	0.00183	0.00225
0.3	0.00085	0.00085	0.00118	0.00118	0.00125	0.00121	0.00337	0.00332
0.5	0.00076	0.00076	0.00119	0.00119	0.00506	0.00373	0.00655	0.00539
0.7	0.00072	0.00073	0.00176	0.00167	0.01332	0.00973	0.01037	0.00842
0.9	0.00085	0.00084	0.00334	0.00284	0.03726	0.02498	0.01662	0.01301

Table D.9: Average Delay for RTRN=3, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00121	0.00121	0.0012	0.0012	0.00184	0.00225	0.00106
0.3	0.00085	0.00118	0.00118	0.00124	0.00121	0.00329	0.00332	0.00085
0.5	0.00076	0.0012	0.00119	0.00559	0.00373	0.00625	0.00539	0.00076
0.7	0.00073	0.00203	0.00167	0.01593	0.00973	0.01259	0.00842	0.00073
0.9	0.00084	0.00365	0.00284	0.0463	0.02498	0.01917	0.01301	0.00084

Average Delay for RTRN=3 and RL=6

Table D.10: Average Delay for RTRN=3, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00105	0.00121	0.00121	0.0012	0.00121	0.00191	0.00249
0.3	0.00085	0.00085	0.00119	0.00119	0.00204	0.00243	0.00388	0.00516
0.5	0.00075	0.00076	0.00128	0.0014	0.00799	0.00775	0.00768	0.00732
0.7	0.00072	0.00073	0.00186	0.00257	0.01304	0.01154	0.01087	0.00994
0.9	0.00078	0.00087	0.00284	0.00284	0.03066	0.02423	0.01482	0.01253

Table D.11: Average Delay for RTRN=3, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00196	0.00249
0.3	0.00084	0.00085	0.00119	0.00119	0.0024	0.00243	0.00465	0.00516
0.5	0.00076	0.00076	0.00133	0.0014	0.00894	0.00775	0.00836	0.00732
0.7	0.00073	0.00073	0.0024	0.00257	0.01554	0.01154	0.01175	0.00994
0.9	0.00084	0.00087	0.00309	0.00284	0.03397	0.02423	0.01652	0.01253

Table D.12: Average Delay for RTRN=3, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00193	0.00249
0.3	0.00085	0.00085	0.00119	0.00119	0.0025	0.00243	0.00519	0.00516
0.5	0.00076	0.00076	0.00138	0.0014	0.01043	0.00775	0.00988	0.00732
0.7	0.00074	0.00073	0.00275	0.00257	0.01656	0.01154	0.01302	0.00994
0.9	0.00092	0.00087	0.00337	0.00284	0.04128	0.02423	0.01784	0.01253

Average Delay for RTRN=5 and RL=4

Table D.13: Average Delay for RTRN=5, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00106	0.00121	0.00121	0.0012	0.0012	0.00182	0.00225
0.3	0.00086	0.00085	0.00118	0.00118	0.00124	0.00121	0.0032	0.00332
0.5	0.00076	0.00076	0.00119	0.00119	0.00452	0.00373	0.00557	0.00539
0.7	0.00072	0.00073	0.00151	0.00167	0.01169	0.00973	0.00969	0.00842
0.9	0.00077	0.00084	0.00299	0.00284	0.03138	0.02498	0.01497	0.01301

Table D.14: Average Delay for RTRN=5, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00106	0.00121	0.00121	0.0012	0.0012	0.00183	0.00225
0.3	0.00085	0.00085	0.00118	0.00118	0.00126	0.00121	0.00342	0.00332
0.5	0.00076	0.00076	0.00119	0.00119	0.00507	0.00373	0.00638	0.00539
0.7	0.00072	0.00073	0.00187	0.00167	0.01444	0.00973	0.01169	0.00842
0.9	0.00085	0.00084	0.00334	0.00284	0.03726	0.02498	0.01662	0.01301

Table D.15: Average Delay for RTRN=5, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00106	0.00121	0.00121	0.0012	0.0012	0.00184	0.00225
0.3	0.00084	0.00085	0.00118	0.00118	0.00124	0.00121	0.00335	0.00332
0.5	0.00076	0.00076	0.0012	0.00119	0.00562	0.00373	0.00629	0.00539
0.7	0.00073	0.00073	0.00201	0.00167	0.01545	0.00973	0.01221	0.00842
0.9	0.00093	0.00084	0.00365	0.00284	0.0463	0.02498	0.01917	0.01301

Average Delay for RTRN=5 and RL=6

Table D.16: Average Delay for RTRN=5, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00105	0.00121	0.00121	0.0012	0.00121	0.00191	0.00249
0.3	0.00085	0.00085	0.00119	0.00119	0.00197	0.00243	0.00402	0.00516
0.5	0.00076	0.00076	0.00127	0.0014	0.00773	0.00775	0.00757	0.00732
0.7	0.00072	0.00073	0.00191	0.00257	0.013	0.01154	0.01091	0.00994
0.9	0.00078	0.00087	0.00282	0.00284	0.03	0.02423	0.01405	0.01253

Table D.17: Average Delay for RTRN=5, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00196	0.00249
0.3	0.00084	0.00085	0.00119	0.00119	0.00226	0.00243	0.00485	0.00516
0.5	0.00075	0.00076	0.00133	0.0014	0.00932	0.00775	0.00909	0.00732
0.7	0.00073	0.00073	0.00251	0.00257	0.0165	0.01154	0.01229	0.00994
0.9	0.00084	0.00087	0.00309	0.00284	0.03486	0.02423	0.01627	0.01253

Table D.18: Average Delay for RTRN=5, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00193	0.00249
0.3	0.00085	0.00085	0.00119	0.00119	0.0025	0.00243	0.00492	0.00516
0.5	0.00076	0.00076	0.00139	0.0014	0.0103	0.00775	0.00988	0.00732
0.7	0.00073	0.00073	0.00274	0.00257	0.0166	0.01154	0.01292	0.00994
0.9	0.00093	0.00087	0.00339	0.00284	0.04199	0.02423	0.01814	0.01253

Average Delay for RTRN=7 and RL=4

Table D.19: Average Delay for RTRN=7, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00106	0.00121	0.00121	0.0012	0.0012	0.00182	0.00225
0.3	0.00086	0.00085	0.00118	0.00118	0.00124	0.00121	0.0032	0.00332
0.5	0.00076	0.00076	0.00119	0.00119	0.00429	0.00373	0.00551	0.00539
0.7	0.00072	0.00073	0.00152	0.00167	0.01139	0.00973	0.00898	0.00842
0.9	0.00077	0.00084	0.00299	0.00284	0.03138	0.02498	0.01497	0.01301

Table D.20: Average Delay for RTRN=7, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00106	0.00121	0.00121	0.0012	0.0012	0.00183	0.00225
0.3	0.00085	0.00085	0.00118	0.00118	0.00126	0.00121	0.00342	0.00332
0.5	0.00076	0.00076	0.00119	0.00119	0.00497	0.00373	0.00624	0.00539
0.7	0.00072	0.00073	0.00187	0.00167	0.01444	0.00973	0.01169	0.00842
0.9	0.00085	0.00084	0.00334	0.00284	0.03726	0.02498	0.01662	0.01301

Table D.21: Average Delay for RTRN=7, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00106	0.00121	0.00121	0.0012	0.0012	0.00184	0.00225
0.3	0.00084	0.00085	0.00118	0.00118	0.00124	0.00121	0.00335	0.00332
0.5	0.00077	0.00076	0.0012	0.00119	0.00543	0.00373	0.00621	0.00539
0.7	0.00073	0.00073	0.00201	0.00167	0.01545	0.00973	0.01221	0.00842
0.9	0.00093	0.00084	0.00365	0.00284	0.0463	0.02498	0.01917	0.01301

Average Delay for RTRN=7 and RL=6

Table D.22: Average Delay for RTRN=7, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00105	0.00105	0.00121	0.00121	0.0012	0.00121	0.00191	0.00249
0.3	0.00085	0.00085	0.00118	0.00119	0.00199	0.00243	0.00409	0.00516
0.5	0.00076	0.00076	0.00129	0.0014	0.00844	0.00775	0.00821	0.00732
0.7	0.00072	0.00073	0.00191	0.00257	0.01325	0.01154	0.0109	0.00994
0.9	0.00079	0.00087	0.00285	0.00284	0.03144	0.02423	0.01457	0.01253

Table D.23: Average Delay for RTRN=7, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00196	0.00249
0.3	0.00084	0.00085	0.00119	0.00119	0.00234	0.00243	0.00498	0.00516
0.5	0.00075	0.00076	0.00133	0.0014	0.00912	0.00775	0.0087	0.00732
0.7	0.00073	0.00073	0.00247	0.00257	0.01606	0.01154	0.01209	0.00994
0.9	0.00084	0.00087	0.00309	0.00284	0.03486	0.02423	0.01627	0.01253

Table D.24: Average Delay for RTRN=7, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00106	0.00105	0.00121	0.00121	0.0012	0.00121	0.00193	0.00249
0.3	0.00085	0.00085	0.00119	0.00119	0.0025	0.00243	0.00492	0.00516
0.5	0.00076	0.00076	0.00139	0.0014	0.01042	0.00775	0.00999	0.00732
0.7	0.00073	0.00073	0.00274	0.00257	0.0166	0.01154	0.01292	0.00994
0.9	0.00093	0.00087	0.00339	0.00284	0.04199	0.02423	0.01814	0.01253

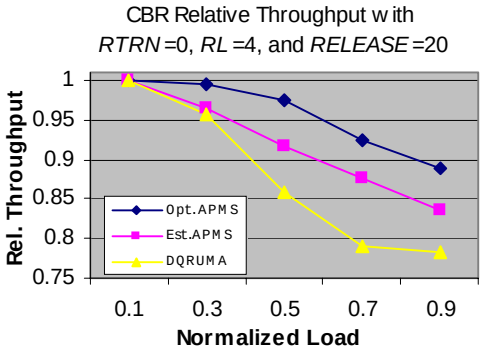
Average Delay for DQRUMA (RTRN=0 and RL=1)

Table D.25: Average Delay for DQRUMA (RTRN=0 and RL=1)

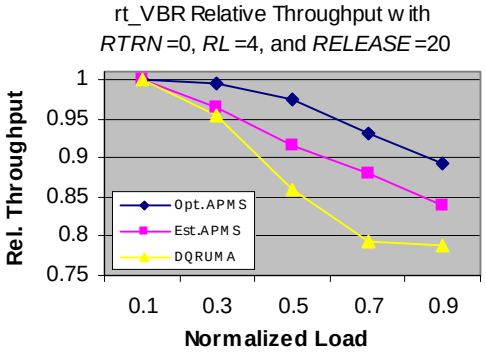
	Traffic Type			
Load	CBR	rt-VBR	nrt-VBR	ABR
0.1	0.00094	0.0011	0.00132	0.0018
0.3	0.00074	0.00101	0.00132	0.00206
0.5	0.00069	0.00099	0.00132	0.00225
0.7	0.00068	0.00099	0.00132	0.0026
0.9	0.00067	0.00102	0.0014	0.00356

Appendix E: Relative Throughput Plots and t-Test Results for Method 2 (Varying the Number of MUs)

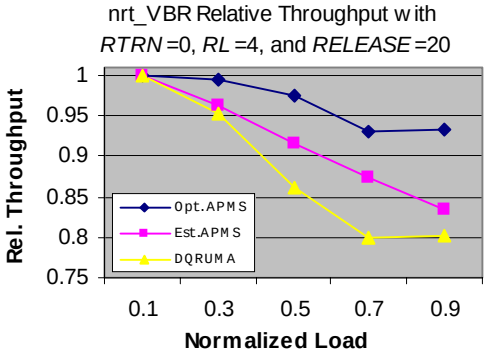
This appendix contains the relative throughput plots and their corresponding t-test results for Method 2 case of determining the normalized load by varying the number of MUs that are active.



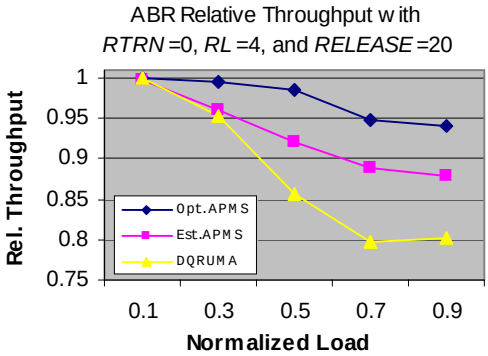
	Est. APMS	*DQRUMA*
Mean	0.91833288	0.87770944
Variance	0.004356709	0.00942216
Observations	5	5
Pearson Correlation	0.976600354	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.554258118	
P(T<=t) one-tail	0.031510133	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.91954504	0.8785806
Variance	0.004055245	0.009061623
Observations	5	5
Pearson Correlation	0.976783245	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.56582879	
P(T<=t) one-tail	0.031126409	
t Critical one-tail	2.131846486	

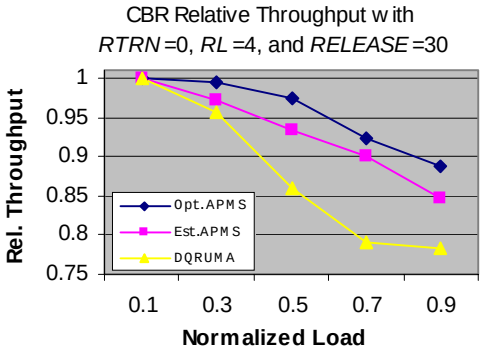


	Est. APMS	*DQRUMA*
Mean	0.91722908	0.88310364
Variance	0.004400056	0.008136462
Observations	5	5
Pearson Correlation	0.967694045	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.467490362	
P(T<=t) one-tail	0.034566394	
t Critical one-tail	2.131846486	

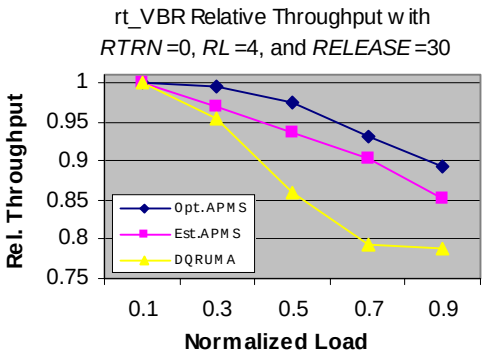


	Est. APMS	*DQRUMA*
Mean	0.9289194	0.88163356
Variance	0.002509329	0.008280526
Observations	5	5
Pearson Correlation	0.990003758	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.517280667	
P(T<=t) one-tail	0.032773221	
t Critical one-tail	2.131846486	

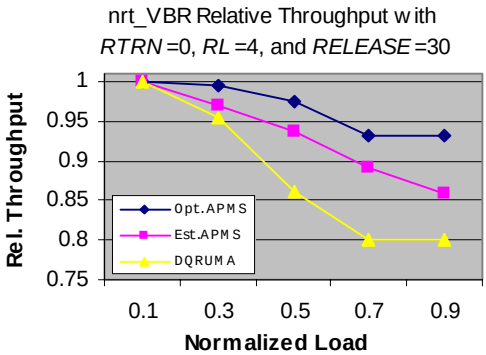
Figure E.1: Relative Throughput with RTRN=0, RL=4, and RELEASE=20



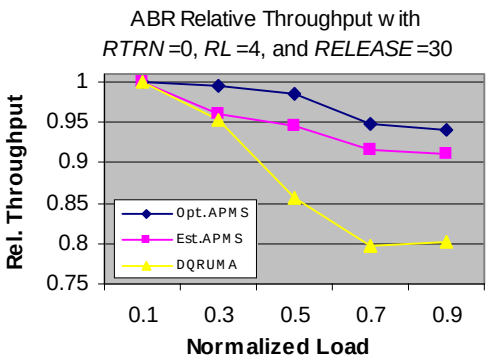
	Est. APMS	*DQRUMA*
Mean	0.93013616	0.87770944
Variance	0.003543346	0.00942216
Observations	5	5
Pearson Correlation	0.949160602	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.623374443	
P(T<=t) one-tail	0.029295819	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93224572	0.8785806
Variance	0.003253033	0.009061623
Observations	5	5
Pearson Correlation	0.944558062	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.64517654	
P(T<=t) one-tail	0.028634808	
t Critical one-tail	2.131846486	

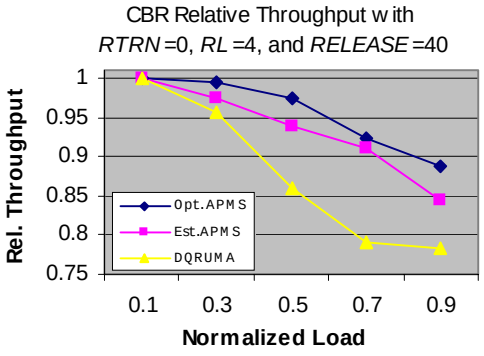


	Est. APMS	*DQRUMA*
Mean	0.9307546	0.88310364
Variance	0.003309897	0.008136462
Observations	5	5
Pearson Correlation	0.959207552	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.759646891	
P(T<=t) one-tail	0.025434023	
t Critical one-tail	2.131846486	

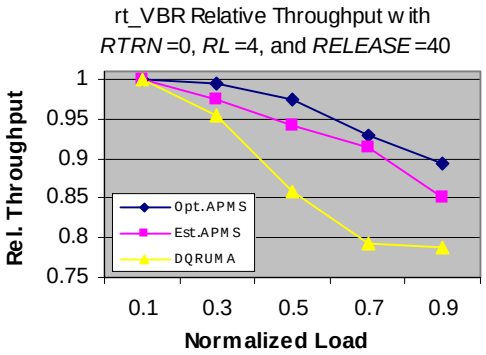


	Est. APMS	*DQRUMA*
Mean	0.94614072	0.88163356
Variance	0.001321811	0.008280526
Observations	5	5
Pearson Correlation	0.971701521	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.560756925	
P(T<=t) one-tail	0.031293947	
t Critical one-tail	2.131846486	

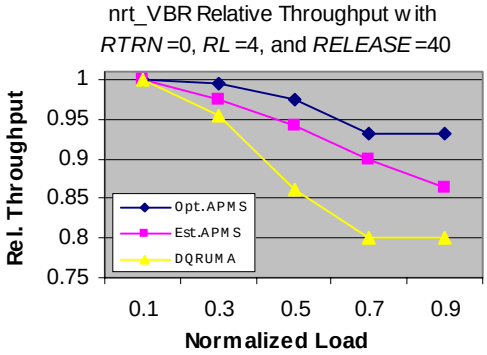
Figure E.2: Relative Throughput with RTRN=0, RL=4, and RELEASE=30



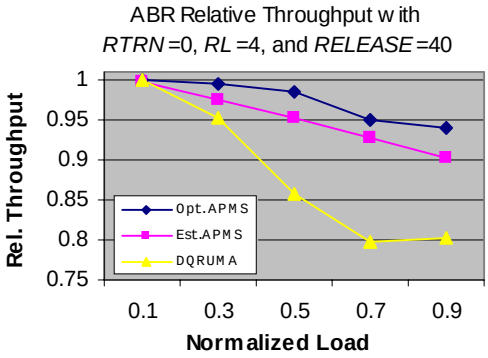
	Est. APMS	*DQRUMA*
Mean	0.933001	0.87770944
Variance	0.003626098	0.00942216
Observations	5	5
Pearson Correlation	0.915686609	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.553877017	
P(T<=t) one-tail	0.031522864	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93599692	0.8785806
Variance	0.003327036	0.009061623
Observations	5	5
Pearson Correlation	0.915191508	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.65493184	
P(T<=t) one-tail	0.028344609	
t Critical one-tail	2.131846486	

