A Token Based MAC Protocol for Wireless Ad Hoc Networks

By Yi-Sheng Liu
University of Natal
2003

Submitted in fulfillment of the academic requirement for the degree of Master of Science in Engineering in the School of Electrical, Electronic and Computer Engineering, University of Natal, 2003

ABSTRACT

The emergence of portable terminals in work and living environments is accelerating the progression of wireless networks. A wireless ad hoc network is a new network concept where users establish peer-to-peer communication among themselves independently, in their small area. Since the wireless medium is a shared resource, it becomes an important design issue to efficiently allocate bandwidth among users. MAC (Medium Access Control) layer arbitrates the channel access to the wireless medium and is also responsible for bandwidth allocation to different users, therefore a large amount of research has been conducted on various MAC protocols for ad hoc wireless networks.

This dissertation begins with a survey of existing wireless MAC protocols. The survey includes protocols designed for different network generations and topologies, classifying them based on architecture and mode of operation. Next, we concentrate on the MAC protocols proposed for distributed wireless networks. We propose a new MAC protocol based on a token-passing strategy; which not only incorporates the advantages of the guaranteed access scheme into the distributed type of wireless networks, but also the data rate and delay level QoS guarantees. Data rate QoS provides fairness into sharing of the channel, while delay level QoS introduces a flexible prioritized access to channels by adjusting transmission permission to the current network traffic activities. A simulation model for the protocol is developed and delay and throughput performance results are presented.

To examine the efficiency and performance of the proposed MAC scheme in an ad hoc wireless environment, it is incorporated into the Bluetooth structured network. The model is then simulated in the Bluetooth environment and performance results are presented. Furthermore, an analytical model is proposed and an approximate delay analysis conducted for the proposed MAC scheme. Analytical results are derived and compared with results obtained from computer simulations. The dissertation concludes with suggestions for improvements and future work.

PREFACE

The research work in this dissertation was performed by Yi-Sheng Liu, under the supervision of Professor Fambirai Takawira, at the University of Natal's School of Electrical, Electronic and Computer Engineering. This work was partially supported by Alcatel SA Telecom, THRIP and Telkom South Africa as part of the Centers of Excellence programme at the University of Natal.

Parts of this dissertation have been presented by the author at the SATNAC '00 conference in Wild Coast, Natal, South Africa, and the SATNAC '01 conference in Drakensburg, Natal, South Africa. The entire dissertation, unless specifically indicated to the contrary in the text, is the author's own work and has not been submitted in whole or in part to any other university.

ACKNOWLEDGEMENTS

I would like to give a special thanks to Professor Takawira for his help, guidance and patience as my supervisor over the last two years.

Thanks are also owed to Telkom SA, Alcatel SA Telecoms, THRIP and the University of Natal for their sponsorship of the Centre of Excellence in Radio Access Technologies at the School of Electrical, Electronic and Computer Engineering. Without their financial support, this degree would not have been possible.

Thanks also go to Dr. Xu and my postgraduate colleagues at the Centre for providing assistance for the past two years. They have made the past two years an enjoyable and memorable experience.

Finally, I wish to thank my family. I am forever indebted to them for their love, support and encouragement during the period of my study.

TABLE OF CONTENTS

TITLE PAGE i
ABSTRACT ii
PREFACE iii
ACKNOWLEDGEMENTS iv
TABLE OF CONTENTS v
LIST OF FIGURES AND TABLES viii
LIST OF ABBREVIATIONS xii
LIST OF SYMBOLS xiv

Chapter 1
INTRODUCTION 1-1

1.1 Background 1-1
1.2 General Concept and Architecture of Wireless Networks 1-3
1.3 MAC Protocols for Ad Hoc Networks 1-4
1.4 Original Contribution of This Dissertation 1-5
1.5 Dissertation Organization 1-5

Chapter 2
LITERATURE SURVEY 2-1

2.1 Introduction 2-1
2.2 Overview of Medium Access Control Protocols 2-1
2.3 Wireless MAC Protocols for Distributed (Ad Hoc) Networks 2-3
2.3.1 Collision avoidance with out of band signalling protocols 2-3
2.3.2 Collision avoidance with control handshaking protocols 2-7
2.4 Wireless MAC Protocols for Centralized Networks 2-16
2.4.1 Centralized random access protocols 2-16
2.4.2 Centralized guaranteed access protocols 2-21
2.4.3 Centralized hybrid access with random reservation protocols 2-24
2.4.4 Centralized hybrid access with demand assignment protocols 2-26
2.5 Wireless MAC Protocols for Hybrid Networks 2-32
2.6 Summary 2-35

Chapter 3
THE PROPOSED TOKEN PASSING MAC PROTOCOL 3-1

3.1 Introduction 3-1
3.2 MAC Protocol Description 3-2
3.2.1 CDMA technique 3-2

3.2.2 Node structure 3-3
3.2.3 Token structure 3-3
3.2.4 Quality of Service (QoS) incorporation 3-4
3.2.4.1 Data rate QoS guarantee 3-4
3.2.4.2 Delay level QoS guarantee 3-5
3.2.5 Channel access control 3-6
3.2.5.1 Access control with data rate and delay level QoS guarantees 3-7
3.2.5.2 Token lost and node out of network scenarios 3-9
3.2.5.3 New nodes join the network scenario 3-10
3.3 Simulation Model 3-12
3.3.1 Simulation model for MAC scheme with data rate QoS guarantee 3-12
3.3.2 Simulation model for MAC scheme with data rate and delay level QoS guarantees 3-13
3.4 Simulation Results 3-14
3.4.1 Simulation results for MAC scheme with data rate QoS guarantee 3-15
3.4.2 Simulation results for MAC scheme with data rate and delay level QoS guarantees 3-18
3.5 Summary 3-23