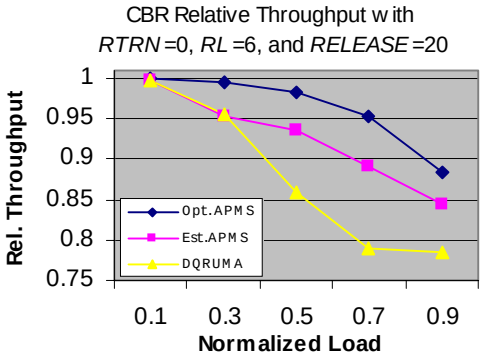


	Est. APMS	*DQRUMA*
Mean	0.93550608	0.88310364
Variance	0.003030571	0.008136462
Observations	5	5
Pearson Correlation	0.952134987	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.832763242	
P(T<=t) one-tail	0.023606283	
t Critical one-tail	2.131846486	

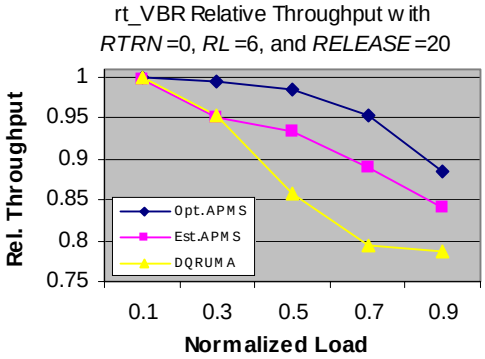


	Est. APMS	*DQRUMA*
Mean	0.95039852	0.88163356
Variance	0.001429197	0.008280526
Observations	5	5
Pearson Correlation	0.95251722	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.736989032	
P(T<=t) one-tail	0.026033336	
t Critical one-tail	2.131846486	

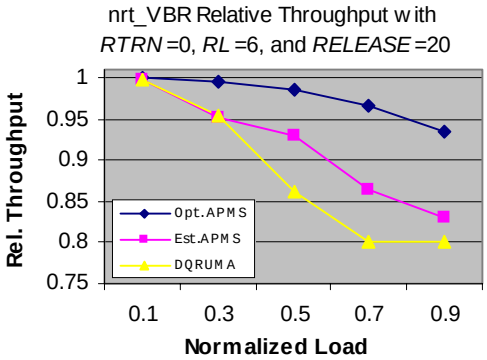
Figure E.3: Relative Throughput with RTRN=0, RL=4, and RELEASE=40



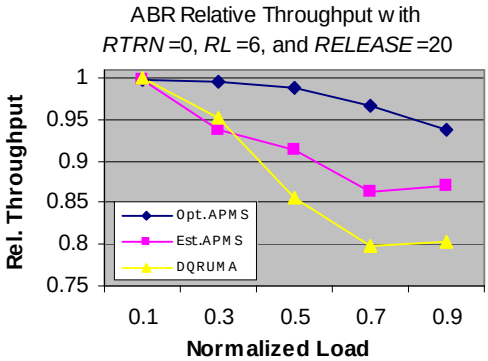
	Est. APMS	*DQRUMA*
Mean	0.9245448	0.87770944
Variance	0.003517621	0.00942216
Observations	5	5
Pearson Correlation	0.933079364	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.234709805	
P(T<=t) one-tail	0.044571798	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.92284084	0.8785806
Variance	0.003586213	0.009061623
Observations	5	5
Pearson Correlation	0.934092023	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.21408916	
P(T<=t) one-tail	0.045605003	
t Critical one-tail	2.131846486	

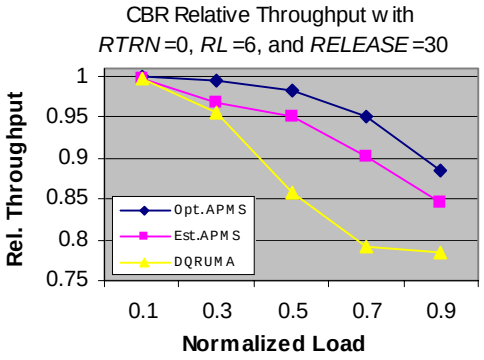


	Est. APMS	*DQRUMA*
Mean	0.91465856	0.88310364
Variance	0.004570441	0.008136462
Observations	5	5
Pearson Correlation	0.946750694	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.071609748	
P(T<=t) one-tail	0.053517402	
t Critical one-tail	2.131846486	

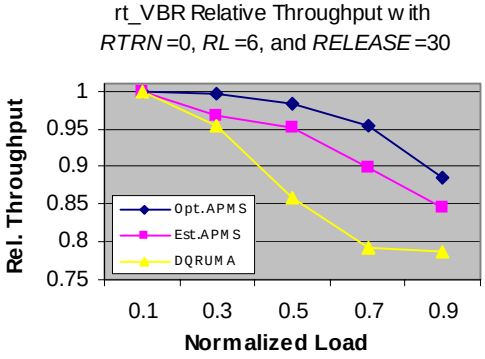


	Est. APMS	*DQRUMA*
Mean	0.9165312	0.88163356
Variance	0.002999785	0.008280526
Observations	5	5
Pearson Correlation	0.971747071	
Hypothesized Mean Difference	0	
df	4	
t Stat	1.954489645	
P(T<=t) one-tail	0.061162661	
t Critical one-tail	2.131846486	

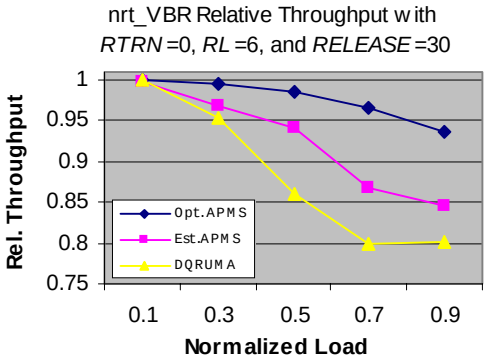
Figure E.4: Relative Throughput with RTRN=0, RL=6, and RELEASE=20



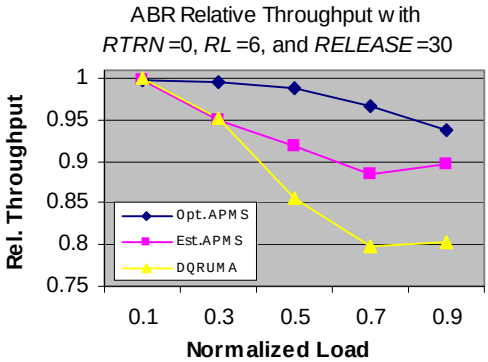
	Est. APMS	*DQRUMA*
Mean	0.93296324	0.87770944
Variance	0.003571503	0.00942216
Observations	5	5
Pearson Correlation	0.913260737	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.522997321	
P(T<=t) one-tail	0.032574219	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93269544	0.8785806
Variance	0.003609725	0.009061623
Observations	5	5
Pearson Correlation	0.916371586	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.586056524	
P(T<=t) one-tail	0.030468376	
t Critical one-tail	2.131846486	

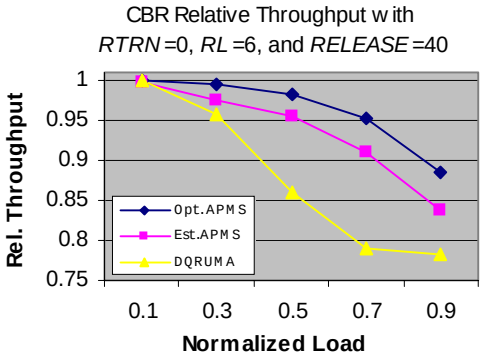


	Est. APMS	*DQRUMA*
Mean	0.92402696	0.88310364
Variance	0.004305667	0.008136462
Observations	5	5
Pearson Correlation	0.95114334	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.660769173	
P(T<=t) one-tail	0.028172581	
t Critical one-tail	2.131846486	

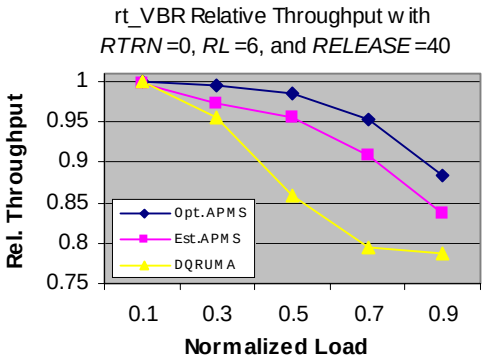


	Est. APMS	*DQRUMA*
Mean	0.92962412	0.88163356
Variance	0.00202135	0.008280526
Observations	5	5
Pearson Correlation	0.977013847	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.233895363	
P(T<=t) one-tail	0.044612106	
t Critical one-tail	2.131846486	

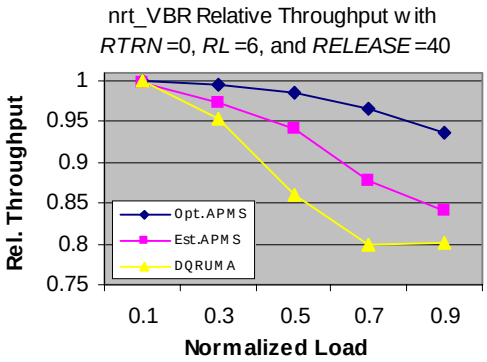
Figure E.5: Relative Throughput with RTRN=0, RL=6, and RELEASE=30



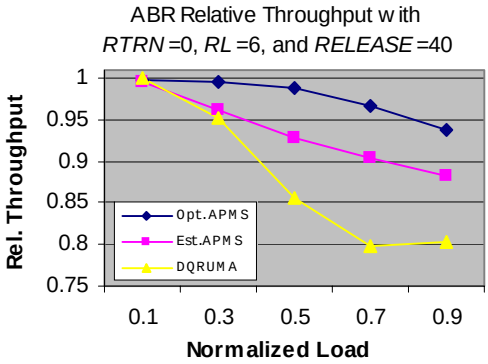
	Est. APMS	*DQRUMA*
Mean	0.93520028	0.87770944
Variance	0.003943172	0.00942216
Observations	5	5
Pearson Correlation	0.88386277	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.525787917	
P(T<=t) one-tail	0.032477578	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93428384	0.8785806
Variance	0.004037745	0.009061623
Observations	5	5
Pearson Correlation	0.880999573	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.520891304	
P(T<=t) one-tail	0.03264737	
t Critical one-tail	2.131846486	

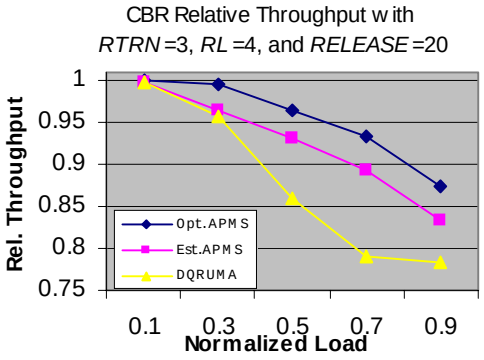


	Est. APMS	*DQRUMA*
Mean	0.92581864	0.88310364
Variance	0.004332208	0.008136462
Observations	5	5
Pearson Correlation	0.944402965	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.696480145	
P(T<=t) one-tail	0.02714599	
t Critical one-tail	2.131846486	

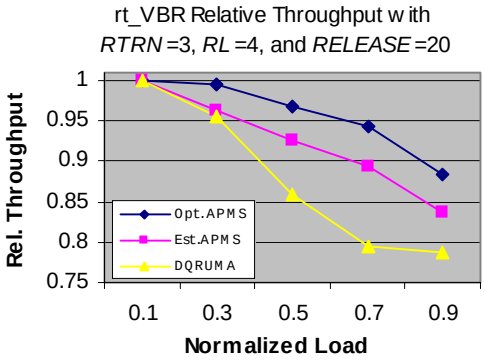


	Est. APMS	*DQRUMA*
Mean	0.93432244	0.88163356
Variance	0.001998798	0.008280526
Observations	5	5
Pearson Correlation	0.97861958	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.447772021	
P(T<=t) one-tail	0.03530725	
t Critical one-tail	2.131846486	

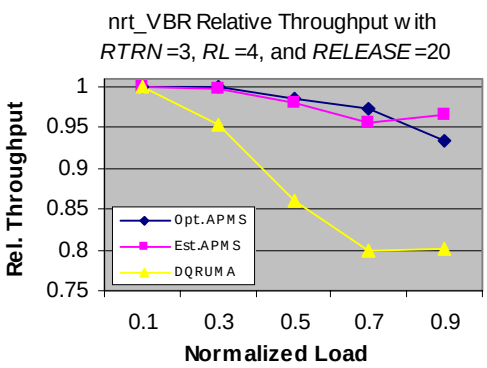
Figure E.6: Relative Throughput with RTRN=0, RL=6, and RELEASE=40



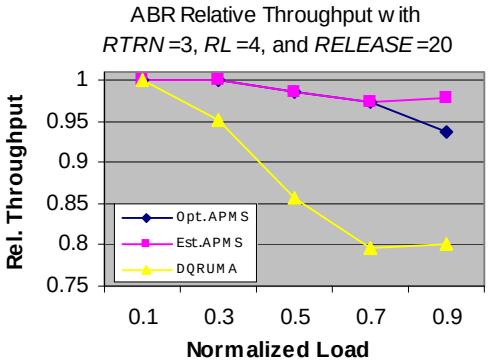
	Est. APMS	*DQRUMA*
Mean	0.92373156	0.87770944
Variance	0.004148071	0.00942216
Observations	5	5
Pearson Correlation	0.939035608	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.406214561	
P(T<=t) one-tail	0.036928526	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.923742	0.8785806
Variance	0.003874468	0.009061623
Observations	5	5
Pearson Correlation	0.941699039	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.395955989	
P(T<=t) one-tail	0.037341638	
t Critical one-tail	2.131846486	

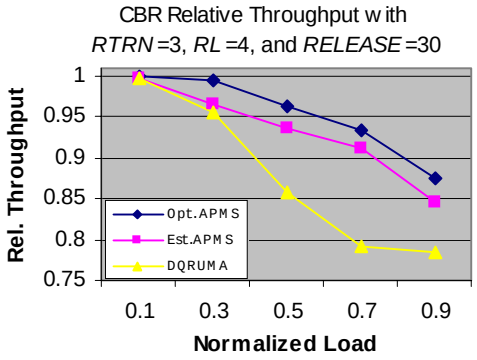


	Est. APMS	*DQRUMA*
Mean	0.98017636	0.88310364
Variance	0.000371301	0.008136462
Observations	5	5
Pearson Correlation	0.961584322	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.020262366	
P(T<=t) one-tail	0.019576572	
t Critical one-tail	2.131846486	

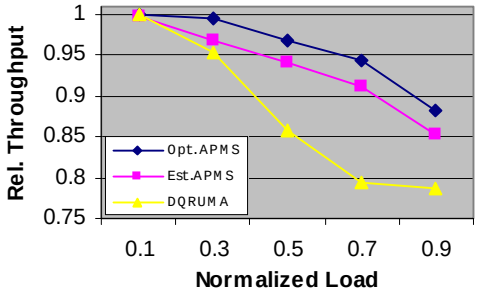


	Est. APMS	*DQRUMA*
Mean	0.98723416	0.88163356
Variance	0.000147329	0.008280526
Observations	5	5
Pearson Correlation	0.974172445	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.980673334	
P(T<=t) one-tail	0.020355873	
t Critical one-tail	2.131846486	

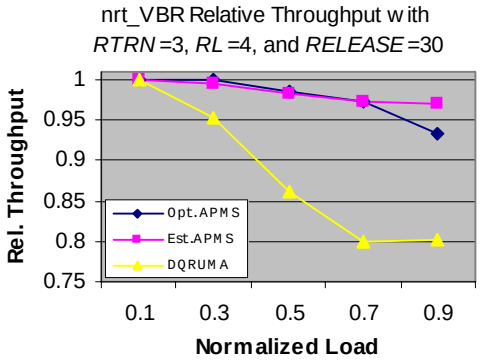
Figure E.7: Relative Throughput with RTRN=3, RL=4, and RELEASE=20



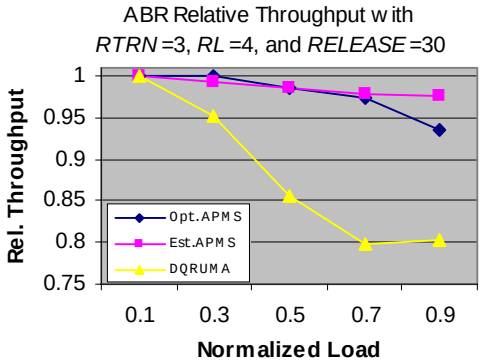
	Est. APMS	*DQRUMA*
Mean	0.93175324	0.87770944
Variance	0.003389532	0.00942216
Observations	5	5
Pearson Correlation	0.910204368	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.405349461	
P(T<=t) one-tail	0.036963163	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93434924	0.8785806
Variance	0.00311485	0.009061623
Observations	5	5
Pearson Correlation	0.915231845	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.51853384	
P(T<=t) one-tail	0.032729478	
t Critical one-tail	2.131846486	

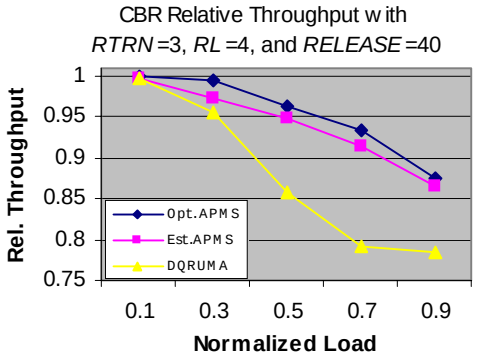


	Est. APMS	*DQRUMA*
Mean	0.98397548	0.88310364
Variance	0.000179252	0.008136462
Observations	5	5
Pearson Correlation	0.993721824	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.932635803	
P(T<=t) one-tail	0.021350698	
t Critical one-tail	2.131846486	

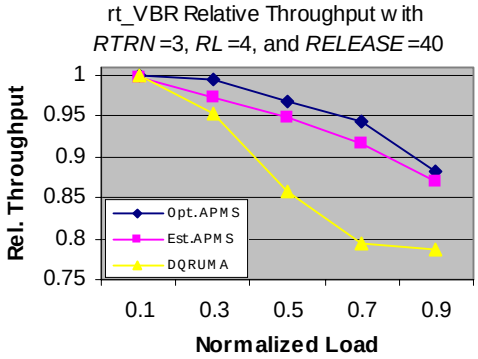


	Est. APMS	*DQRUMA*
Mean	0.98667548	0.88163356
Variance	0.000102711	0.008280526
Observations	5	5
Pearson Correlation	0.992680376	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.901691025	
P(T<=t) one-tail	0.022021609	
t Critical one-tail	2.131846486	

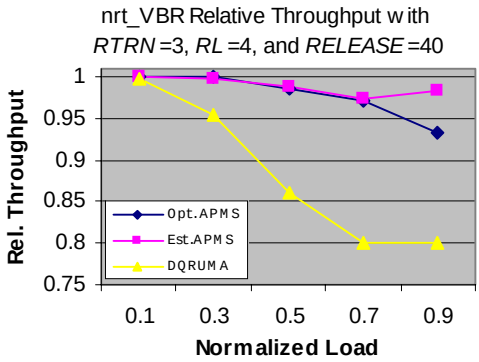
Figure E.8: Relative Throughput with RTRN=3, RL=4, and RELEASE=30



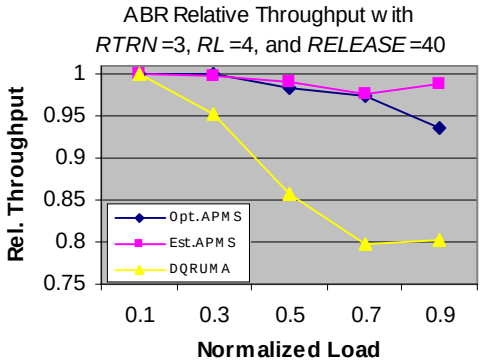
	Est. APMS	*DQRUMA*
Mean	0.9400374	0.87770944
Variance	0.002726722	0.00942216
Observations	5	5
Pearson Correlation	0.935430319	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.699181447	
P(T<=t) one-tail	0.027070107	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.94180348	0.8785806
Variance	0.002523735	0.009061623
Observations	5	5
Pearson Correlation	0.934052695	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.745313234	
P(T<=t) one-tail	0.025811277	
t Critical one-tail	2.131846486	

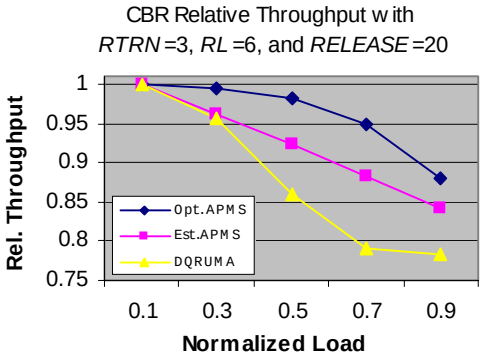


	Est. APMS	*DQRUMA*
Mean	0.98865416	0.88310364
Variance	0.000122813	0.008136462
Observations	5	5
Pearson Correlation	0.927379066	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.949029204	
P(T<=t) one-tail	0.021004949	
t Critical one-tail	2.131846486	

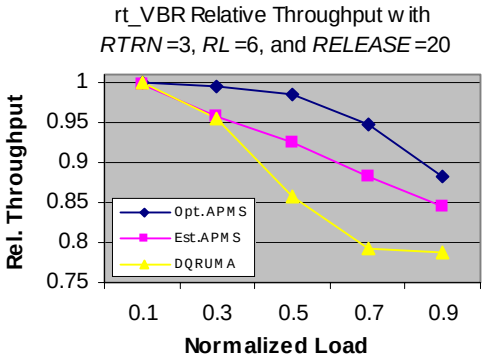


	Est. APMS	*DQRUMA*
Mean	0.99110312	0.88163356
Variance	8.15053E-05	0.008280526
Observations	5	5
Pearson Correlation	0.896724309	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.949244878	
P(T<=t) one-tail	0.021000444	
t Critical one-tail	2.131846486	

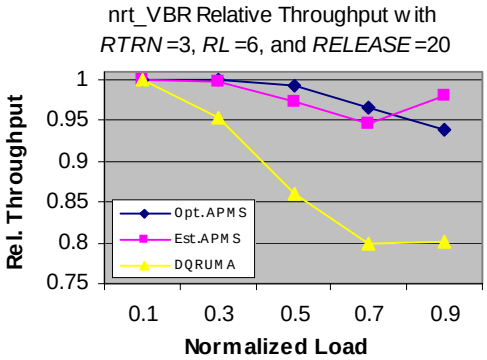
Figure E.9: Relative Throughput with RTRN=3, RL=4, and RELEASE=40



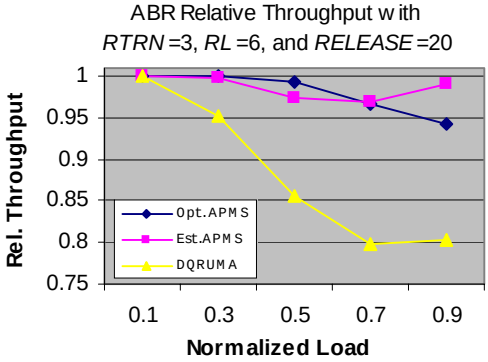
	Est. APMS	*DQRUMA*
Mean	0.92118008	0.87770944
Variance	0.003878944	0.00942216
Observations	5	5
Pearson Correlation	0.967107477	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.42417028	
P(T<=t) one-tail	0.036217867	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.921776	0.8785806
Variance	0.003685053	0.009061623
Observations	5	5
Pearson Correlation	0.966054434	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.428598556	
P(T<=t) one-tail	0.036044998	
t Critical one-tail	2.131846486	

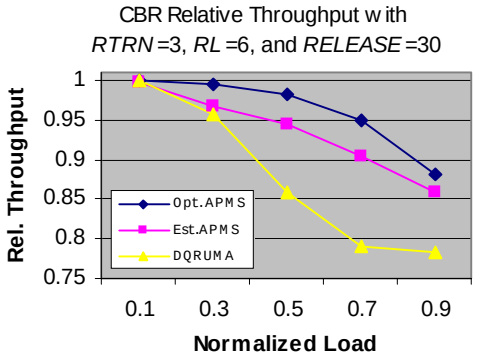


	Est. APMS	*DQRUMA*
Mean	0.97935096	0.88310364
Variance	0.000484278	0.008136462
Observations	5	5
Pearson Correlation	0.811362639	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.928822189	
P(T<=t) one-tail	0.021432079	
t Critical one-tail	2.131846486	

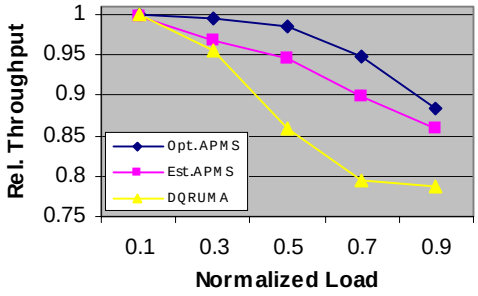


	Est. APMS	*DQRUMA*
Mean	0.98610296	0.88163356
Variance	0.000200823	0.008280526
Observations	5	5
Pearson Correlation	0.7410637	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.881964781	
P(T<=t) one-tail	0.022462119	
t Critical one-tail	2.131846486	

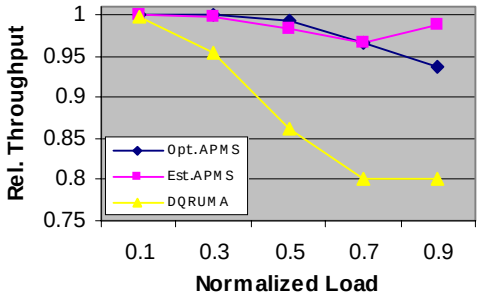
Figure E.10: Relative Throughput with RTRN=3, RL=6, and RELEASE=20



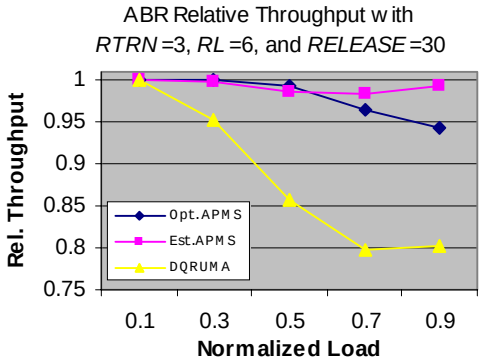
	Est. APMS	*DQRUMA*
Mean	0.93465576	0.87770944
Variance	0.002980402	0.00942216
Observations	5	5
Pearson Correlation	0.939964356	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.577632942	
P(T<=t) one-tail	0.030740449	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93386384	0.8785806
Variance	0.003007919	0.009061623
Observations	5	5
Pearson Correlation	0.945678215	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.638422642	
P(T<=t) one-tail	0.028837726	
t Critical one-tail	2.131846486	

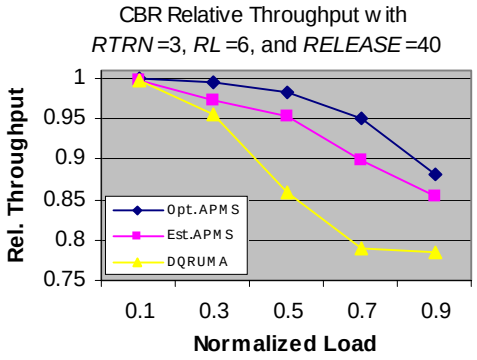


	Est. APMS	*DQRUMA*
Mean	0.9874312	0.88310364
Variance	0.000169655	0.008136462
Observations	5	5
Pearson Correlation	0.811746374	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.916346338	
P(T<=t) one-tail	0.02170085	
t Critical one-tail	2.131846486	

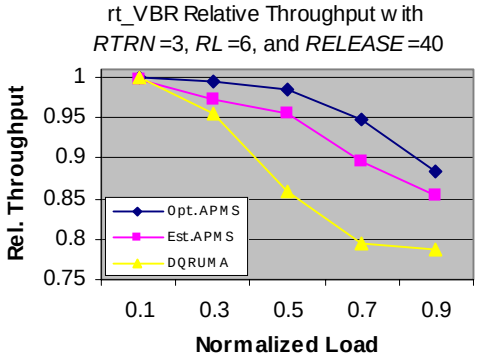


	Est. APMS	*DQRUMA*
Mean	0.99229216	0.88163356
Variance	5.08259E-05	0.008280526
Observations	5	5
Pearson Correlation	0.809559046	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.899851464	
P(T<=t) one-tail	0.02206226	
t Critical one-tail	2.131846486	

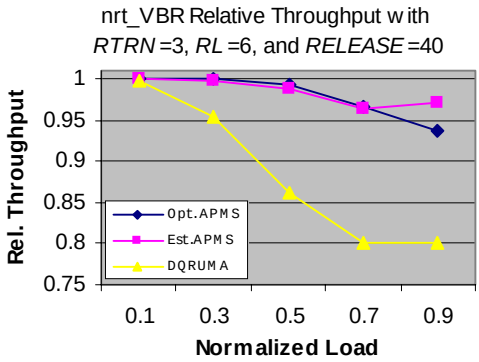
Figure E.11: Relative Throughput with RTRN=3, RL=6, and RELEASE=30



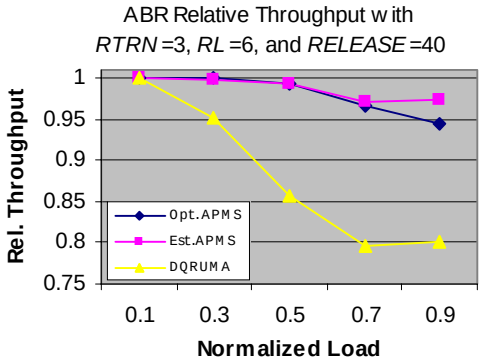
	Est. APMS	*DQRUMA*
Mean	0.93564648	0.87770944
Variance	0.003420678	0.00942216
Observations	5	5
Pearson Correlation	0.929145613	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.705438723	
P(T<=t) one-tail	0.02689527	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93532664	0.8785806
Variance	0.003515388	0.009061623
Observations	5	5
Pearson Correlation	0.923789528	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.737038773	
P(T<=t) one-tail	0.026032002	
t Critical one-tail	2.131846486	

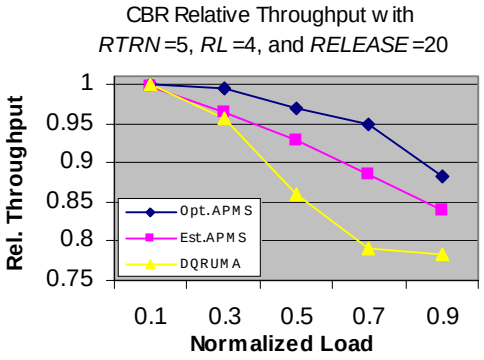


	Est. APMS	*DQRUMA*
Mean	0.9840824	0.88310364
Variance	0.000271005	0.008136462
Observations	5	5
Pearson Correlation	0.937237804	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.010872493	
P(T<=t) one-tail	0.019758203	
t Critical one-tail	2.131846486	

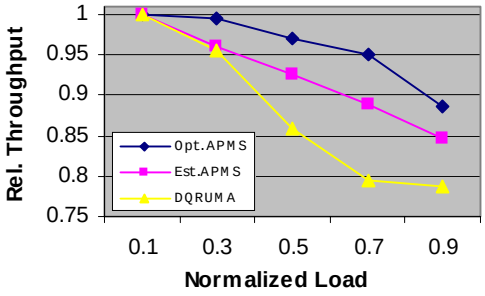


	Est. APMS	*DQRUMA*
Mean	0.98720468	0.88163356
Variance	0.000180977	0.008280526
Observations	5	5
Pearson Correlation	0.928876024	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.001092922	
P(T<=t) one-tail	0.019949474	
t Critical one-tail	2.131846486	

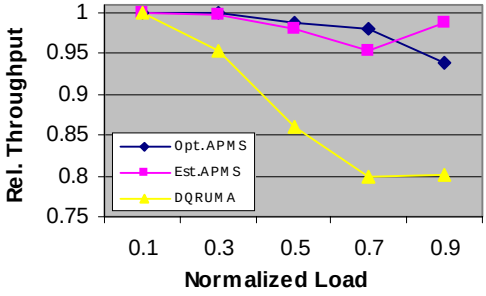
Figure E.12: Relative Throughput with RTRN=3, RL=6, and RELEASE=40



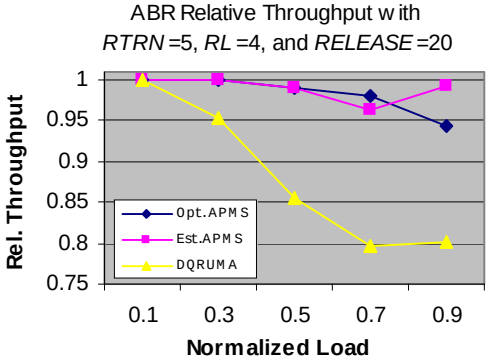
	Est. APMS	*DQRUMA*
Mean	0.92317444	0.87770944
Variance	0.003913413	0.00942216
Observations	5	5
Pearson Correlation	0.961000529	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.491758653	
P(T<=t) one-tail	0.03367877	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.92432196	0.8785806
Variance	0.003585038	0.009061623
Observations	5	5
Pearson Correlation	0.963707304	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.509605331	
P(T<=t) one-tail	0.033042595	
t Critical one-tail	2.131846486	

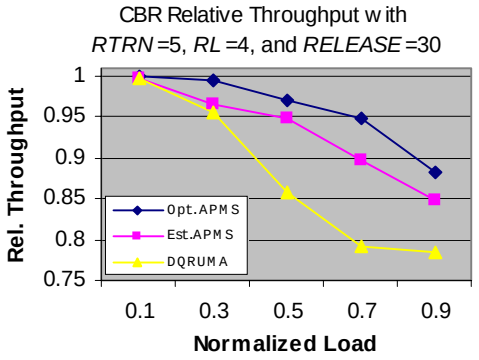


	Est. APMS	*DQRUMA*
Mean	0.98382576	0.88310364
Variance	0.00035611	0.008136462
Observations	5	5
Pearson Correlation	0.753258222	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.925155709	
P(T<=t) one-tail	0.021510662	
t Critical one-tail	2.131846486	

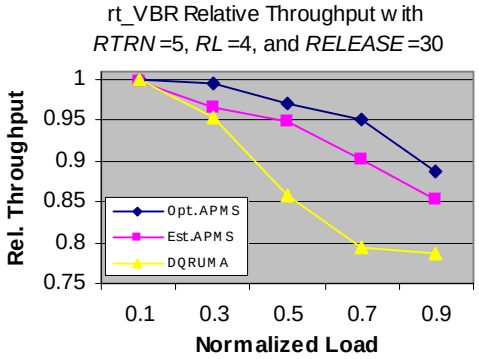


	Est. APMS	*DQRUMA*
Mean	0.9891836	0.88163356
Variance	0.00021559	0.008280526
Observations	5	5
Pearson Correlation	0.71732179	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.964877212	
P(T<=t) one-tail	0.020676903	
t Critical one-tail	2.131846486	

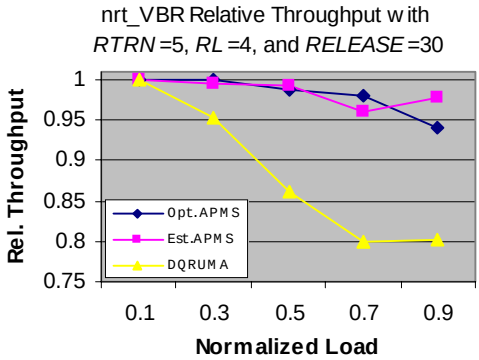
Figure E.13: Relative Throughput with RTRN=5, RL=4, and RELEASE=20



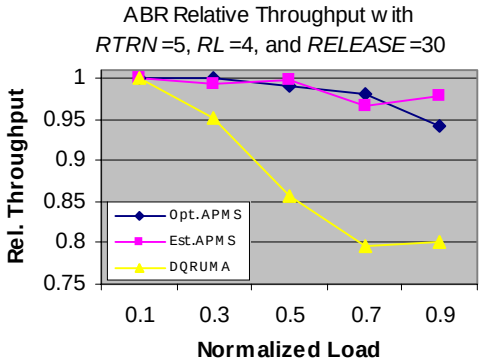
	Est. APMS	*DQRUMA*
Mean	0.93131308	0.87770944
Variance	0.003579584	0.00942216
Observations	5	5
Pearson Correlation	0.928061238	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.542624695	
P(T<=t) one-tail	0.031901409	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93381708	0.8785806
Variance	0.003300514	0.009061623
Observations	5	5
Pearson Correlation	0.924121286	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.601314522	
P(T<=t) one-tail	0.029982563	
t Critical one-tail	2.131846486	

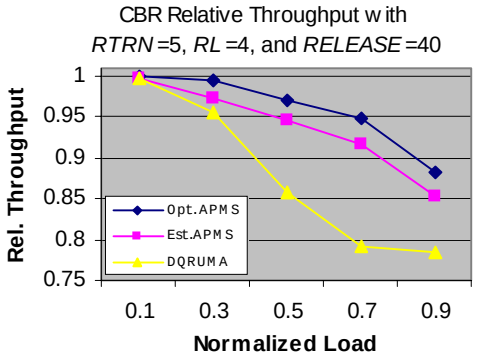


	Est. APMS	*DQRUMA*
Mean	0.98511316	0.88310364
Variance	0.000265789	0.008136462
Observations	5	5
Pearson Correlation	0.846757737	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.966643016	
P(T<=t) one-tail	0.020640724	
t Critical one-tail	2.131846486	