Chapter 4

THE PROPOSED MAC SCHEME WITHIN THE BLUETOOTH PAN **4-1**

4.1 Introduction 4-1
4.2 Bluetooth System 4-1
4.2.1 Master/Slave definition and network topology 4-2
4.2.2 Establishing network connections 4-3
4.2.3 Link types 4-5
4.2.4 MAC protocol 4-5
4.3 Adapting Proposed MAC in Bluetooth Environment 4-6
4.3.1 Wireless Bluetooth environment 4-6
4.3.1.1 Node or token drops out scenario 4-6
4.3.1.2 New nodes join the network 4-8
4.3.2 Node and token structure 4-9
4.4 Simulation Model and Parameters 4-10
4.5 Simulation Results 4-12
4.6 Summary 4-18

Chapter 5

APPROXIMATE DELAY ANALYSIS ON THE PROPOSED MAC PROTOCOL WITH DATA RATE QoS GUARANTEE **5-1**

5.1 Introduction 5-1
5.2 Analytical Model Description 5-2
5.2.1 Data rate QoS incorporation 5-3

5.3 Mean Packet Delay Analysis 5-4
5.4 Approximate Mean Queue Length Analysis 5-5
5.4.1 Derivation of p_{jk} 5-9
5.4.2 Average expressions of p_{jk} 5-10
5.5 Derivation for Token Rotation Time (*TRT*), Interarrival time (α) 5-16
5.5.1 Mean token rotation time ($\overline{TRT}$) 5-16
5.5.2 Mean token interarrival time ($\overline{\alpha}$) 5-18
5.5.3 Variance of token interarrival time (Var[α]) 5-18
5.5.4 Mean and variance of serviced packets (B) 5-20
5.6 Computation Parameters and Results 5-23
5.7 Summary 5-25

Chapter 6

CONCLUSIONS 6-1

APPENDIX A A-1

REFERENCES R-1

LIST OF FIGURES AND TABLES

CHAPTER 1

Figure 1.1: Network layers for wireless networks 1-1

Figure 1.2: Two common topologies in wireless networks: (a) centralized, and (b) distributed 1-3

CHAPTER 2

Figure 2.1: Classification of wireless MAC protocols 2-2

Figure 2.2: Frame structure for TPMA protocol 2-7

Figure 2.3: Four-way handshaking in DFWMAC 2-9

Figure 2.4: Illustration of RBAR protocol 2-13

Figure 2.5: Operation of ISMA and r-ISMA protocols 2-17

Figure 2.6: Illustration of the RAP protocol 2-18

Figure 2.7: Description of RAMA protocol 2-20

Figure 2.8: Zhang's proposal 2-21

Figure 2.9: Disposable token MAC protocol 2-22

Figure 2.10: Acampora's proposal 2-23

Figure 2.11: Upstream frame structure of PRMA and its operation 2-25

Figure 2.12: Details of the DQRUMA protocol 2-27

Figure 2.13: Frame structure of the MASCARA protocol 2-28

Figure 2.14: Uplink frame structure for D-PRMA protocol 2-30

Figure 2.15: Uplink frame structure for TDMA/DR protocol 2-31

Figure 2.16: Frame structure for CRDA protocol 2-32

Figure 2.17: Illustration of MAC protocol used in Bluetooth 2-34

Figure 2.18: Frame structure for C-ICAMA protocol 2-35

CHAPTER 3

Figure 3.1: Node model with data rate QoS guarantee	3-5
Figure 3.2: Node model for the MAC scheme with data rate and delay level QoS guarantees	3-6
Figure 3.3: Graphical display of the described scenarios	3-9
Figure 3.4: System model for the proposed MAC protocol	3-11
Figure 3.5: Overall packet delay when varying QoS's parameter k under different traffic conditions	3-15
Figure 3.6: Comparison of packet delay in two different traffic classes and overall delay in the network that consists of 20 nodes and 8 CDMA code channels with k = 0.5	3-16
Figure 3.7: Comparison of throughput for two traffic classes and overall in the network that consists of 20 nodes and 8 CDMA code channels with k = 0.5	3-16
Figure 3.8: Average code utilization for the network and all classes with the channel reservation parameter N set to half of the codes whereby reserving half of the codes for the high priority classes	3-18
Figure 3.9: Packet delay performance for the network and each class with the channel reservation parameter N set to half of the codes whereby reserving half of the codes for the high priority classes	3-19
Figure 3.10: Packet delay performance for the network and two classes with low data rate (1 and 2) with the channel reservation parameter N set to half of the codes whereby reserving half of the codes for the high priority classes	3-19
Figure 3.11: Packet delay performance for the network and two classes with high data rates (class 3 and 4) with the channel reservation parameter N set to half of the codes whereby reserving half of the codes for the high priority classes	3-20
Figure 3.12: Monitoring the effectiveness of the delay level QoS by observing the packet delay performance of each class with fixed system load and different channel reservation parameter (N) values	3-20
Table 3.1: Token structure	3-3
Table 3.2: Code status table	3-8

CHAPTER 4

Figure 4.1: Bluetooth network scenario: (a) piconet and (b) scatternet 4-2

Figure 4.2: Connection state machine 4-3

Figure 4.3: Node list in the token 4-7

Figure 4.4: Code list in the token 4-7

Figure 4.5: Flow charts for both (a) node or token lost and (b) new cycle scenarios 4-9

Figure 4.6: Average code utilization for the piconet Bluetooth wireless network and all classes with 5% dropout probability, 3 code channels (*NOC*=3) and channel reservation parameter *N* set to the value of one 4-13

Figure 4.7: Packet delay performance for the piconet network and two classes with low data rate (1 and 2) under conditions that 5% dropout probability is assigned, 3 code channels (*NOC*=3) are used and channel reservation parameter *N* set to the value of one 4-13

Figure 4.8: Packet delay performance for the piconet network and two classes with high data rates (class 3 and 4) with 5% dropout probability, 3 code channels (*NOC*=3) and channel reservation parameter *N* set to the value of one 4-14

Figure 4.9: Monitoring the effectiveness of the delay level QoS by observing the packet delay performance of each class, this is achieved by fixing the *NOC* value to three and varying channel reservation parameter (*N*) values 4-14