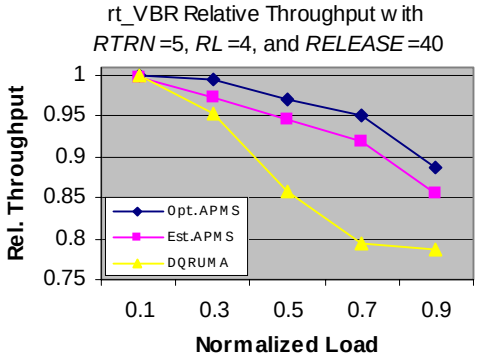


	Est. APMS	*DQRUMA*
Mean	0.98737264	0.88163356
Variance	0.000198173	0.008280526
Observations	5	5
Pearson Correlation	0.794409067	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.94552345	
P(T<=t) one-tail	0.021078335	
t Critical one-tail	2.131846486	

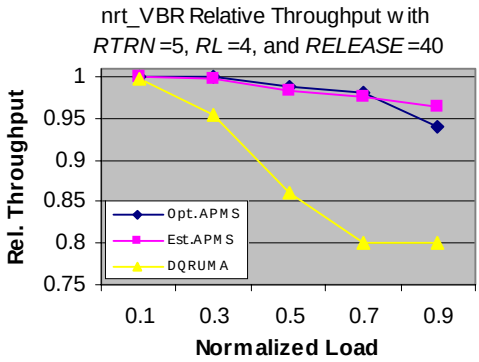
Figure E.14: Relative Throughput with RTRN=5, RL=4, and RELEASE=30



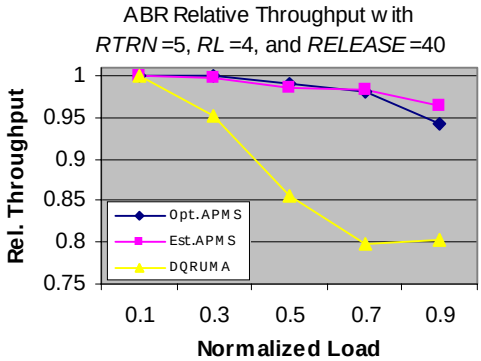
	Est. APMS	*DQRUMA*
Mean	0.93759552	0.87770944
Variance	0.003218123	0.00942216
Observations	5	5
Pearson Correlation	0.906239705	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.596466973	
P(T<=t) one-tail	0.030135939	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93896164	0.8785806
Variance	0.003049961	0.009061623
Observations	5	5
Pearson Correlation	0.904041783	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.644721013	
P(T<=t) one-tail	0.028648443	
t Critical one-tail	2.131846486	

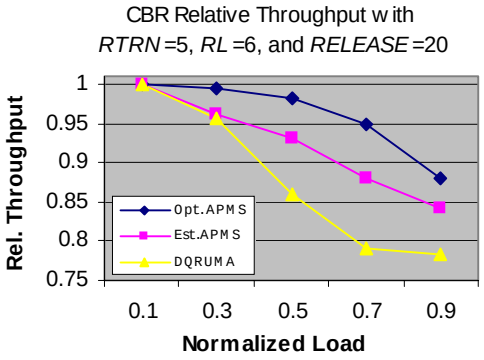


	Est. APMS	*DQRUMA*
Mean	0.98451892	0.88310364
Variance	0.000231278	0.008136462
Observations	5	5
Pearson Correlation	0.94637051	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.985046013	
P(T<=t) one-tail	0.020268037	
t Critical one-tail	2.131846486	

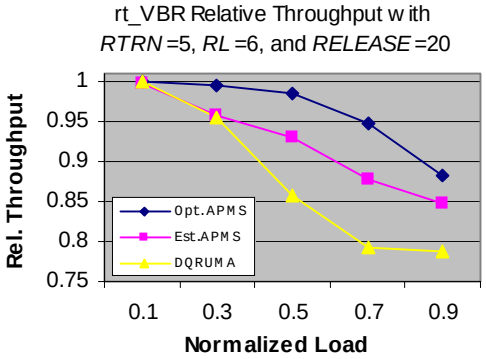


	Est. APMS	*DQRUMA*
Mean	0.98688664	0.88163356
Variance	0.000196876	0.008280526
Observations	5	5
Pearson Correlation	0.862502699	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.971093672	
P(T<=t) one-tail	0.020549863	
t Critical one-tail	2.131846486	

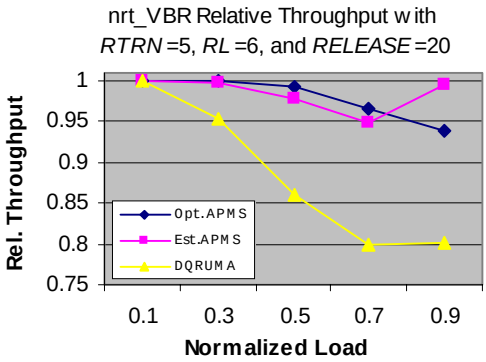
Figure E.15: Relative Throughput with RTRN=5, RL=4, and RELEASE=40



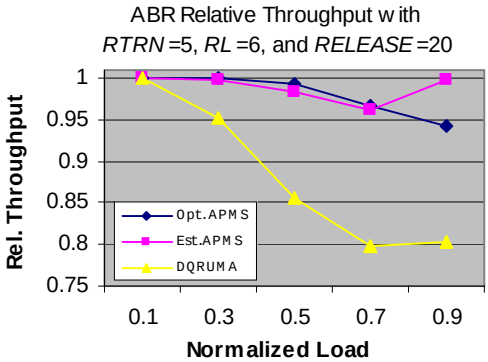
	Est. APMS	*DQRUMA*
Mean	0.92191944	0.87770944
Variance	0.003966511	0.00942216
Observations	5	5
Pearson Correlation	0.963466559	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.464763413	
P(T<=t) one-tail	0.034667789	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.92211076	0.8785806
Variance	0.003714731	0.009061623
Observations	5	5
Pearson Correlation	0.970136363	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.497301621	
P(T<=t) one-tail	0.033479697	
t Critical one-tail	2.131846486	

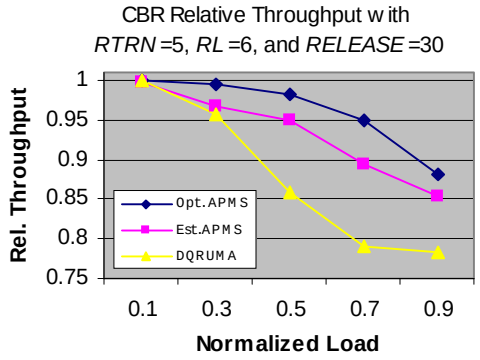


	Est. APMS	*DQRUMA*
Mean	0.98404336	0.88310364
Variance	0.000456541	0.008136462
Observations	5	5
Pearson Correlation	0.636369621	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.880465839	
P(T<=t) one-tail	0.02249601	
t Critical one-tail	2.131846486	

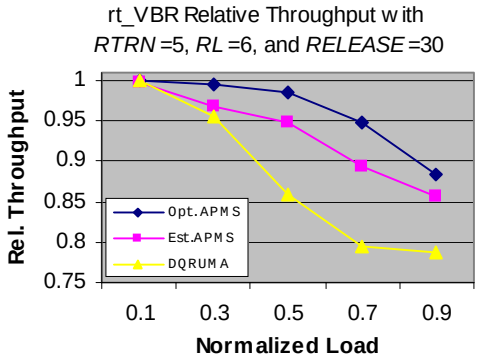


	Est. APMS	*DQRUMA*
Mean	0.98806572	0.88163356
Variance	0.000260257	0.008280526
Observations	5	5
Pearson Correlation	0.608672636	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.895922271	
P(T<=t) one-tail	0.022149382	
t Critical one-tail	2.131846486	

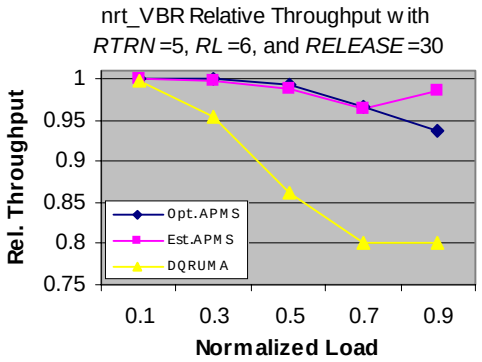
Figure E.16: Relative Throughput with RTRN=5, RL=6, and RELEASE=20



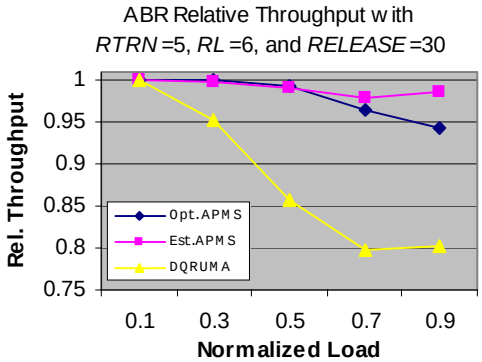
	Est. APMS	*DQRUMA*
Mean	0.93240208	0.87770944
Variance	0.003364943	0.00942216
Observations	5	5
Pearson Correlation	0.940241719	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.60818536	
P(T<=t) one-tail	0.029766705	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93255384	0.8785806
Variance	0.003313638	0.009061623
Observations	5	5
Pearson Correlation	0.939823345	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.649192685	
P(T<=t) one-tail	0.028514924	
t Critical one-tail	2.131846486	

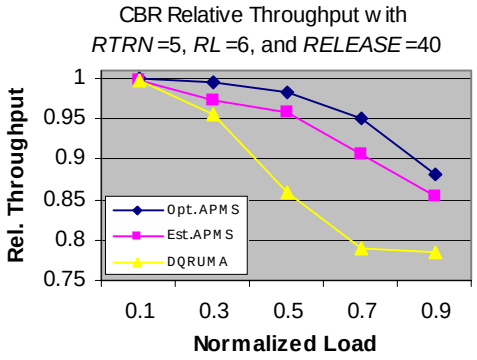


	Est. APMS	*DQRUMA*
Mean	0.98683728	0.88310364
Variance	0.000199496	0.008136462
Observations	5	5
Pearson Correlation	0.842285822	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.948278754	
P(T<=t) one-tail	0.021020633	
t Critical one-tail	2.131846486	

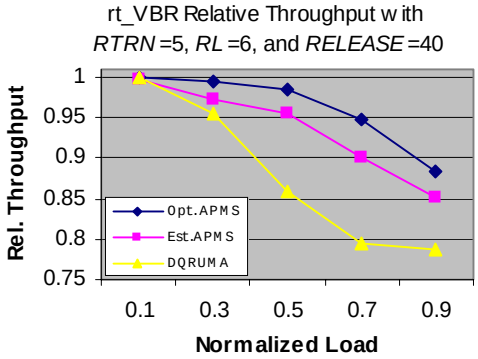


	Est. APMS	*DQRUMA*
Mean	0.9912106	0.88163356
Variance	7.42032E-05	0.008280526
Observations	5	5
Pearson Correlation	0.92256466	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.947920783	
P(T<=t) one-tail	0.02102812	
t Critical one-tail	2.131846486	

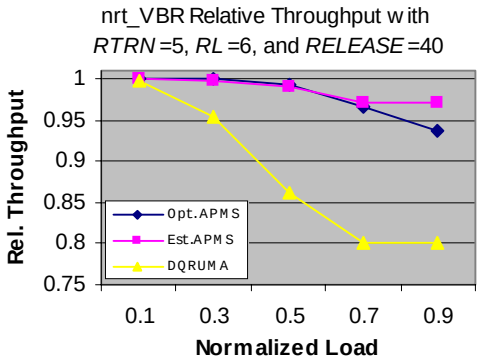
Figure E.17: Relative Throughput with RTRN=5, RL=6, and RELEASE=30



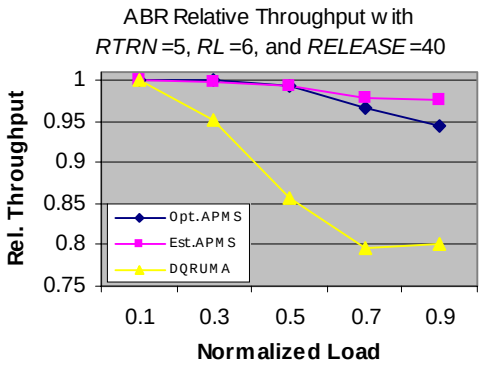
	Est. APMS	*DQRUMA*
Mean	0.93777116	0.87770944
Variance	0.003304189	0.00942216
Observations	5	5
Pearson Correlation	0.913709998	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.670085971	
P(T<=t) one-tail	0.027900497	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93585964	0.8785806
Variance	0.003531625	0.009061623
Observations	5	5
Pearson Correlation	0.914494063	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.702223771	
P(T<=t) one-tail	0.026984937	
t Critical one-tail	2.131846486	

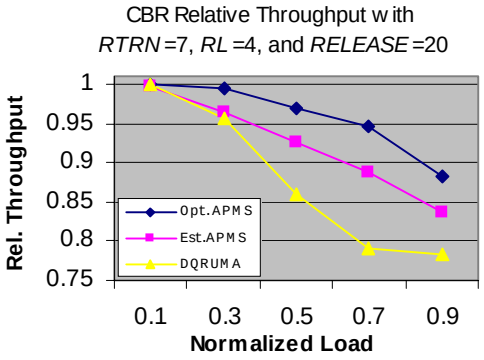


	Est. APMS	*DQRUMA*
Mean	0.98567992	0.88310364
Variance	0.000217413	0.008136462
Observations	5	5
Pearson Correlation	0.944572746	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.001087657	
P(T<=t) one-tail	0.019949578	
t Critical one-tail	2.131846486	

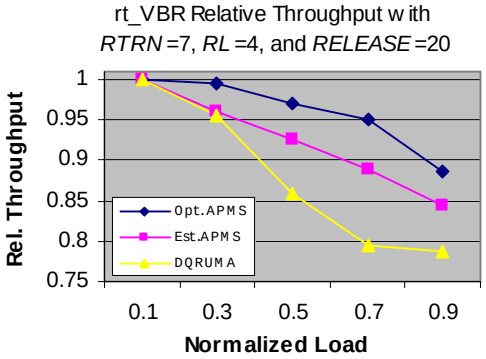


	Est. APMS	*DQRUMA*
Mean	0.98940236	0.88163356
Variance	0.000131469	0.008280526
Observations	5	5
Pearson Correlation	0.90201699	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.982160376	
P(T<=t) one-tail	0.020325952	
t Critical one-tail	2.131846486	

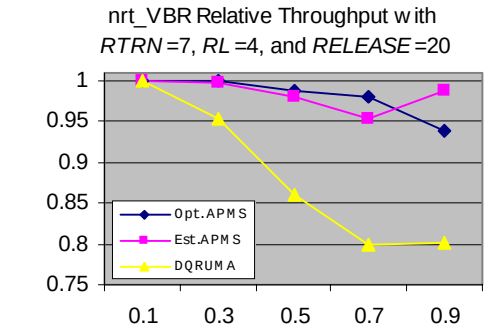
Figure E.18: Relative Throughput with RTRN=5, RL=6, and RELEASE=40



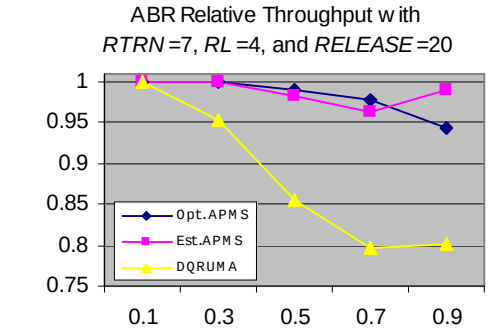
	Est. APMS	*DQRUMA*
Mean	0.92320864	0.87770944
Variance	0.003964904	0.00942216
Observations	5	5
Pearson Correlation	0.954495692	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.453803382	
P(T<=t) one-tail	0.035078739	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.92356652	0.8785806
Variance	0.003680907	0.009061623
Observations	5	5
Pearson Correlation	0.962840728	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.498453428	
P(T<=t) one-tail	0.033438499	
t Critical one-tail	2.131846486	

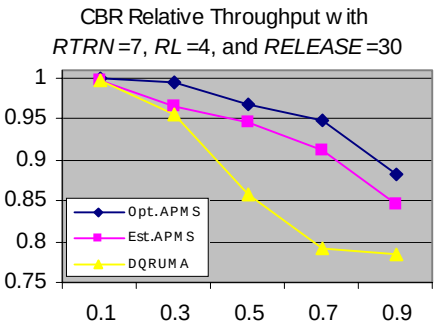


	Est. APMS	*DQRUMA*
Mean	0.98405012	0.88310364
Variance	0.000344587	0.008136462
Observations	5	5
Pearson Correlation	0.749662908	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.92125669	
P(T<=t) one-tail	0.021594599	
t Critical one-tail	2.131846486	

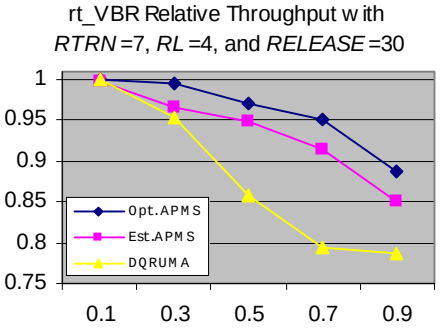


	Est. APMS	*DQRUMA*
Mean	0.98739732	0.88163356
Variance	0.000220147	0.008280526
Observations	5	5
Pearson Correlation	0.773296386	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.953298664	
P(T<=t) one-tail	0.020915979	
t Critical one-tail	2.131846486	

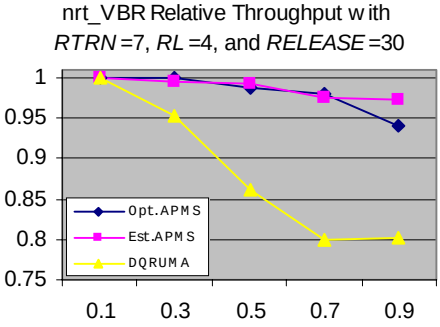
Figure E.19: Relative Throughput with RTRN=7, RL=4, and RELEASE=20



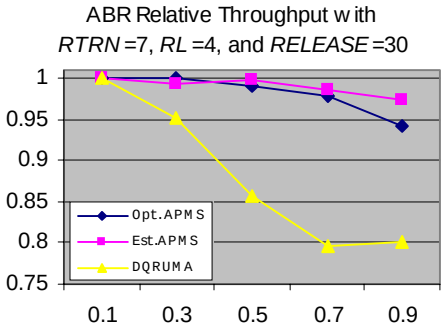
	Est. APMS	*DQRUMA*
Mean	0.933817	0.87770944
Variance	0.003387352	0.00942216
Observations	5	5
Pearson Correlation	0.900547096	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.444390572	
P(T<=t) one-tail	0.035436104	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93586132	0.8785806
Variance	0.003179069	0.009061623
Observations	5	5
Pearson Correlation	0.897537803	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.509009844	
P(T<=t) one-tail	0.0330636	
t Critical one-tail	2.131846486	

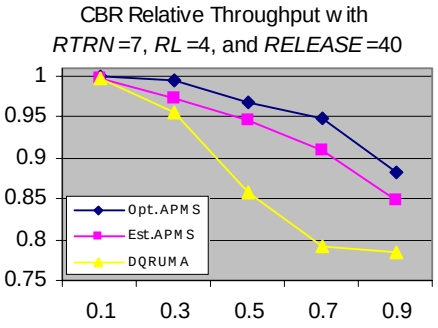


	Est. APMS	*DQRUMA*
Mean	0.98705452	0.88310364
Variance	0.000151711	0.008136462
Observations	5	5
Pearson Correlation	0.925200192	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.944341905	
P(T<=t) one-tail	0.021103136	
t Critical one-tail	2.131846486	