Figure 4.10: Monitoring the effect on delay performance when adjusting the dropout probability of the token and nodes from 5 % to 50% under the conditions that *NOC* = 3 and channel reservation parameter *N* = 1 4-15

Figure 4.11: Monitoring the effect on throughput performance when adjusting the dropout probability of the token and nodes from 5 % to 50% with *NOC* = 3 and the channel reservation parameter *N* = 1 4-15

Table 4.1: Proposed token structure for Bluetooth network 4-10

CHAPTER 5

Figure 5.1: A multiple queue system with M multiple cyclic servers 5-2

Figure 5.2: Queue model of the proposed system 5-3

Figure 5.3: Steady state probabilities 5-8

Figure 5.4: Comparison of approximate packet delay analysis to simulation, $N = 20, M = 8$ 5-24

Figure 5.5: Comparison of approximate packet delay analysis to simulation, $N = 30, M = 8$ 5-24

APPENDIX A

Figure A.1: Example of hidden and exposed terminal problems A-2

LIST OF ABBREVIATIONS

Acronym For MAC Protocols	Description
ABROAD	Adaptive Broadcast
BSMA	Broadcast Support Multiple Access
BTMA	Busy Tone Multiple Access
CATS	Collision Avoidance Transmission Scheduling
CDMA	Code Division Multiple Access
C-ICAMA	Centralized intelligent channel assigned multiple access
CRDA	Collision Resolution and Dynamic Allocation
CSMA	Carrier Sense Multiple Access
DBTMA	Dual Busy Tone Multiple Access
DCA	Dynamic Channel Assignment
DFWMAC	Distributed Foundation Wireless MAC
DQRUMA	Distributed-Queuing Request Update Multiple Access
DTMP	Disposable Token MAC Protocol
D-PRMA	Dynamic Packet Reservation Multiple Access
FAMA	Floor Acquisition Multiple Access
FH-CDMA	Frequency hopping code division multiple access
F-RAMA	Fair-resource auction multiple access
IPRMA	Integrated packet reservation multiple access
ISMA	Idle Sense Multiple Access
MACA	Multiple Access with Collision Avoidance
MASCARA	Mobile Access Scheme based on Contention and Reservation
PRADOS	Prioritized regulated allocation delay orientated scheduling
PRMA	Packet Reservation Multiple Access
RAMA	Resource Auction Multiple Access
RAP	Randomly Addressed Polling
RBAR	Receiver-Based Auto Rate
RI-BTMA	Receiver Initiated-BTMA
R-ISMA	Reservation idle sense multiple access
TPMA	Three Phase Multiple Access
TDMA/DR	Time Division Multiple Access with Dynamic Reservation
WCD	Wireless Collision Detect

Acronym	Description
MAC	Medium Access Control
OSI	Open System Interconnect
QoS	Quality of Service
RTS/CTS	Request To Send/Clear To Send
ACK	Acknowledgement
MUD	Multi User Detector

ID	Identity
NOC	Number of Codes
MMPP	Markov Modulated Poisson Process
PAN	Personal Area Network
TDD	Time Division Duplex
SCO	Synchronous Connection Oriented link
ACL	Asynchronous Connectionless link
RTG	Randomly Timed Gated
PDF	Probability Distribution Function
CDF	Cumulative Distribution Function
BTt	Transmit busy tone
BTr	Receive busy tone
CD	Carrier detect
FT	Feedback tone
RDI	Receiver detection interval
IDI	Idle detection interval
CR	Collision report
RA	Receiver available
DIFS	Distributed inter frame space
SIFS	Short inter frame space
NAV	Network allocation vector
NACK	Negative acknowledgement
RTB	Request to broadcast
CTB	Clear to broadcast
NCTB	Negative clear to broadcast
SCH	Signalling channel
DCH	Data channel
BCH	Broadcast channel
LRB	Link reservation beacon
RUB	Request uni-cast beacon
CUB	Concur uni-cast beacon
RMB	Request multicast beacon
RBB	Request broadcast beacon
SBB	Stop broadcast beacon
BS	Base station
IS	Idle signal
ISA	Idle signal with acknowledgement
RP	Reservation packet
PS	Polling signal
FDD	Frequency division duplex
ATM	Asynchronous transfer mode
CBR	Constant bit rate
VBR	Variable bit rate
UAV	Unmanned aerial vehicle
FDDI	Fiber Distributed Data Interface

LIST OF SYMBOLS

Symbol	Description
M	Number of servers in the network
N	Number of queues
λ_i	Packet arrival rate to queue *i*
s	Length of a packet
T_p	Service time of a packet
$\overline{h_i}$	Mean token walk time between two queues
C	Exponentially distributed random variable
θ	Mean of C
n	Number of service epochs
X_n	Number of packets in queue *i* just before the *nth* service cycle
B_{n-1}	Number of packets that received service in the *(n-1)th* service cycle
A_{n-1}	Number of packets that arrived during the *(n-1)th* and *nth* cycles
α	Token inter-arrival time
π_j	Steady state probability of the queue *i* being in state *j*
Θ	State transition probability matrix
p_{jk}	Probability that the queue size is *k* immediately before the *nth* service cycle given that the queue size was *j* before the *(n-1)th* service cycle
$P_A(k)$	Poisson distributed pdf
$P_c(C)$	Exponential pdf
$P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)$	Gamma pdf
$TRT_{i,j}$	Time between successive visits of sever *j* to queue *i*.
$\overline{B_i}$	Mean number of packets serviced from queue *i* during the token interarrival time
$\overline{\alpha}$	Mean token inter-arrival time
ρ	Correlation variable for queue length (*X*) and serviced packets (*B*)
$\overline{S_i}$	Waiting time of a tagged packet arriving at queue *i*
$\overline{W}$	Mean waiting time between packet arrival and the time it is removed from the queue by a server
$\overline{X}$	Mean queue length
$\overline{B}$	Mean number of packets that received service
$\overline{B^2}$	Second moment of *B*

CHAPTER 1

INTRODUCTION

1.1 Background

The latest trend in the modern communication world is the desire for mobility; this can be seen from the tremendous increase in mobile phone subscribers in recent years [ITU, 2001]. 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. The most typical example is witnessed in the evolution from analogue technology based first-generation, to digital technology based second-generation mobile networks. Now, with the emergence of third-generation mobile networks [Prasad, 1998], broadband wireless technology is possible, providing a bandwidth sufficient for supporting multimedia traffic.