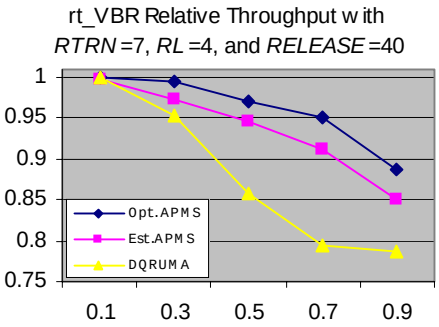


	Est. APMS	*DQRUMA*
Mean	0.9901226	0.88163356
Variance	0.000110932	0.008280526
Observations	5	5
Pearson Correlation	0.747910033	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.908265321	
P(T<=t) one-tail	0.02187704	
t Critical one-tail	2.131846486	

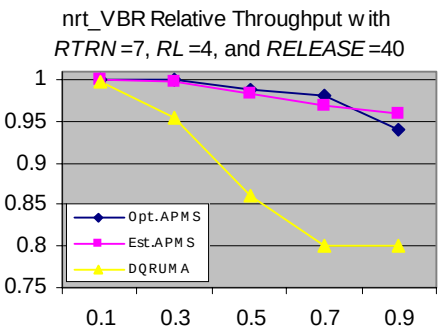
Figure E.20: Relative Throughput with RTRN=7, RL=4, and RELEASE=30



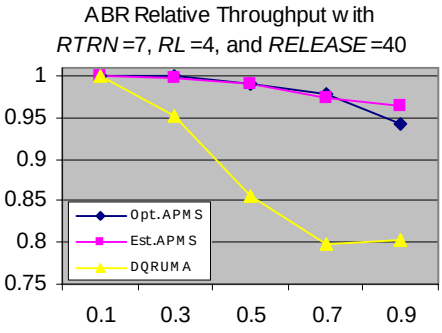
	Est. APMS	*DQRUMA*
Mean	0.93507064	0.87770944
Variance	0.003491522	0.00942216
Observations	5	5
Pearson Correlation	0.918839023	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.632796822	
P(T<=t) one-tail	0.029008015	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93642332	0.8785806
Variance	0.00332718	0.009061623
Observations	5	5
Pearson Correlation	0.913856203	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.666343372	
P(T<=t) one-tail	0.028009429	
t Critical one-tail	2.131846486	

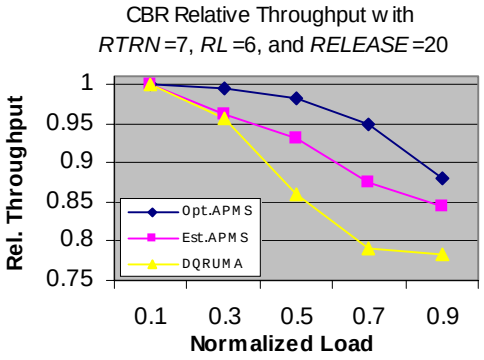


	Est. APMS	*DQRUMA*
Mean	0.98178528	0.88310364
Variance	0.000334136	0.008136462
Observations	5	5
Pearson Correlation	0.949092896	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.019386715	
P(T<=t) one-tail	0.019593427	
t Critical one-tail	2.131846486	

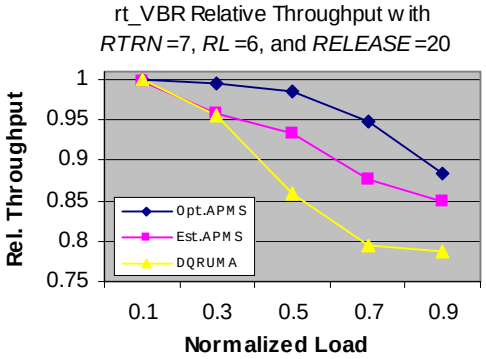


	Est. APMS	*DQRUMA*
Mean	0.98569004	0.88163356
Variance	0.000249822	0.008280526
Observations	5	5
Pearson Correlation	0.897927349	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.017105213	
P(T<=t) one-tail	0.019637423	
t Critical one-tail	2.131846486	

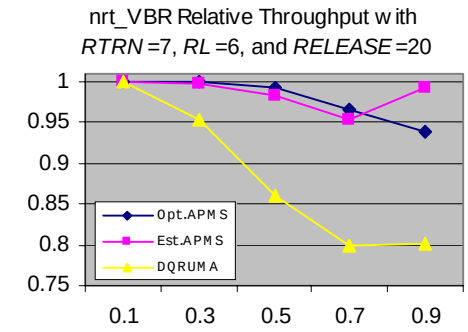
Figure E.21: Relative Throughput with RTRN=7, RL=4, and RELEASE=40



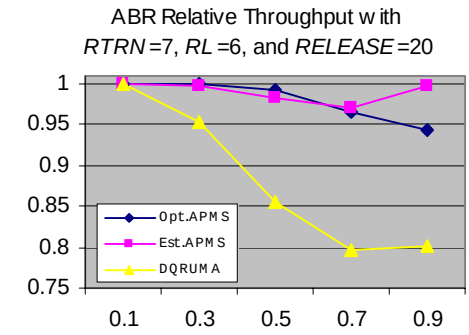
	Est. APMS	*DQRUMA*
Mean	0.92211268	0.87770944
Variance	0.003893069	0.00942216
Observations	5	5
Pearson Correlation	0.969092103	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.500546971	
P(T<=t) one-tail	0.033363765	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.92298848	0.8785806
Variance	0.003683767	0.009061623
Observations	5	5
Pearson Correlation	0.966846794	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.503499371	
P(T<=t) one-tail	0.033258696	
t Critical one-tail	2.131846486	

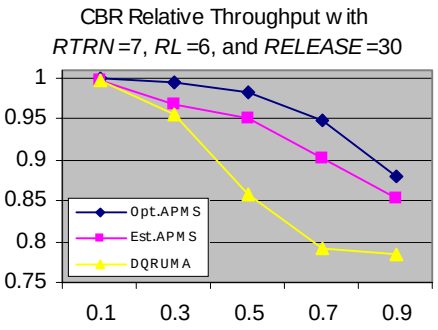


	Est. APMS	*DQRUMA*
Mean	0.98517536	0.88310364
Variance	0.000369106	0.008136462
Observations	5	5
Pearson Correlation	0.667525562	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.900525301	
P(T<=t) one-tail	0.02204736	
t Critical one-tail	2.131846486	

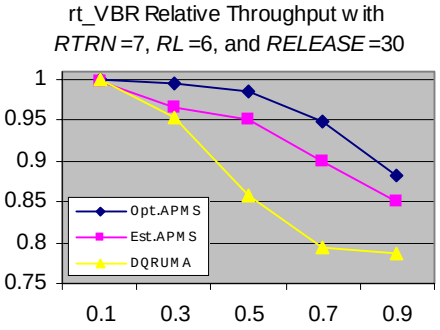


	Est. APMS	*DQRUMA*
Mean	0.9894986	0.88163356
Variance	0.000157909	0.008280526
Observations	5	5
Pearson Correlation	0.660234734	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.897652997	
P(T<=t) one-tail	0.022110958	
t Critical one-tail	2.131846486	

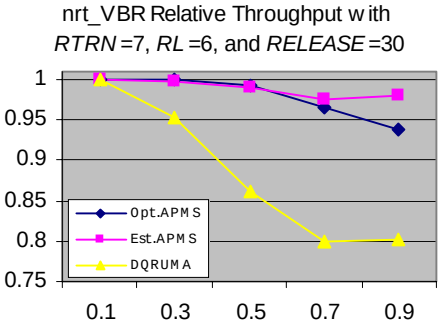
Figure E.22: Relative Throughput with RTRN=7, RL=6, and RELEASE=20



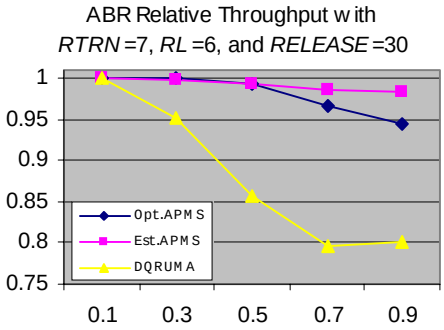
	Est. APMS	*DQRUMA*
Mean	0.93449648	0.87770944
Variance	0.003231998	0.00942216
Observations	5	5
Pearson Correlation	0.923465905	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.559062416	
P(T<=t) one-tail	0.031350152	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93360724	0.8785806
Variance	0.003377338	0.009061623
Observations	5	5
Pearson Correlation	0.919544443	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.585416138	
P(T<=t) one-tail	0.030488963	
t Critical one-tail	2.131846486	

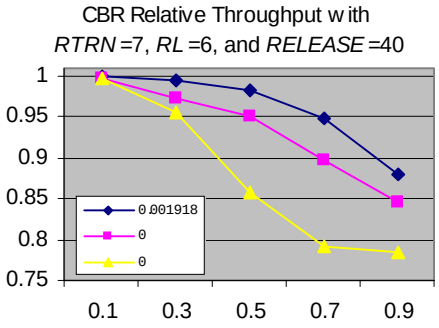


	Est. APMS	*DQRUMA*
Mean	0.98854164	0.88310364
Variance	0.0001142	0.008136462
Observations	5	5
Pearson Correlation	0.9635213	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.948671038	
P(T<=t) one-tail	0.021012433	
t Critical one-tail	2.131846486	

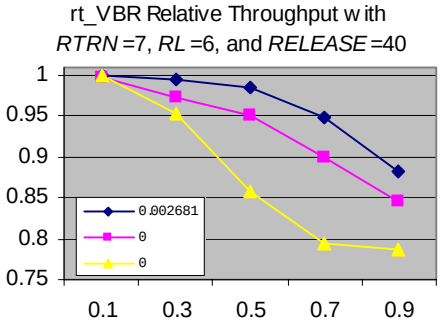


	Est. APMS	*DQRUMA*
Mean	0.99207456	0.88163356
Variance	5.61444E-05	0.008280526
Observations	5	5
Pearson Correlation	0.946871023	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.942125569	
P(T<=t) one-tail	0.021149749	
t Critical one-tail	2.131846486	

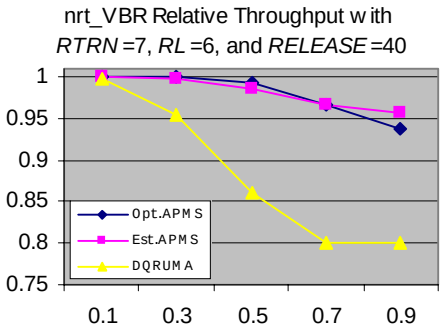
Figure E.23: Relative Throughput with RTRN=7, RL=6, and RELEASE=30



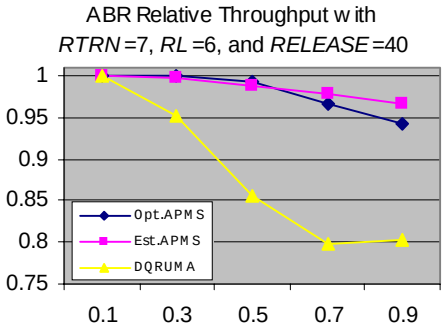
	Est. APMS	*DQRUMA*
Mean	0.9330322	0.87770944
Variance	0.003831969	0.00942216
Observations	5	5
Pearson Correlation	0.922725496	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.658505932	
P(T<=t) one-tail	0.028239136	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.93337588	0.8785806
Variance	0.003799591	0.009061623
Observations	5	5
Pearson Correlation	0.916239354	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.66821997	
P(T<=t) one-tail	0.027954748	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.98159968	0.88310364
Variance	0.000353799	0.008136462
Observations	5	5
Pearson Correlation	0.942155794	
Hypothesized Mean Difference	0	
df	4	
t Stat	3.027220308	
P(T<=t) one-tail	0.019443242	
t Critical one-tail	2.131846486	



	Est. APMS	*DQRUMA*
Mean	0.98621276	0.88163356
Variance	0.000198691	0.008280526
Observations	5	5
Pearson Correlation	0.913436951	
Hypothesized Mean Difference	0	
df	4	
t Stat	2.985315607	
P(T<=t) one-tail	0.020262636	
t Critical one-tail	2.131846486	

Figure E.24: Relative Throughput with RTRN=7, RL=6, and RELEASE=40

Appendix F: Confidence Interval for Method 2 (Varying the Number of MUs)

This appendix contains the confidence intervals of the relative throughputs for traffic following Method 2, where the number of MUs in the system is fixed varied. Table F.1 to F.24 list the confidence intervals for various combinations of factors pertaining the APMS protocol. Table F.25 shows the confidence interval for the relative throughput resulting from the DQRUMA protocol.

Confidence Interval for RTRN=0 and RL=4

Table F.1: Confidence Interval for RTRN=0, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00043	0.00014	5.4E-05	4.6E-05	8E-05	3.7E-05	0.00085	0.00032
0.3	0.00141	0.00049	0.00056	0.00011	0.00038	0.00012	0.00623	0.00098
0.5	0.00705	0.00548	0.00667	0.00503	0.00644	0.00584	0.00524	0.00416
0.7	0.0111	0.01375	0.01233	0.01368	0.01057	0.01265	0.00897	0.01468
0.9	0.01133	0.00765	0.01114	0.00613	0.01276	0.00869	0.01615	0.0093

Table F.2: Confidence Interval forRTRN=0, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00025	0.00014	0.00013	4.6E-05	9.4E-05	3.7E-05	0.00103	0.00032
0.3	0.00068	0.00049	0.00055	0.00011	0.00065	0.00012	0.00234	0.00098
0.5	0.01184	0.00548	0.01081	0.00503	0.00976	0.00584	0.01037	0.00416
0.7	0.01204	0.01375	0.01229	0.01368	0.01022	0.01265	0.01279	0.01468
0.9	0.0069	0.00765	0.00762	0.00613	0.00954	0.00869	0.0082	0.0093

Table F.3: Confidence Interval forRTRN=0, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00047	0.00014	0.00015	4.6E-05	6.1E-05	3.7E-05	0.00209	0.00032
0.3	0.00058	0.00049	0.00037	0.00011	0.00039	0.00012	0.00314	0.00098
0.5	0.01307	0.00548	0.01181	0.00503	0.01131	0.00584	0.01018	0.00416
0.7	0.0066	0.01375	0.0084	0.01368	0.00712	0.01265	0.00891	0.01468
0.9	0.00556	0.00765	0.0039	0.00613	0.00798	0.00869	0.0107	0.0093

Confidence Interval for RTRN=0 and RL=6

Table F.4: Confidence Interval for RTRN=0, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00031	0.00033	0.00025	5.3E-05	0.00018	7.3E-05	0.00051	0.00075
0.3	0.00928	0.00034	0.00969	0.00014	0.00932	0.00016	0.01231	0.0015
0.5	0.00612	0.0075	0.00636	0.007	0.00646	0.00702	0.00879	0.00838
0.7	0.01504	0.01073	0.01493	0.00976	0.01261	0.00922	0.01615	0.00701
0.9	0.00632	0.00135	0.00784	0.00218	0.0102	0.00269	0.01287	0.00603

Table F.5: Confidence Interval for RTRN=0, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00028	0.00033	0.00026	5.3E-05	0.00018	7.3E-05	0.00163	0.00075
0.3	0.00144	0.00034	0.00055	0.00014	0.00053	0.00016	0.00304	0.0015
0.5	0.00173	0.0075	0.00321	0.007	0.00328	0.00702	0.00708	0.00838
0.7	0.01237	0.01073	0.01056	0.00976	0.00951	0.00922	0.01179	0.00701
0.9	0.00554	0.00135	0.00671	0.00218	0.00857	0.00269	0.01361	0.00603

Table F.6: Confidence Interval for RTRN=0, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00037	0.00033	0.00013	5.3E-05	0.0002	7.3E-05	0.00235	0.00075
0.3	0.00096	0.00034	0.00049	0.00014	0.00052	0.00016	0.00599	0.0015
0.5	0.00817	0.0075	0.00738	0.007	0.00697	0.00702	0.00594	0.00838
0.7	0.00563	0.01073	0.006	0.00976	0.00641	0.00922	0.00532	0.00701
0.9	0.01558	0.00135	0.01356	0.00218	0.0188	0.00269	0.02451	0.00603

Confidence Interval for RTRN=3 and RL=4

Table F.7: Confidence Interval for RTRN=3, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00056	7.3E-05	0.00012	6.3E-05	6.4E-06	1.9E-05	#NUM!	#NUM!
0.3	0.00094	0.00039	0.00046	0.00023	4.4E-05	2.4E-05	0.00047	0.00017
0.5	0.00596	0.01217	0.0084	0.01059	0.00898	0.0107	0.0096	0.01056
0.7	0.00761	0.01316	0.00848	0.00984	0.00907	0.0104	0.00901	0.00734
0.9	0.00853	0.01028	0.00727	0.00626	0.01232	0.00806	0.01116	0.00827

Table F.8: Confidence Interval for RTRN=3, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00017	7.3E-05	0.00015	6.3E-05	2.5E-05	1.9E-05	#NUM!	#NUM!
0.3	0.00729	0.00039	0.00461	0.00023	0.00497	2.4E-05	0.00851	0.00017
0.5	0.00217	0.01217	0.0027	0.01059	0.0026	0.0107	0.00403	0.01056
0.7	0.00688	0.01316	0.00794	0.00984	0.00978	0.0104	0.01263	0.00734
0.9	0.01234	0.01028	0.00984	0.00626	0.01645	0.00806	0.01456	0.00827

Table F.9: Confidence Interval for RTRN=3, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00064	7.3E-05	0.00011	6.3E-05	1.8E-05	1.9E-05	#NUM!	#NUM!
0.3	0.00137	0.00039	0.00045	0.00023	4.6E-05	2.4E-05	0.00103	0.00017
0.5	0.00852	0.01217	0.00715	0.01059	0.00722	0.0107	0.00853	0.01056
0.7	0.00955	0.01316	0.00918	0.00984	0.01082	0.0104	0.01255	0.00734
0.9	0.0049	0.01028	0.00291	0.00626	0.00441	0.00806	0.00459	0.00827

Confidence Interval for RTRN=3 and RL=6

Table F.10: Confidence Interval for RTRN=3, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00037	0.00037	0.00025	9.3E-05	1.7E-05	1.1E-05	#NUM!	#NUM!
0.3	0.00146	0.00054	0.00121	9.3E-05	5.1E-05	3.1E-05	0.00092	0.00012
0.5	0.01243	0.00886	0.01069	0.00602	0.01159	0.0074	0.01259	0.00738
0.7	0.01178	0.00752	0.0092	0.00917	0.01237	0.00933	0.01772	0.01141
0.9	0.00209	0.0111	0.00235	0.01164	0.00332	0.0182	0.00308	0.01848

Table F.11: Confidence Interval for RTRN=3, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00059	0.00037	0.00022	9.3E-05	1.5E-05	1.1E-05	#NUM!	#NUM!
0.3	0.00202	0.00054	0.00106	9.3E-05	0.0013	3.1E-05	0.0002	0.00012
0.5	0.00738	0.00886	0.00714	0.00602	0.00681	0.0074	0.00803	0.00738
0.7	0.00234	0.00752	0.002	0.00917	0.00189	0.00933	0.00211	0.01141
0.9	0.00297	0.0111	0.0023	0.01164	0.00378	0.0182	0.00308	0.01848

Table F.12: Confidence Interval for RTRN=3, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00026	0.00037	0.00016	9.3E-05	1.7E-05	1.1E-05	0.0001	#NUM!
0.3	0.00115	0.00054	0.00025	9.3E-05	4.9E-05	3.1E-05	0.00029	0.00012
0.5	0.00486	0.00886	0.00332	0.00602	0.00409	0.0074	0.0027	0.00738
0.7	0.01057	0.00752	0.01063	0.00917	0.01274	0.00933	0.01253	0.01141
0.9	0.00957	0.0111	0.00883	0.01164	0.01207	0.0182	0.0114	0.01848

Confidence Interval for RTRN=5 and RL=4

Table F.13: Confidence Interval for RTRN=5, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00056	7.3E-05	0.00012	6.3E-05	6.4E-06	1.9E-05	#NUM!	#NUM!
0.3	0.00128	0.00039	0.00043	0.00023	6E-05	2.4E-05	0.0003	0.00017
0.5	0.01386	0.0116	0.01397	0.01102	0.01453	0.01118	0.01041	0.00972
0.7	0.00927	0.0083	0.00806	0.00811	0.01117	0.00879	0.01069	0.00992
0.9	0.00489	0.01266	0.00411	0.01056	0.00721	0.01481	0.00746	0.01758

Table F.14: Confidence Interval for RTRN=5, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00017	7.3E-05	0.00015	6.3E-05	2.5E-05	1.9E-05	#NUM!	#NUM!
0.3	0.00715	0.00039	0.00453	0.00023	0.00498	2.4E-05	0.00848	0.00017
0.5	0.00308	0.0116	0.00314	0.01102	0.00354	0.01118	0.00187	0.00972
0.7	0.01481	0.0083	0.01339	0.00811	0.01569	0.00879	0.01742	0.00992
0.9	0.00366	0.01266	0.0058	0.01056	0.0088	0.01481	0.01118	0.01758

Table F.15: Confidence Interval for RTRN=5, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00064	7.3E-05	0.00011	6.3E-05	1.8E-05	1.9E-05	#NUM!	#NUM!
0.3	0.00137	0.00039	0.00045	0.00023	4.6E-05	2.4E-05	0.00103	0.00017
0.5	0.00411	0.0116	0.0042	0.01102	0.00508	0.01118	0.00765	0.00972
0.7	0.00908	0.0083	0.01016	0.00811	0.0123	0.00879	0.00917	0.00992
0.9	0.01121	0.01266	0.0096	0.01056	0.017	0.01481	0.0196	0.01758

Confidence Interval for RTRN=5 and RL=6

Table F.16: Confidence Interval for RTRN=5, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00037	0.00037	0.00025	9.3E-05	1.7E-05	1.1E-05	#NUM!	#NUM!
0.3	0.00144	0.00054	0.00133	9.3E-05	6.1E-05	3.1E-05	0.00089	0.00012
0.5	0.00539	0.00886	0.00546	0.00602	0.00574	0.0074	0.00634	0.00738
0.7	0.00805	0.00752	0.00973	0.00917	0.01162	0.00933	0.01355	0.01141
0.9	0.00263	0.0111	0.00122	0.01164	0.00036	0.0182	0.00047	0.01848

Table F.17: Confidence Interval for RTRN=5, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00059	0.00037	0.00022	9.3E-05	1.5E-05	1.1E-05	#NUM!	#NUM!
0.3	0.00202	0.00054	0.00106	9.3E-05	0.0013	3.1E-05	0.0002	0.00012
0.5	0.00233	0.00886	0.00374	0.00602	0.00393	0.0074	0.00459	0.00738
0.7	0.00849	0.00752	0.0072	0.00917	0.01015	0.00933	0.00797	0.01141
0.9	0.00527	0.0111	0.00448	0.01164	0.00591	0.0182	0.00757	0.01848

Table F.18: Confidence Interval for RTRN=5, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00026	0.00037	0.00016	9.3E-05	1.7E-05	1.1E-05	0.0001	#NUM!
0.3	0.00115	0.00054	0.00025	9.3E-05	4.9E-05	3.1E-05	0.00029	0.00012
0.5	0.002	0.00886	0.00329	0.00602	0.00404	0.0074	0.00409	0.00738
0.7	0.00345	0.00752	0.00493	0.00917	0.00537	0.00933	0.00554	0.01141
0.9	0.00726	0.0111	0.00614	0.01164	0.00896	0.0182	0.00973	0.01848

Confidence Interval for RTRN=7 and RL=4

Table F.19: Confidence Interval for RTRN=7, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00056	7.3E-05	0.00012	6.3E-05	6.4E-06	1.9E-05	#NUM!	#NUM!
0.3	0.00128	0.00039	0.00043	0.00023	6E-05	2.4E-05	0.0003	0.00017
0.5	0.0073	0.01235	0.00978	0.0107	0.00987	0.01115	0.00924	0.00836
0.7	0.00061	0.00847	0.00347	0.00822	0.00392	0.009	0.00753	0.01007
0.9	0.00382	0.01266	0.00283	0.01056	0.0048	0.01481	0.00562	0.01758

Table F.20: Confidence Interval for RTRN=7, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00017	7.3E-05	0.00015	6.3E-05	2.5E-05	1.9E-05	#NUM!	#NUM!
0.3	0.00715	0.00039	0.00453	0.00023	0.00498	2.4E-05	0.00848	0.00017
0.5	0.004	0.01235	0.00316	0.0107	0.00351	0.01115	0.00177	0.00836
0.7	0.00601	0.00847	0.00592	0.00822	0.00662	0.009	0.00649	0.01007
0.9	0.00622	0.01266	0.00586	0.01056	0.00864	0.01481	0.0114	0.01758

Table F.21: Confidence Interval for RTRN=7, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00064	7.3E-05	0.00011	6.3E-05	1.8E-05	1.9E-05	#NUM!	#NUM!
0.3	0.00137	0.00039	0.00045	0.00023	4.6E-05	2.4E-05	0.00103	0.00017
0.5	0.00705	0.01235	0.00671	0.0107	0.00732	0.01115	0.00583	0.00836
0.7	0.00427	0.00847	0.007	0.00822	0.00745	0.009	0.00999	0.01007
0.9	0.0139	0.01266	0.01462	0.01056	0.02147	0.01481	0.01923	0.01758

Confidence Interval for RTRN=7 and RL=6

Table F.22: Confidence Interval for RTRN=7, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00037	0.00037	0.00025	9.3E-05	1.7E-05	1.1E-05	#NUM!	#NUM!
0.3	0.00144	0.00054	0.00133	9.3E-05	6.1E-05	3.1E-05	0.00089	0.00012
0.5	0.00886	0.00886	0.00647	0.00602	0.00686	0.0074	0.00792	0.00738
0.7	0.01058	0.00752	0.01073	0.00917	0.01503	0.00933	0.01627	0.01141
0.9	0.00188	0.0111	0.00173	0.01164	0.00234	0.0182	0.00304	0.01848

Table F.23: Confidence Interval for RTRN=7, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00059	0.00037	0.00022	9.3E-05	1.5E-05	1.1E-05	#NUM!	#NUM!
0.3	0.00202	0.00054	0.00106	9.3E-05	0.0013	3.1E-05	0.0002	0.00012
0.5	0.00349	0.00886	0.00237	0.00602	0.00283	0.0074	0.00427	0.00738
0.7	0.0053	0.00752	0.00412	0.00917	0.00674	0.00933	0.00839	0.01141
0.9	0.00839	0.0111	0.00835	0.01164	0.00988	0.0182	0.01097	0.01848

Table F.24: Confidence Interval for RTRN=7, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00026	0.00037	0.00016	9.3E-05	1.7E-05	1.1E-05	0.0001	#NUM!
0.3	0.00115	0.00054	0.00025	9.3E-05	4.9E-05	3.1E-05	0.00029	0.00012
0.5	0.00382	0.00886	0.00485	0.00602	0.00598	0.0074	0.00665	0.00738
0.7	0.00874	0.00752	0.00606	0.00917	0.00731	0.00933	0.00665	0.01141
0.9	0.0155	0.0111	0.01228	0.01164	0.01821	0.0182	0.01622	0.01848

Confidence Interval for DQRUMA (RTRN=0 and RL=1)

Table F.25: Confidence Interval for DQRUMA (RTRN=0 and RL=1)

	Traffic Type			
Load	CBR	rt-VBR	nrt-VBR	ABR
0.1	0.00058	9.7E-05	4.8E-05	#NUM!
0.3	0.00211	0.00484	0.00512	0.00317
0.5	0.00457	0.0037	0.00272	0.00371
0.7	0.01589	0.0132	0.0135	0.0125
0.9	0.00906	0.00508	0.00492	0.0076

Appendix G: Average Delay for Method 2 (Varying the Number of MUs)

This appendix contains the average delay of for the uplink using Method 1, where the number of MUs in the system is varied. Table G.1 to G.24 shows the average delay for each type of traffic using the APMS protocol. The average delay for the DQRUMA protocol is given in Table G.25.

Average Delay for RTRN=0 and RL=4

Table G.1: Average Delay for RTRN=0, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00187	0.00227
0.3	0.00147	0.00147	0.00158	0.00158	0.00158	0.00159	0.00247	0.00263
0.5	0.00192	0.00199	0.00205	0.00213	0.0021	0.00221	0.00333	0.00362
0.7	0.00245	0.0025	0.0026	0.00269	0.00296	0.00371	0.00554	0.00732
0.9	0.00287	0.00328	0.00308	0.00357	0.00384	0.00784	0.00675	0.00994

Table G.2: Average Delay for RTRN=0, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00117	0.00126	0.00126	0.00126	0.00126	0.00182	0.00227
0.3	0.00146	0.00147	0.00157	0.00158	0.00157	0.00159	0.00258	0.00263
0.5	0.00193	0.00199	0.00206	0.00213	0.00213	0.00221	0.00364	0.00362
0.7	0.00243	0.0025	0.00261	0.00269	0.00329	0.00371	0.00693	0.00732
0.9	0.00299	0.00328	0.00327	0.00357	0.00463	0.00784	0.00858	0.00994

Table G.3: Average Delay for RTRN=0, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00117	0.00126	0.00126	0.00126	0.00126	0.00184	0.00227
0.3	0.00147	0.00147	0.00158	0.00158	0.00158	0.00159	0.00261	0.00263
0.5	0.00193	0.00199	0.00207	0.00213	0.00218	0.00221	0.00423	0.00362
0.7	0.00249	0.0025	0.00267	0.00269	0.00397	0.00371	0.00925	0.00732
0.9	0.00317	0.00328	0.0035	0.00357	0.00594	0.00784	0.01015	0.00994

Average Delay for RTRN=0 and RL=6

Table G.4: Average Delay for RTRN=0, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00186	0.00291
0.3	0.00146	0.00147	0.00157	0.00159	0.00157	0.0016	0.00262	0.00321
0.5	0.00191	0.00196	0.00205	0.00211	0.00215	0.00246	0.00364	0.00533
0.7	0.00243	0.00275	0.00259	0.00297	0.00295	0.00597	0.00498	0.00913
0.9	0.00287	0.00331	0.0031	0.00363	0.00376	0.00858	0.00638	0.01043

Table G.5: Average Delay for RTRN=0, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00186	0.00291
0.3	0.00146	0.00147	0.00158	0.00159	0.00158	0.0016	0.00303	0.00321
0.5	0.00191	0.00196	0.00206	0.00211	0.00224	0.00246	0.00416	0.00533
0.7	0.00247	0.00275	0.00264	0.00297	0.00314	0.00597	0.00615	0.00913
0.9	0.00299	0.00331	0.00326	0.00363	0.00426	0.00858	0.00778	0.01043

Table G.6: Average Delay for RTRN=0, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00194	0.00291
0.3	0.00146	0.00147	0.00157	0.00159	0.00158	0.0016	0.00305	0.00321
0.5	0.00193	0.00196	0.00207	0.00211	0.00238	0.00246	0.00477	0.00533
0.7	0.00264	0.00275	0.00283	0.00297	0.00365	0.00597	0.00779	0.00913
0.9	0.00317	0.00331	0.0035	0.00363	0.00522	0.00858	0.00951	0.01043

Average Delay for RTRN=3 and RL=4

Table G.7: Average Delay for RTRN=3, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00186	0.0024
0.3	0.00147	0.00148	0.00158	0.00159	0.00165	0.0016	0.0028	0.0027
0.5	0.00194	0.00199	0.00207	0.00213	0.00237	0.00227	0.00438	0.00373
0.7	0.00237	0.00252	0.00252	0.00271	0.00492	0.00472	0.00957	0.00834
0.9	0.00259	0.00323	0.00293	0.00354	0.00896	0.00832	0.01238	0.01064

Table G.8: Average Delay for RTRN=3, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.0019	0.0024
0.3	0.00146	0.00148	0.00157	0.00159	0.00163	0.0016	0.00282	0.0027
0.5	0.00193	0.00199	0.00206	0.00213	0.00238	0.00227	0.00462	0.00373
0.7	0.00245	0.00252	0.00262	0.00271	0.00651	0.00472	0.01147	0.00834
0.9	0.0029	0.00323	0.0033	0.00354	0.01034	0.00832	0.0125	0.01064

Table G.9: Average Delay for RTRN=3, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00117	0.00126	0.00126	0.00126	0.00126	0.00185	0.0024
0.3	0.00147	0.00148	0.00158	0.00159	0.00163	0.0016	0.00287	0.0027
0.5	0.00193	0.00199	0.00207	0.00213	0.00256	0.00227	0.00594	0.00373
0.7	0.00257	0.00252	0.00277	0.00271	0.00792	0.00472	0.0124	0.00834
0.9	0.00324	0.00323	0.00372	0.00354	0.01164	0.00832	0.01378	0.01064

Average Delay for RTRN=3 and RL=6

Table G.10: Average Delay for RTRN=3, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00118	0.00126	0.00126	0.00126	0.00126	0.00187	0.00255
0.3	0.00146	0.00145	0.00157	0.00158	0.00166	0.0016	0.00333	0.00314
0.5	0.00188	0.00196	0.00202	0.0021	0.00257	0.0025	0.00606	0.0056
0.7	0.00237	0.00271	0.00255	0.00295	0.00513	0.00595	0.00934	0.00883
0.9	0.00253	0.00328	0.00293	0.00361	0.00939	0.00857	0.01185	0.01016

Table G.11: Average Delay for RTRN=3, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00118	0.00126	0.00126	0.00126	0.00126	0.00192	0.00255
0.3	0.00146	0.00145	0.00157	0.00158	0.00165	0.0016	0.00336	0.00314
0.5	0.00192	0.00196	0.00206	0.0021	0.00306	0.0025	0.0072	0.0056
0.7	0.00253	0.00271	0.00272	0.00295	0.00682	0.00595	0.01055	0.00883
0.9	0.00289	0.00328	0.00334	0.00361	0.01064	0.00857	0.013	0.01016

Table G.12: Average Delay for RTRN=3, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00118	0.00126	0.00126	0.00126	0.00126	0.00188	0.00255
0.3	0.00147	0.00145	0.00158	0.00158	0.00165	0.0016	0.00354	0.00314
0.5	0.00195	0.00196	0.0021	0.0021	0.00358	0.0025	0.00858	0.0056
0.7	0.00273	0.00271	0.00295	0.00295	0.0083	0.00595	0.01224	0.00883
0.9	0.00323	0.00328	0.00369	0.00361	0.01153	0.00857	0.01288	0.01016

Average Delay for RTRN=5 and RL=4

Table G.13: Average Delay for RTRN=5, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00186	0.0024
0.3	0.00147	0.00148	0.00158	0.00159	0.00165	0.0016	0.00276	0.0027
0.5	0.00194	0.00199	0.00208	0.00213	0.00236	0.00226	0.00433	0.00383
0.7	0.00235	0.00255	0.0025	0.00275	0.00522	0.00501	0.00931	0.00879
0.9	0.00255	0.00325	0.00295	0.00354	0.01003	0.0084	0.01203	0.01045

Table G.14: Average Delay for RTRN=5, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.0019	0.0024
0.3	0.00147	0.00148	0.00157	0.00159	0.00163	0.0016	0.00293	0.0027
0.5	0.00195	0.00199	0.00208	0.00213	0.00243	0.00226	0.00521	0.00383
0.7	0.00242	0.00255	0.00259	0.00275	0.00655	0.00501	0.0109	0.00879
0.9	0.00288	0.00325	0.00332	0.00354	0.01087	0.0084	0.01319	0.01045

Table G.15: Average Delay for RTRN=5, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00117	0.00126	0.00126	0.00126	0.00126	0.00185	0.0024
0.3	0.00147	0.00148	0.00158	0.00159	0.00163	0.0016	0.00287	0.0027
0.5	0.00192	0.00199	0.00206	0.00213	0.00255	0.00226	0.00603	0.00383
0.7	0.0026	0.00255	0.00279	0.00275	0.00804	0.00501	0.01209	0.00879
0.9	0.00321	0.00325	0.00365	0.00354	0.01152	0.0084	0.01395	0.01045

Average Delay for RTRN=5 and RL=6

Table G.16: Average Delay for RTRN=5, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00118	0.00126	0.00126	0.00126	0.00126	0.00187	0.00255
0.3	0.00146	0.00145	0.00157	0.00158	0.00166	0.0016	0.00326	0.00314
0.5	0.0019	0.00196	0.00203	0.0021	0.00255	0.0025	0.00573	0.0056
0.7	0.00236	0.00271	0.00253	0.00295	0.00595	0.00595	0.00969	0.00883
0.9	0.0025	0.00328	0.00296	0.00361	0.01046	0.00857	0.01318	0.01016

Table G.17: Average Delay for RTRN=5, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00118	0.00126	0.00126	0.00126	0.00126	0.00192	0.00255
0.3	0.00146	0.00145	0.00157	0.00158	0.00165	0.0016	0.00336	0.00314
0.5	0.00191	0.00196	0.00206	0.0021	0.00304	0.0025	0.00734	0.0056
0.7	0.00253	0.00271	0.0027	0.00295	0.00742	0.00595	0.011	0.00883
0.9	0.00288	0.00328	0.00333	0.00361	0.01094	0.00857	0.01308	0.01016

Table G.18: Average Delay for RTRN=5, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00118	0.00126	0.00126	0.00126	0.00126	0.00188	0.00255
0.3	0.00147	0.00145	0.00158	0.00158	0.00165	0.0016	0.00354	0.00314
0.5	0.00195	0.00196	0.00211	0.0021	0.00363	0.0025	0.00956	0.0056
0.7	0.00276	0.00271	0.00297	0.00295	0.00862	0.00595	0.01196	0.00883
0.9	0.00323	0.00328	0.00368	0.00361	0.01167	0.00857	0.01412	0.01016

Average Delay for RTRN=7 and RL=4

Table G.19: Average Delay for RTRN=7, RL=4, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.00186	0.0024
0.3	0.00147	0.00148	0.00158	0.00159	0.00165	0.0016	0.00276	0.0027
0.5	0.00192	0.002	0.00206	0.00213	0.00235	0.00227	0.00445	0.00395
0.7	0.00234	0.00255	0.0025	0.00275	0.0052	0.005	0.00924	0.00887
0.9	0.00254	0.00325	0.00295	0.00354	0.01029	0.0084	0.01218	0.01045