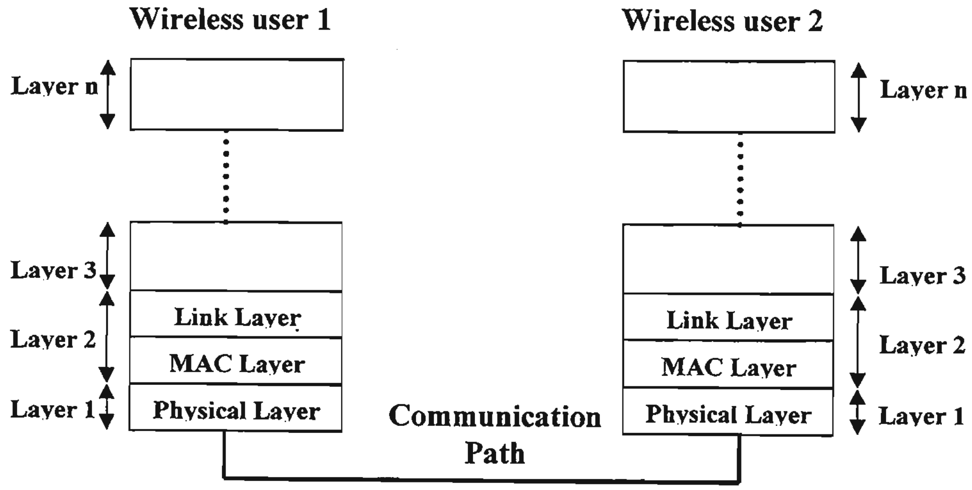


Figure 1.1: Network layers for wireless networks

Built upon the OSI model [Spragins, 1991]; a wireless network architecture is similar to that of the wired network. Figure 1.1 illustrates the layered architecture of the wireless network, normally consisting of three or more layers. The lowest layer in the model is the physical layer. It provides the physical medium for the information flow which it is responsible for activating, maintaining, and deactivating the physical circuit between the sender and the receiver. The second layer, the data link layer, consists of two sub-layers: the medium access control (MAC) layer and the link layer. The MAC layer is responsible for controlling the access to the radio medium. It decides when, how, and who should transmit data. The link layer is responsible for controlling the link between communicating users. Depending on the type of the wireless network, the third layer is generally the transport layer, providing transparent transfer of data between two users.

In wireless networks, the available medium (in frequency or time) is scarce; its effective utilization is considered as the most important design factor, therefore the MAC protocol is a critical part of the network stack which determines to a large extent the correct and efficient operation of the wireless network.

1.2 General Concept and Architecture of Wireless Networks

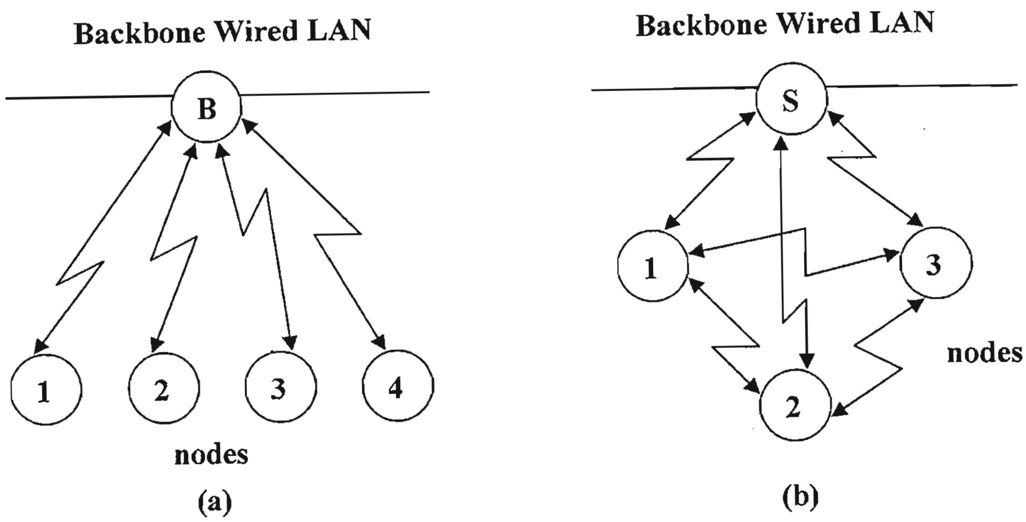


Figure 1.2: Two common topologies in wireless networks: (a) centralized, and (b) distributed

The basic topologies commonly used in wireless network are centralized and distributed configurations, as shown in Figure 1.2. Detailed descriptions of the configurations follow.

Centralized wireless networks:

For the centralized wireless network structure shown in Figure 1.2(a), the base station **(B)** acts as the interface between wireless or wired networks and the wireless nodes. Any communication from one node to another node goes through the base station, hence there is no provision in the centralized topology for direct peer-to-peer communication. The base station controls the wireless nodes and monitors each node's transmission. Downlink transmissions (from base station to the wireless nodes) are broadcast, and can be received by all nodes on the network. The up link (from wireless node to the base station) is shared by all the nodes and is therefore a multiple access channel. Thus the base station is involved in managing the wireless nodes' access to the channel by controlling network's allocated bandwidth.

Distributed networks:

Distributed networks are also called ad hoc networks [Chandra, 2000]. In such network, as shown in Figure 1.2(b), all wireless nodes can communicate with one another directly with no pre-existing infrastructure installed. The peer-to-peer capability of this topology provides an instant connectivity without any need for a centralized controller. In a distributed topology, wireless nodes have a wireless interface and the information is exchanged between one another in a distributed fashion. The connection to a backbone network is normally provided by a server node (S) that acts as a bridge or gateway, as shown in Figure 1.2(b). An important advantage of this topology is that the network does not collapse when one of the nodes is powered down or moves away, since it has no central administration.

1.3 MAC Protocols for Ad Hoc Networks

As discussed in section 1.2, a wireless ad hoc network is a system of wireless mobile nodes dynamically self-organizing, in arbitrary and temporary network topologies. Since the environment is wireless and the air interface provides less capacity than a cable, the efficiency of the medium access protocol is fundamental. Wireless ad hoc networks rely on a common transmission medium and the transmissions of the network must be coordinated by the Medium Access Control (MAC) protocol. Two methods are implemented to achieve this coordination. In the first method, the coordination can be provided by the medium itself, using the carrier sensing to identify the channel's state (idle or active). The second method achieves coordination by means of control information carried by a control message travelling along the medium.

The majority of MAC protocols used in this type of network implement a random access scheme as the method of sharing the common channel. A typical example is the carrier sense multiple access (CSMA) scheme. However, it is known from [Kleinrock, 1975] that with a CSMA based scheme, the performance is degraded by the hidden and exposed terminal problems. In a wireless ad hoc network there is no guarantee that some terminals may not be hidden from the other terminals. Chapter 2 discusses some of the MAC protocols proposed to alleviate the problem.

1.4 Original Contribution of This Dissertation

The research pursued in this dissertation is aimed at proposing and developing a new type of medium access control (MAC) protocol for distributed wireless networks, built based on the concept of the token passing technique. Also, quality of service (QoS) guarantees are incorporated in the conceptual MAC scheme proposed. This proposed MAC scheme is then simulated and its performance is evaluated. An approximate packet delay analysis is also conducted in this dissertation and the analytical result compared with simulation result.

1.5 Dissertation Organisation

The dissertation is organized into six chapters. In this chapter, an introduction to the structure of the wireless network is provided, and concepts of centralized and distributed wireless networks briefly discussed. The objectives of this research are stated in section 1.3 with the remainder of this dissertation organized as follows:

Chapter 2 presents a literature survey of relevant work that has been published on the subject of wireless MAC protocols. Based on the topology that is implemented, we classify the wireless MAC protocols into three different categories. The first and second categories discuss the MAC protocols used in the centralized and distributed wireless networks respectively. Third category describes the MAC protocols used in hybrid distributed-centralized wireless networks such as Bluetooth networks.

In chapter 3, a new MAC protocol is introduced for distributed wireless networks. The proposed MAC protocol is based on the token passing scheme, and incorporates two types of quality of service: data rate and delay level QoS guarantees. The simulation model is then presented, along with the environment for the proposed MAC scheme. The simulation is event driven developed using the C++ Builder 4 software package. The performance of the scheme is then presented and evaluated.

In chapter 4, the implementation of the proposed MAC scheme in the Bluetooth piconet wireless environment is considered. A brief discussion of the Bluetooth network, and a more comprehensive and formal description of techniques that the proposed MAC scheme used to resolve errors like nodes drop out of network are presented and finally the simulation model and parameters are described and the results evaluated.

In chapter 5, an approximate delay analysis of a generalized version of the proposed MAC scheme is presented. To conduct the delay analysis, the proposed MAC scheme is treated as a multiple server network with multiple queues in the network. To verify the analysis, the analytical result is presented and compared with the simulation result.

Chapter 6 concludes this dissertation with a summary of the research findings, including important concepts and techniques behind this research effort. Finally, the chapter presents recommendations for future research.

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

Wireless medium access control (MAC) protocols have been heavily researched and an abundance of protocols proposed and implemented for different types of architectures, different applications and different media. This chapter summarizes various existing and proposed wireless MAC protocols and classifying them based on method of resource sharing and their multiple access technology.

This chapter is organized as follows. Section 2.2 discusses the categories of MAC protocols whilst section 2.3 proceeds to classify MAC protocols that have been proposed for wireless networks. Sections 2.4 to 2.6 discuss different classes of wireless MAC protocols. Section 2.4 presents MAC protocols for wireless ad hoc networks and section 2.5 discusses MAC protocols proposed for centralized networks. In section 2.6, wireless protocols that have been published for ad hoc centralized networks are discussed.

2.2 Overview of Medium Access Control protocols

In general, MAC schemes may be classified based on the bandwidth allocation scheme which may be implemented using a centralized controller or in a distributed manner. Saadawi [Saadawi, 1994] has classified the MAC schemes into five categories:

- Fixed assignment
- Random access
- Demand assignment with distributed control

- Demand assignment with centralized control
- Hybrid modes

Medium access control protocols for wireless networks are more complicated than the protocols for wired networks as one needs to take into consideration the wireless aspects when designing a wireless MAC protocol. These wireless aspects include location-dependent carrier sensing, time varying channel and half duplex operations, and many protocols have been proposed in this field since the 1970s.