Table G.20: Average Delay for RTRN=7, RL=4, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00117	0.00126	0.00126	0.00126	0.00126	0.0019	0.0024
0.3	0.00147	0.00148	0.00157	0.00159	0.00163	0.0016	0.00293	0.0027
0.5	0.00195	0.002	0.00208	0.00213	0.00243	0.00227	0.00506	0.00395
0.7	0.00246	0.00255	0.00263	0.00275	0.00678	0.005	0.01087	0.00887
0.9	0.00289	0.00325	0.00331	0.00354	0.01073	0.0084	0.01273	0.01045

Table G.21: Average Delay for RTRN=7, RL=4, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00117	0.00126	0.00126	0.00126	0.00126	0.00185	0.0024
0.3	0.00147	0.00148	0.00158	0.00159	0.00163	0.0016	0.00287	0.0027
0.5	0.00194	0.002	0.00206	0.00213	0.00254	0.00227	0.00582	0.00395
0.7	0.00258	0.00255	0.00277	0.00275	0.00791	0.005	0.01184	0.00887
0.9	0.00319	0.00325	0.00362	0.00354	0.01148	0.0084	0.01345	0.01045

Average Delay for RTRN=7 and RL=6

Table G.22: Average Delay for RTRN=7, RL=6, and RELEASE=20

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00118	0.00118	0.00126	0.00126	0.00126	0.00126	0.00187	0.00255
0.3	0.00146	0.00145	0.00157	0.00158	0.00166	0.0016	0.00326	0.00314
0.5	0.0019	0.00196	0.00204	0.0021	0.00258	0.0025	0.00633	0.0056
0.7	0.00235	0.00271	0.00253	0.00295	0.00632	0.00595	0.00978	0.00883
0.9	0.00253	0.00328	0.00294	0.00361	0.01023	0.00857	0.01253	0.01016

Table G.23: Average Delay for RTRN=7, RL=6, and RELEASE=30

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00118	0.00126	0.00126	0.00126	0.00126	0.00192	0.00255
0.3	0.00146	0.00145	0.00157	0.00158	0.00165	0.0016	0.00336	0.00314
0.5	0.00192	0.00196	0.00206	0.0021	0.00299	0.0025	0.00705	0.0056
0.7	0.00254	0.00271	0.00273	0.00295	0.00772	0.00595	0.0112	0.00883
0.9	0.00288	0.00328	0.00331	0.00361	0.01087	0.00857	0.01366	0.01016

Table G.24: Average Delay for RTRN=7, RL=6, and RELEASE=40

	Traffic Type							
Load	CBR (Est.)	CBR (Opt.)	rt-VBR (Est.)	rt-VBR (Opt.)	nrt-VBR (Est.)	nrt-VBR (Opt.)	ABR (Est.)	ABR (Opt.)
0.1	0.00117	0.00118	0.00126	0.00126	0.00126	0.00126	0.00188	0.00255
0.3	0.00147	0.00145	0.00158	0.00158	0.00165	0.0016	0.00354	0.00314
0.5	0.00196	0.00196	0.0021	0.0021	0.00373	0.0025	0.00884	0.0056
0.7	0.00275	0.00271	0.00295	0.00295	0.00865	0.00595	0.01224	0.00883
0.9	0.00322	0.00328	0.00365	0.00361	0.01153	0.00857	0.01312	0.01016

Average Delay for DQRUMA (RTRN=0 and RL=1)

Table G.25: Average Delay for DQRUMA (RTRN=0 and RL=1)

	Traffic Type			
Load	CBR	rt-VBR	nrt-VBR	ABR
0.1	0.00111	0.00119	0.00133	0.00191
0.3	0.00138	0.0015	0.00166	0.00219
0.5	0.00191	0.00207	0.00228	0.00283
0.7	0.00254	0.00275	0.00318	0.00389
0.9	0.00288	0.00319	0.00425	0.00546

Appendix H: Queues Size and Queue Delay Results

This appendix contains the graphs corresponding to the variation in queue size and delay within simulation runs for different combination of simulation factors. These are presented to give a more detailed look into each simulation results presented in Section 6.1 and 6.2. All plots are related to simulations of DQRUMA or APMS with 0.1 or 0.9 normalized load levels.

Section H.1 lists the plots of queue size and queue delay corresponding to simulation figures presented in Section 6.1. Section H.2 lists the plots of queue size and queue delay corresponding to simulation figures presented in Section 6.2. These results show that the queue sizes are not monotonically increasing or decreasing as the simulation progresses from 0 second to 30 seconds, which reinforces the validity of the simulation results. A more detailed discussion about the simulation length was presented in Section 5.6.1.

H.1 Queue Size and Queue Delay Results Corresponding to Section 6.1

H.1.1 Simulation Corresponding to 0.1 Normalized Load Level

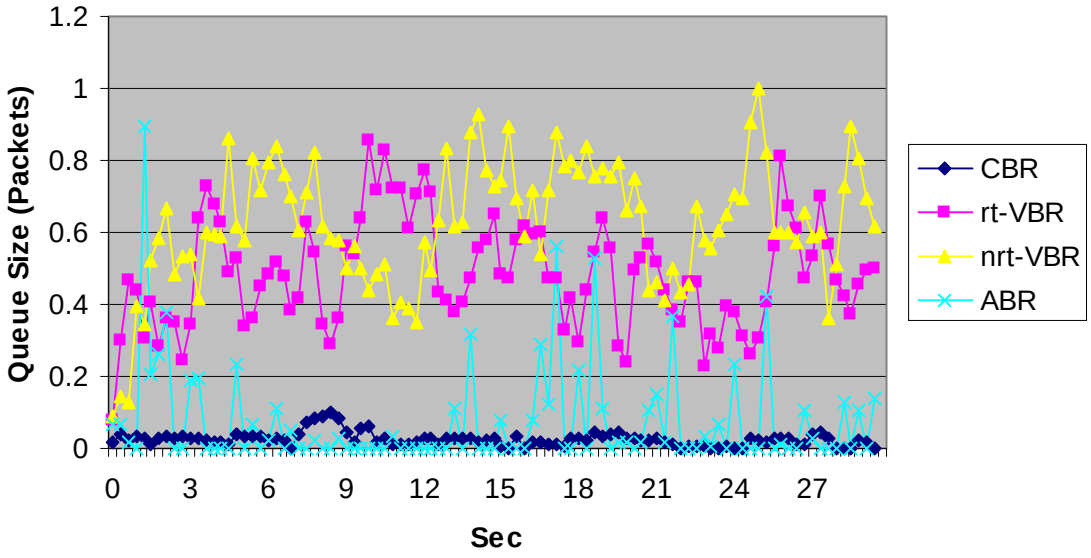


Figure H.1: Queue Size Plots for DQRUMA protocol (Section 6.1, 0.1 Load)

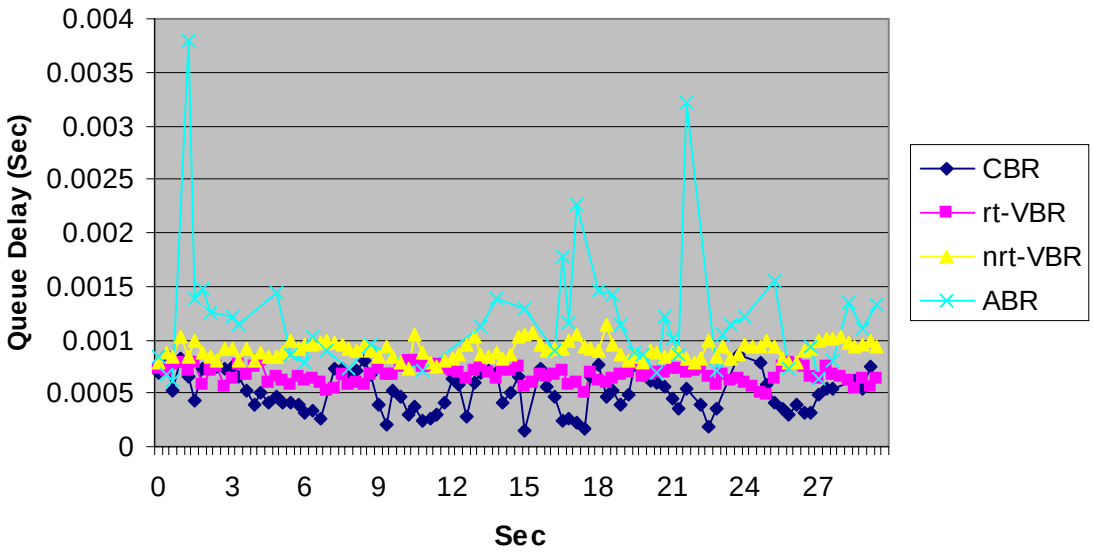


Figure H.2: Queue Delay of Plots for DQRUMA protocol (Section 6.1, 0.1 Load)

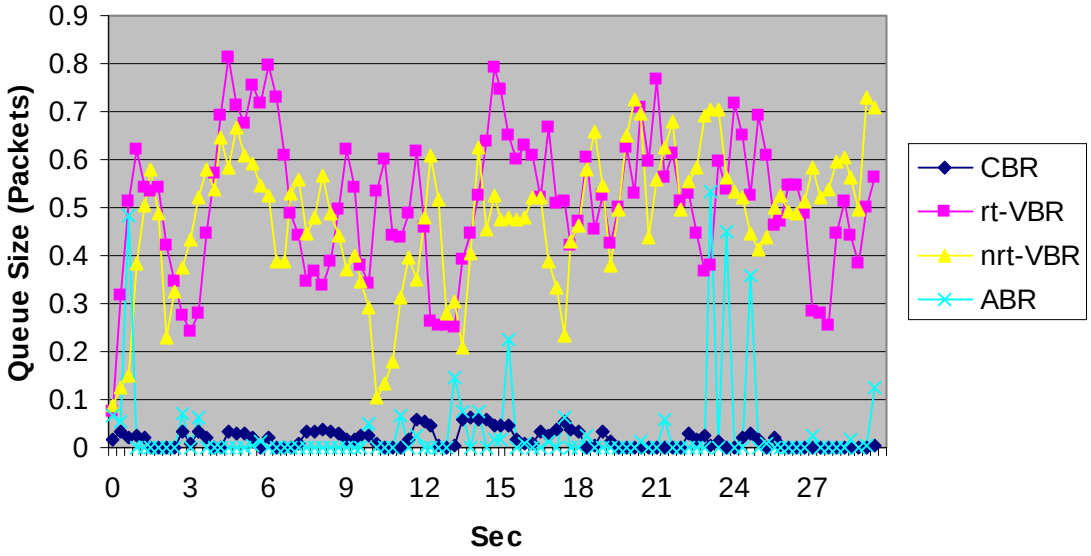


Figure H.3: Queue Size Plots for APMS Protocol (Section 6.1, 0.1 Load)

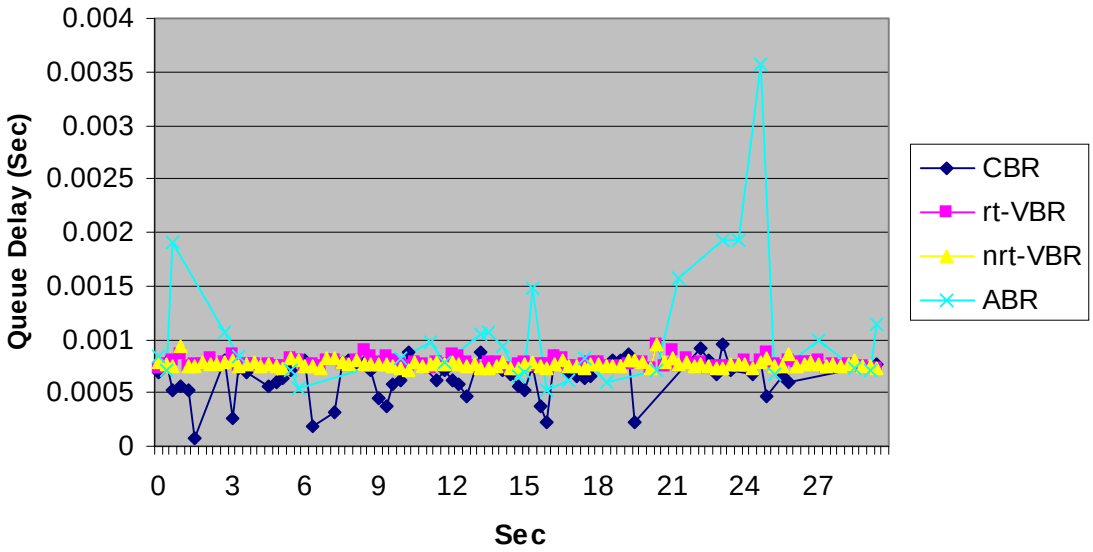


Figure H.4: Queue Delay of Plots for APMS Protocol (Section 6.1, 0.1 Load)

H.1.2 Simulation Corresponding to 0.9 Normalized Load Level

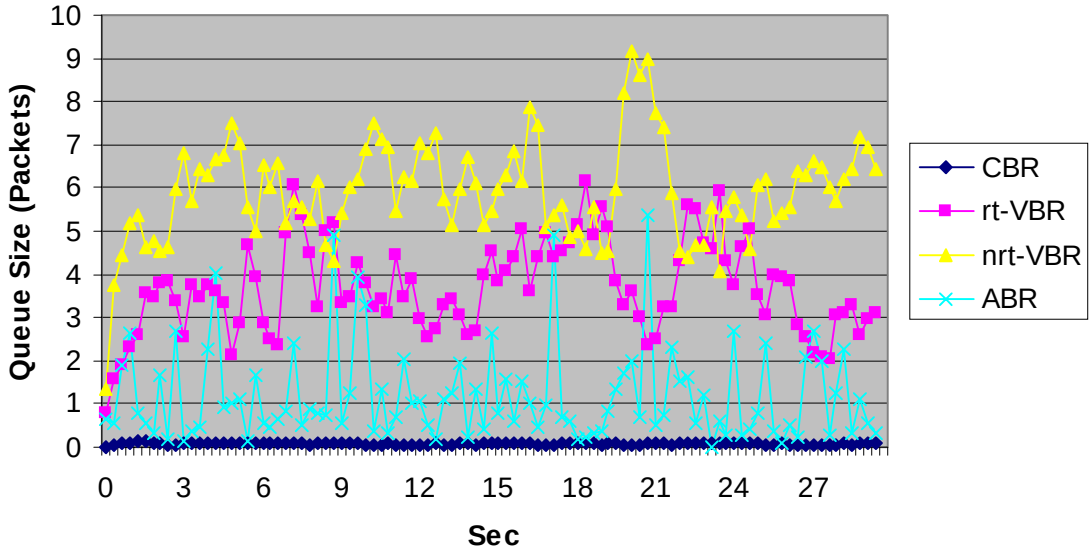


Figure H.5: Queue Size Plots for DQRUMA protocol (Section 6.1, 0.9 Load)

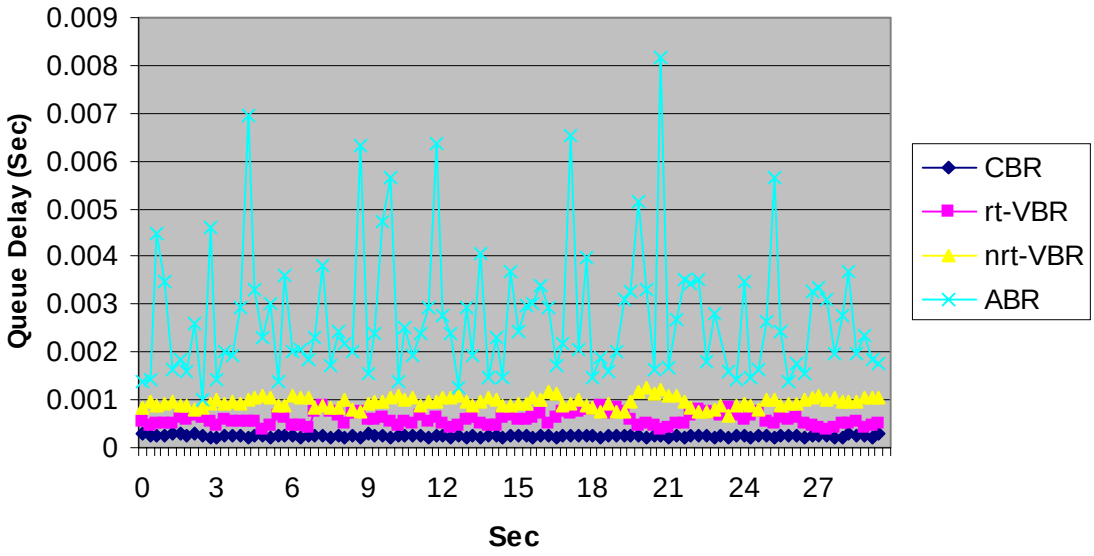


Figure H.6: Queue Delay of Plots for DQRUMA protocol (Section 6.1, 0.9 Load)

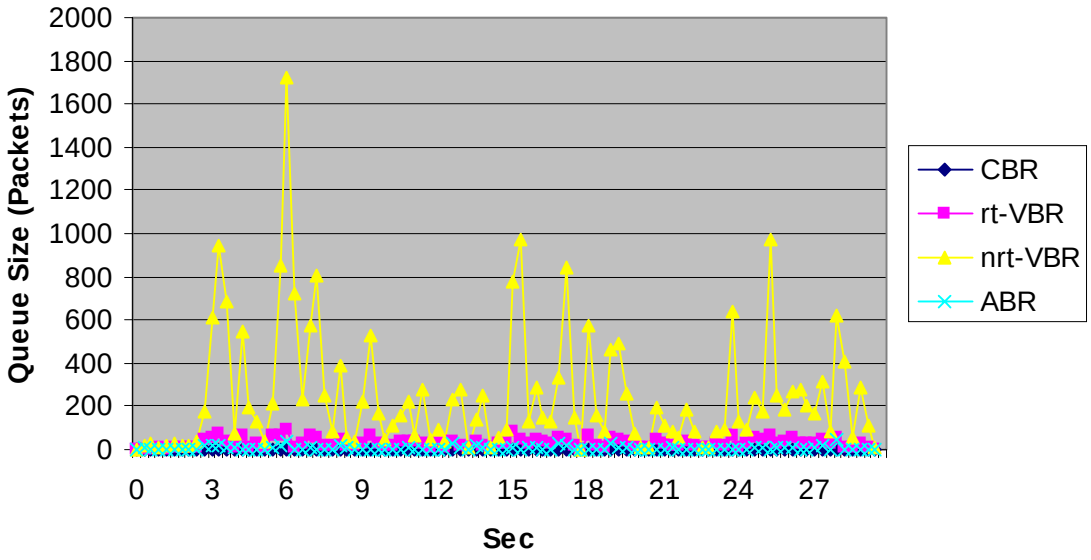


Figure H.7: Queue Size Plots for APMS Protocol (Section 6.1, 0.9 Load)