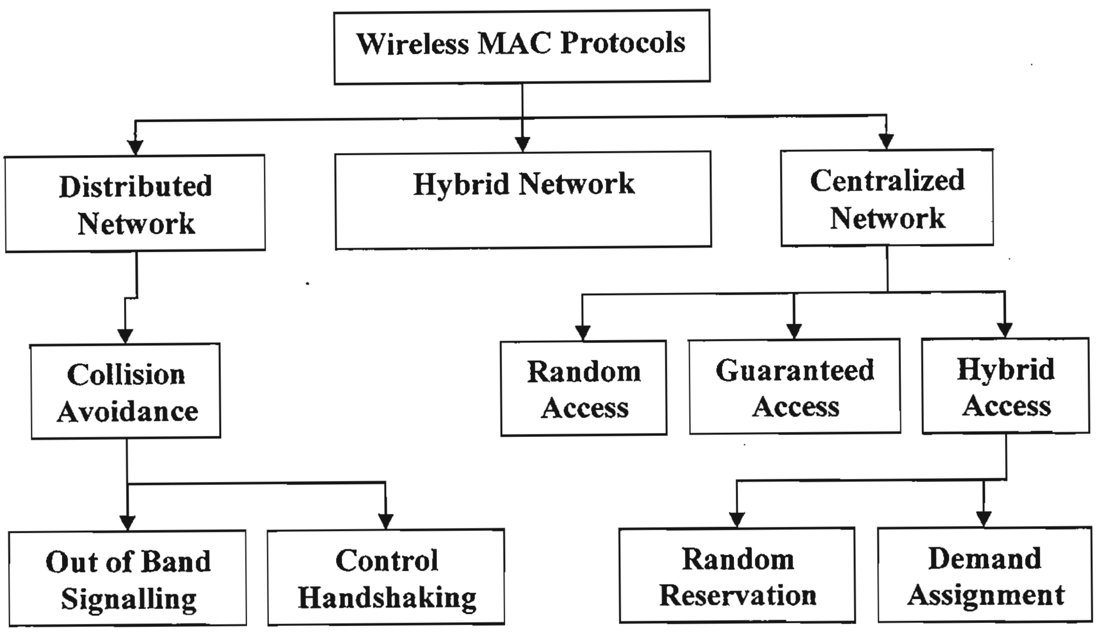


Figure 2.1: Classification of wireless MAC protocols

Chandra [Chandra, 2000] has logically classified wireless MAC protocols based on the type of network architecture that the protocols are designed for. The protocols are first classified based on the network topology they use: distributed, and centralized. However, we have expanded the classification tree by inserting a new branch called hybrid network topology, see Figure 2.1. Both the distributed and centralized topologies were discussed in Chapter 1. The hybrid topology implements both of these two, where it presents protocols used in distributed ad hoc networks with centralized topology.

MAC protocols for distributed networks can be further classified into out of band signalling and control handshaking protocols, based on their mode of operation. Both of these classes avoid collisions by sensing the channel before transmission.

MAC protocols for centralized networks can be divided into three classes: random access, guaranteed access, and hybrid access. For random access protocols, the nodes contend access to the medium. Guaranteed access protocols allow nodes access to the medium in an orderly manner, usually in a round-robin fashion. Hybrid access protocols can be further divided into two classes: random reservation and demand assignment. Random reservation protocols contend and reserve the bandwidth in a free fashion. Demand assignment protocols transmit a request and wait for the base station to assign a bandwidth to them. MAC protocols for each class are discussed in the subsequent sections.

2.3 Wireless MAC Protocols for Distributed (Ad hoc) Networks

An ad hoc network has no centralized structure, it is built up based on a set of nodes that communicate with each other without any pre-existing infrastructure. Collision avoidance algorithms are comprehensively used in these types of networks. The algorithms are implemented using one of two approaches: the first is called out of band signalling, and the second is known as control handshaking.

2.3.1 Collision avoidance with out of band signalling protocols

A technique called out-of-band busy-tone signaling is used when designing busy-tone protocol. It uses a very narrow frequency band (or channel) to carry the busy-tone signal, which warns the surrounding nodes not to transmit. The nodes in the network are required to listen to the busy-tone channel before they transmit any packet. If there is a transmission on the busy-tone channel, nodes are prohibited from transmitting and must defer their transmissions to a later time according to the scheme used by their protocol. The node that is transmitting usually transmits a busy tone in the busy-tone channel. The busy-tone based MAC protocols are discussed below.

Busy Tone Multiple Access (BTMA)

The typical example of the busy-tone protocol is BTMA [Tobagi, 1975]. This protocol consists of two channels; one is used for busy-tone signaling, and the other for data transmission. When a node needs to transmit data, it transmits its data packet in the data channel according to the slotted-Aloha protocol. When a neighboring node hears this ongoing transmission, it will transmit a busy tone in the busy-tone channel, and if a node who also needs to transmit data hears the busy-tone, it has to back off until the busy-tone ends. The busy-tone creates a double radius inhibition zone, and all nodes within this zone are inhibited from transmission. This eliminates the hidden nodes that surround the host node and the target node, but increases the number of exposed nodes.

Receiver Initiated-BTMA (RI-BTMA) [Wu, 1988] was later introduced to alleviate the problem of the large number of exposed nodes produced by the BTMA protocol. In RI-BTMA, the host node sends a short message to the intended receiver node. Any node that hears this ongoing transmission decodes the short message and then identifies the intended receiver. If the node is the intended receiver, it broadcasts a busy tone in the busy-tone channel. The busy tone acts as an acknowledgement to the host node, and it can then begin its data transmission. Again, other nodes back off after hearing the busy tone. RI-BTMA does not completely eliminate the hidden nodes but it does minimize the number of exposed nodes.

Dual Busy Tone Multiple Access (DBTMA)

DBTMA [Deng, 1998] is the improved version of BTMA and was proposed to address the loss of channel utilization in the RTS/CTS-based BTMA networks. In the DBTMA scheme, the single common channel is divided into two sub-channels: a data channel and a control channel. Two busy tones are assigned on the control channel: BT_t (transmit busy tone), which show that a node is transmitting on the data channel and BT_r (receive busy tone), which shows that a node is receiving on the data channel.

Unlike BTMA protocol, if a host node has data to transmit, it first checks the control channel for the BT_r signal before it attains the channel. If there is no BT_r signal (no one is receiving data in host node's transmission area), the host node then sends an RTS packet on the control channel. If the target node receives the RTS packet, it then senses the BT_t busy tone signal. If there is no BT_t signal (no one is transmitting on the data channel in target node's transmission area), the target node replies with a CTS packet and switches on the BT_r signal to inform surrounding nodes that it is receiving data.

Once the host node has received the CTS packet from target node, it switches on the BT_t signal to inform the surrounding nodes that it is transmitting, and then begins its data packet transmission on the data channel. The host node switches off the BT_t signal immediately after the transmission and if the target node successfully receives the data, it then switches off the BT_r signal.

Wireless Collision Detect (WCD)

To alleviate the problem of hidden and exposed nodes, the WCD protocol [Gummalla, 2000] was introduced. This protocol is designed for a short radius network (< 50m), and it splits the frequency channel into a data-channel and a feedback channel, where the feedback channel splits into a further two logical channels called the carrier detect (CD) channel and the feedback-tone (FT) channel.

A node can only operate in one of two modes, the data reception mode and the data transmission mode. In the data reception mode, the node listens to the data-channel. If a transmission is detected, it sets up the CD signal. By receiving the header from the channel, the node then determines the destination of the packet, achieved by matching the destination address with its own address. If the two match, the node is the target node. The target node then stops setting on the CD signal and sets up the FT signal.

However, if the address does not match, the node simply setting off the CD signal without sets on the FT signal.

In data transmission mode, the node samples the FT channel and CD channel before transmission. A back off algorithm will be implemented by the node that intends to transmit if it detects a signal from the FT channel, and if the node detects a signal the CD channel, it will sample the channel again after a Receiver Detection Interval (RDI). RDI is the time required to determine the destination of the current transmission signals on the feedback channel.

If no signal is detected on either channel, the node will sense the CD channel for an Idle Detection Interval (IDI). IDI is the round trip time plus the time to both detect a carrier and to transmit the feedback signal. A transmission attempt will be made if no signals are set up in CD channel during that period. After making the transmission attempt, the node will wait for a RDI before it samples the FT channel for feedback. If the FT signal is not sensed, it will assume that a collision has occurred and it will abort its transmission. The node then backs off for a random period before attempting to transmit again.

Three Phase Multiple Access (TPMA)

TPMA [Hou, 2001] uses a single channel for out-of-band signalling. The channel is divided into fixed size frames as shown in Figure 2.2, each consisting of two slots: an elimination slot (to resolve contention) and a data slot. The elimination slot is further divided into M mini-slots, each mini-slot consisting of three phases: Request to send (RTS) phase, collision report (CR) phase and receiver available (RA) phase. Both the RTS and RA phases perform similar functions to the RTS/CTS dialogue. The CR phase is used to indicate a collision occurrence if a node receives more than one RTS packet in RTS phase.

When a host node wants to transmit a data packet, it sends an RTS packet in the first mini-slot of the elimination slot in the RTS phase. If other nodes that surround the host node also send RTS packets in the first mini slot, the target node that receives multiple RTS packets will send a CR packet in the CR phase to indicate a collision. If the host node detects a CR packet after sending the RTS, it knows a collision has occurred. The host node then attempts to send the RTS in the next mini-slot, with contending probability p. If the target node only receives one RTS packet in RTS phase, it will send a RA packet to acknowledge the RTS request. On receiving the RA packet from the target node, the host node begins its data transmission.

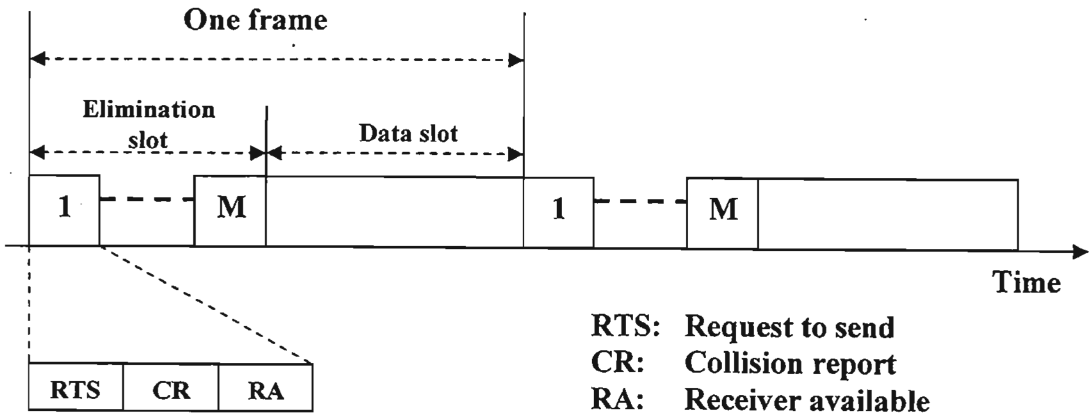


Figure 2.2: Frame structure for TPMA protocol

2.3.2 Collision avoidance with control handshaking protocols

Control handshakes are used when designing a control handshaking type of collision avoidance protocol. Control handshakes are defined as short packets carrying messages to inform the nodes in the network about the packet transmissions. The handshake technique is similar to the busy tone technique, but carries more information. Three types of handshakes are commonly used by the collision avoidance protocols: request to send (RTS), clear to send (CTS), and acknowledgement (ACK).

RTS is usually sent by a host node to a target node. The main purpose of the RTS is to inform the target node that the host node has data to transmit, and also to ensure the target node is available,. to avoid collisions. CTS is used by the target node to reply to the host node after receiving an RTS, and ACK is used to inform the host node that its data has been successfully transmitted. The handshakes are also used to warn the surrounding node that a transmission is ongoing, and they must defer their transmission if the handshakes are heard. The collision avoidance protocols usually operate in a single channel mode (single frequency band), with handshakes being exchanged within the channel. Multi-channel protocols also exist in which multiple frequency bands are used for handshakes and data transmission.

Multiple Access with Collision Avoidance (MACA)

MACA [Karn, 1990] uses a three-way handshake mechanism to avoid collisions. When the host node wants to transmit data to the target node, it sends an RTS packet to the target node. All nodes that surround the host defer their transmissions when they hear the RTS. If the target node receives the RTS successfully, it responds by broadcasting a CTS packet. The CTS is used to warn the nodes surrounding the target node not to transmit. On receiving the CTS, the host node assumes that the channel is acquired and sends its data to the target node.