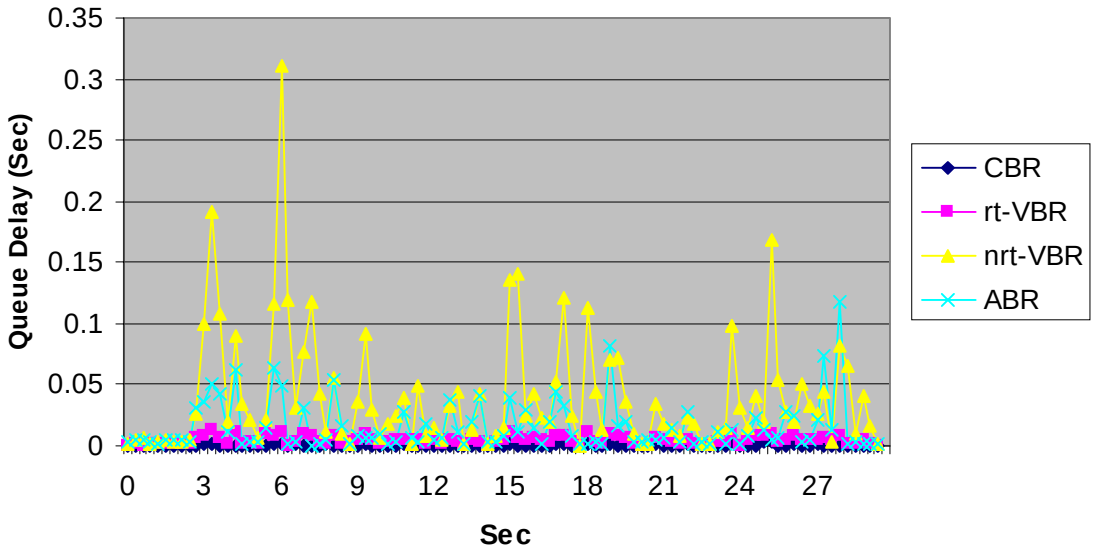


Figure H.8: Queue Delay of Plots for APMS Protocol (Section 6.1, 0.9 Load)

H.2 Queue Size and Queue Delay Results Corresponding to Section 6.2

H.2.1 Simulation Corresponding to 0.1 Normalized Load Level

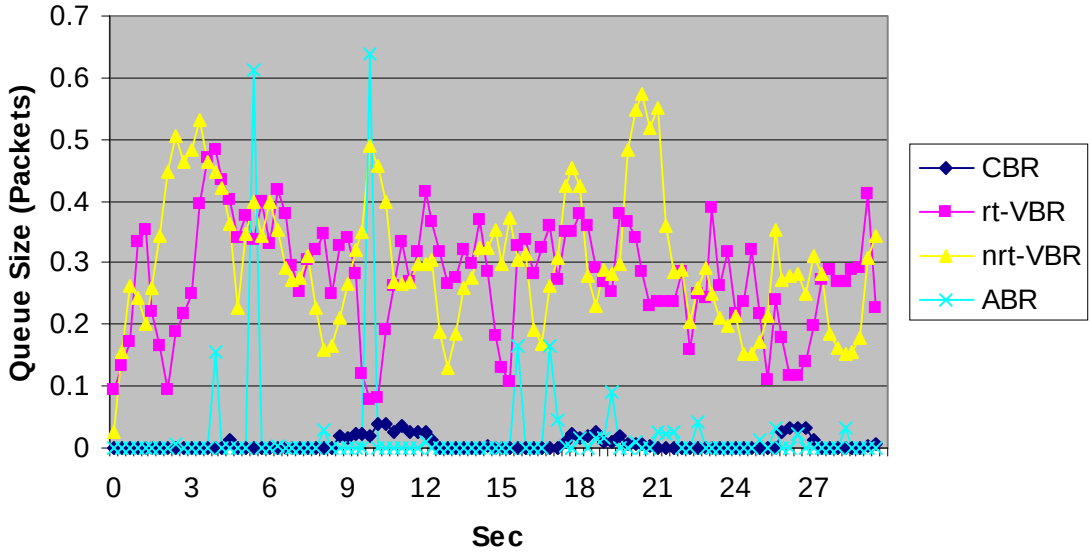


Figure H.9: Queue Size Plots for DQRUMA protocol (Section 6.2, 0.1 Load)

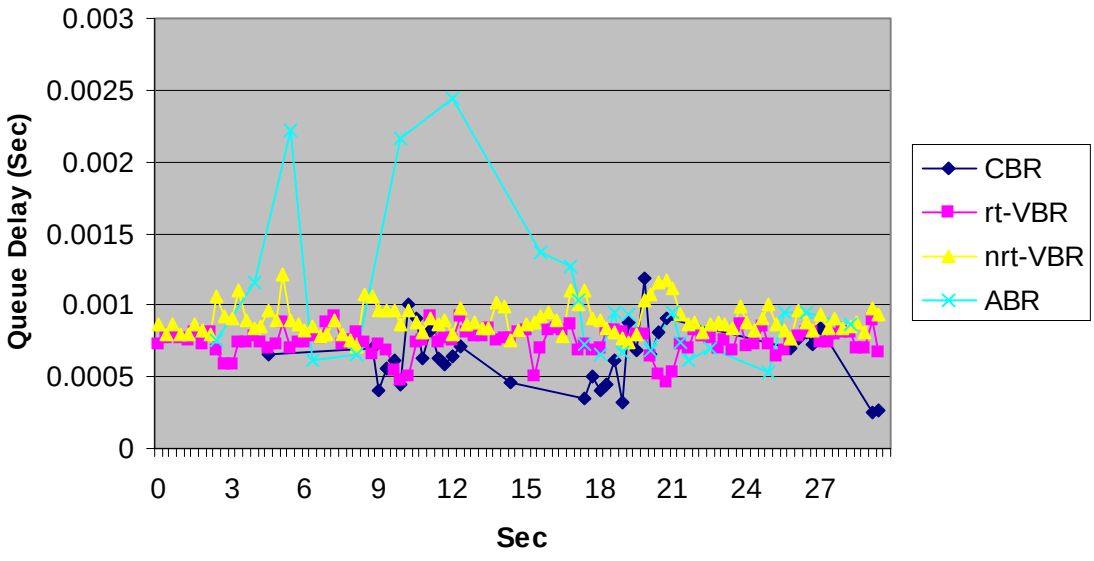


Figure H.10: Queue Delay of Plots for DQRUMA protocol (Section 6.2, 0.1 Load)

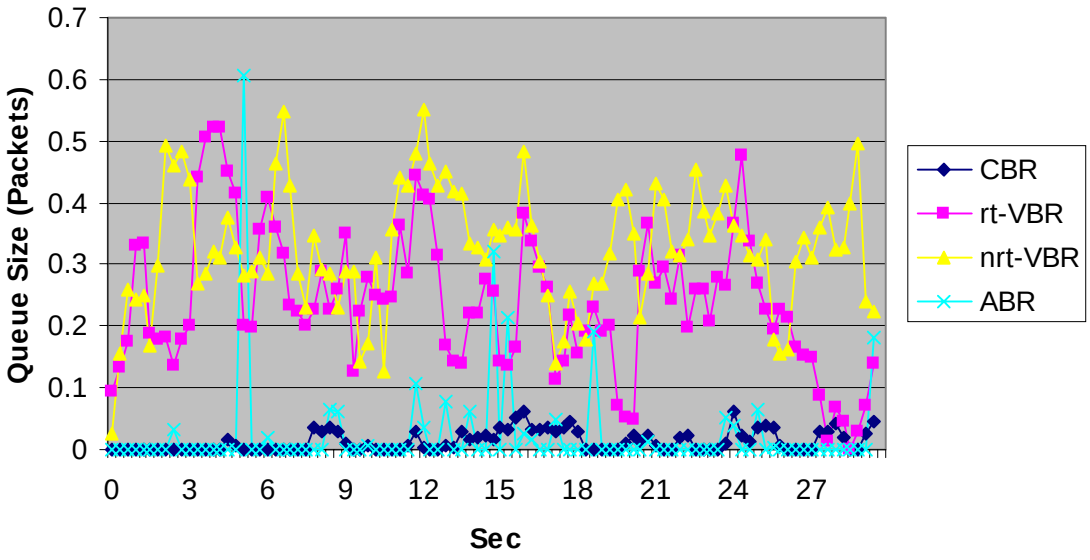


Figure H.11: Queue Size Plots for APMS Protocol (Section 6.2, 0.1 Load)

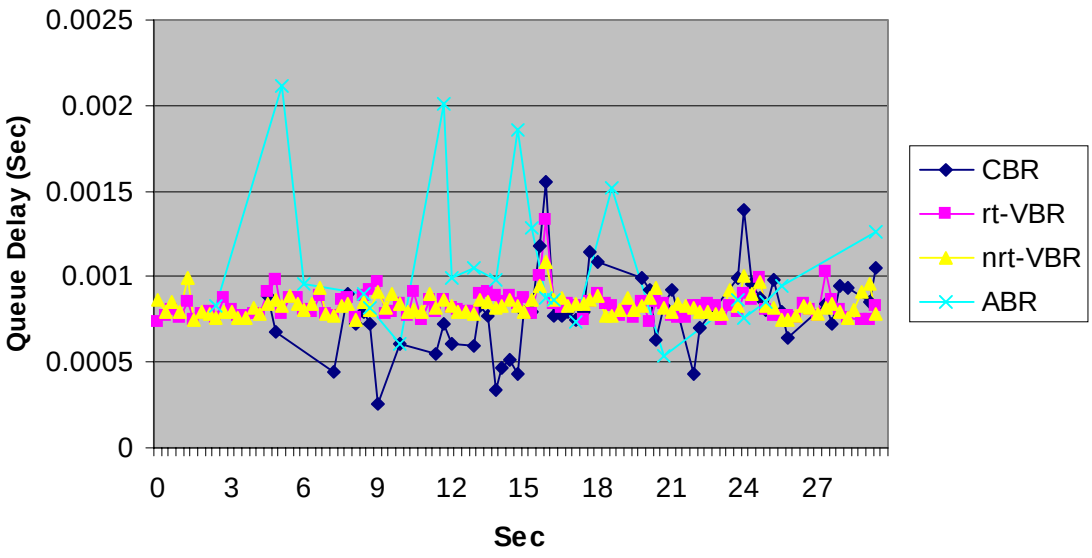


Figure H.12: Queue Delay of Plots for APMS Protocol (Section 6.2, 0.1 Load)

H.2.2 Simulation Corresponding to 0.9 Normalized Load Level

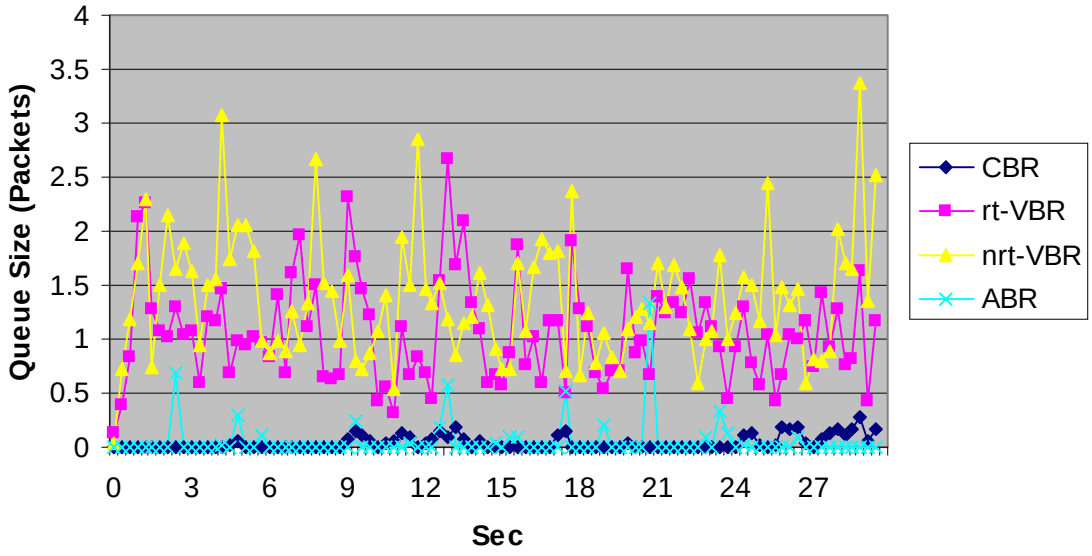


Figure H.13: Queue Size Plots for DQRUMA protocol (Section 6.2, 0.9 Load)

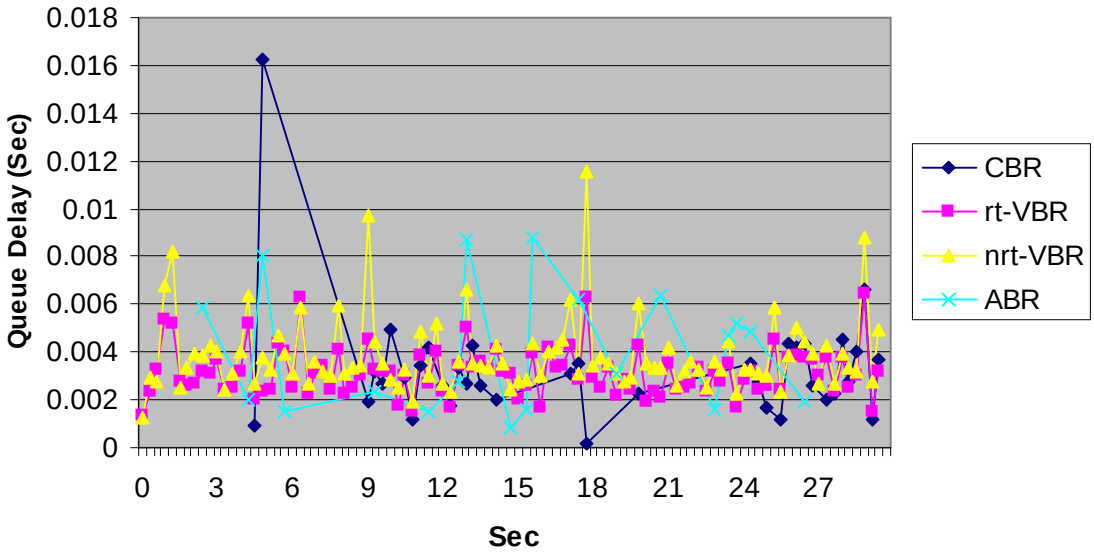


Figure H.14: Queue Delay of Plots for DQRUMA protocol (Section 6.2, 0.9 Load)

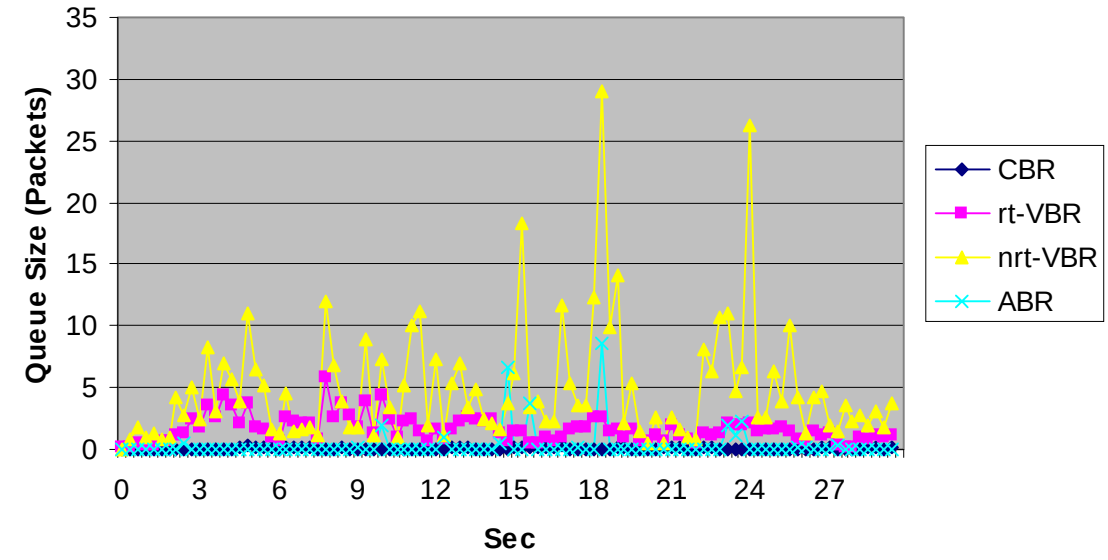


Figure H.15: Queue Size Plots for APMS Protocol (Section 6.2, 0.9 Load)

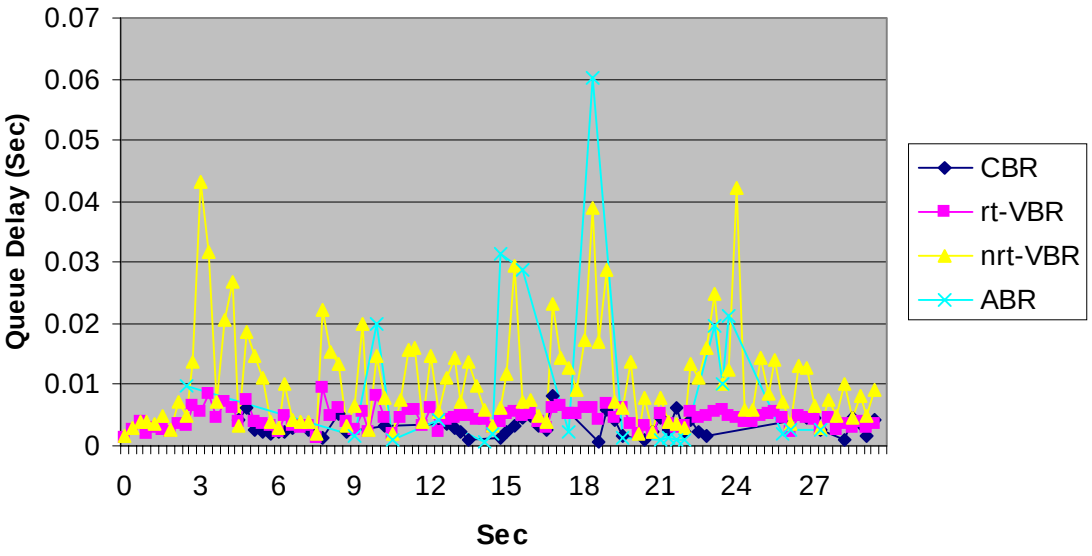


Figure H.16: Queue Delay of Plots for APMS Protocol (Section 6.2, 0.9 Load)

Vita

Tae-In Hyon

Tae-In Hyon was born in Seoul, Republic of Korea, on June 18, 1964. He received the Bachelor of Science and Master of Science degree in Control and Instrumentation Engineering from Seoul National University in February 1987 and February 1989, respectively. He worked at the LG Industrial Systems R&D center from January 1989 to July 1995. His area of work involved designing the monitoring system for the programmable logic controller (PLC) system used in factory automation and designing the graphic user interface for PLC user programs. He produced three patents while working for the LG Industrial systems. In August 1995, he started his Ph.D. program at Virginia Polytechnic Institute and State University. He was awarded a scholarship from LGIS from August 1995 to August 1998. He served as a GTA for computer networks related courses from fall 1998 to spring 2001. His current research interests are MAC protocols for wireless ATM and wireless LANs.

His publications include:

Tae-In Hyon and Nathaniel J. Davis IV, "Adaptive Parallel Multiple Sub-Stream CDMA MAC for Wireless ATM Networks," *OPNETWORK 2000*, Washington D.C., August 2000.

Tae-In Hyon and Nathaniel J. Davis IV, "An Adaptive Multi-Code CDMA MAC Protocol for Wireless ATM," *2001 IEEE International Conference on Third Generation Wireless and Beyond* (Accepted for publication)