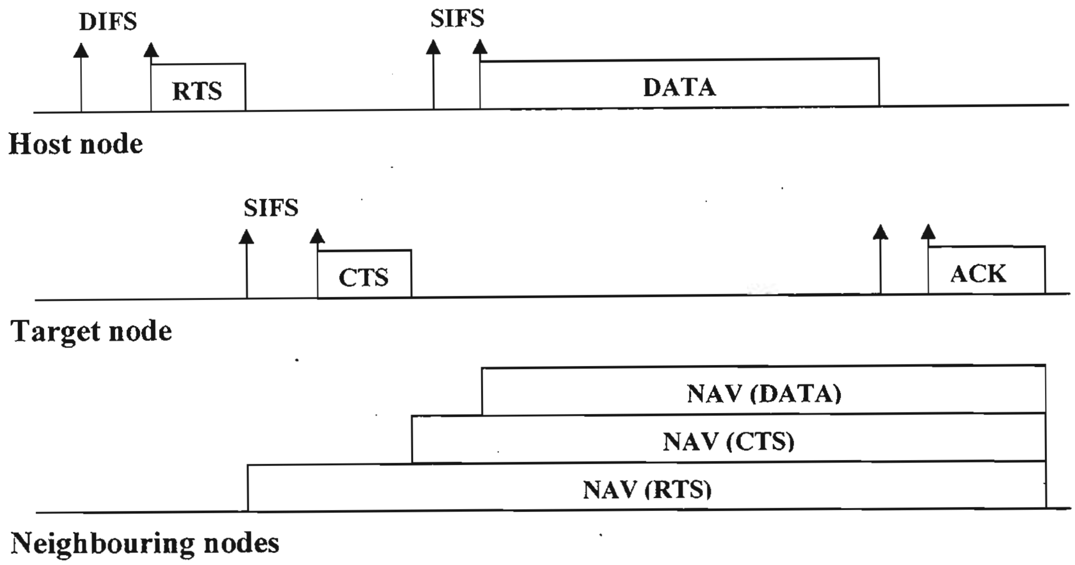


Figure 2.3: Four-way handshaking in DFWMAC

Distributed Foundation Wireless MAC (DFWMAC)

DFWMAC [Crow, 1997] is a derivative of the MACA protocol and is the basic access protocol for distributed systems described by the IEEE 802.11 wireless LAN standard. The DFWMAC protocol consists of four handshakes, RTS-CTS-DATA-ACK. The handshaking is illustrated in Figure 2.3. When the host node wishes to transmit data to the target node, it must detect the channel idle for a specified time interval before attempting an RTS transmission. This interval is called the Distributed Inter-Frame Space (DIFS). When the DIFS expires, the host node then tries to acquire the channel by sending an RTS packet.

If the RTS packet is received by the target node, it waits for a Short Inter-Frame Space (SIFS) interval before sending a CTS packet. Once the host receives the CTS, it waits for a SIFS interval before beginning its data transmission. If the data is received by the target node successfully, it waits for a SIFS interval before sending an ACK packet back to the host node. The SIFS packet length is shorter than DIFS's packet which provides a priority scheme in favor of transmission attached to SIFS. The inter-frame space is used as a form of collision avoidance.

Neighboring nodes listen to the channel traffic and predict the length of the transmissions based on virtual carrier sensing. Virtual carrier sensing is achieved by using time fields in the packets, which indicate to other nodes the duration of ongoing transmission. This time field is called the Network Allocation Vector (NAV) and indicates the duration of the current transmission. All nodes that receive RTS or CTS packets back off NAV amount of time before sensing the channel again.

Broadcast Support Multiple Access (BSMA)

BSMA protocol [Tang, 2000] is an extension of the IEEE 802.11 protocol. The objective of this protocol is to provide an efficient broadcasting ability by incorporating both the collision avoidance scheme and the four handshakes control of IEEE 802.11. It relies on negative acknowledgement (NACK) to deliver broadcasted packets.

The host node that has a packet to broadcast first goes through the collision avoidance phase that is identical to IEEE 802.11. The host then sends out an RTS packet to its neighbors and sets the WAIT_FOR_CTS timer. The step is repeated if the host node doesn't receive a CTS packet by the time the timer expires. The nodes that receive this CTS packet and are not part of the transmission then change their state to a prohibited state until the end of this transmission, predicted from the NAV.

After successful reception of the RTS packet, the neighboring nodes that are not in a prohibited state transmit a CTS packet and set the WAIT_FOR_DATA timer. Upon receiving the CTS packet, the host sends its data and sets the WAIT_FOR_NACK timer. If the target node does not receive the data successfully from the host node before the WAIT_FOR_CTS timer expires, it then transmits a NACK to the host node. If the host node does not receive any NACK before the WAIT_FOR_NACK timer expires, it assumes that the transmission has been successful.

Floor Acquisition Multiple Access (FAMA)

In FAMA [Fullmer, 1995], a node must acquire the surrounding channel "floor" before transmitting its data. To acquire the floor, the host node first transmits an RTS to its neighbors, and if the target node receives the RTS, the CTS message is sent back to the host node. The host node then begins sending its data packets. The CTS also serves to warn other nodes against transmitting to the target node.

An improved version of FAMA was introduced in 1999 called Floor Acquisition Multiple Access with Non-persistent Carrier Sensing (FAMA-NCS) [Fullmer, 1999]. It provides better collision avoidance by modifying the length of the CTS. In the new protocol, the length of a CTS message has been expanded, now including four different periods: time required for transmitting an RTS message, maximum roundtrip time, turn around time, and the processing time. Once the target node begins transmitting CTS, its neighbors who are transmitting an RTS will receive at least a portion of the CTS and back off their transmission. Consequently, this allows the host node to transmit its data packets without colliding with the traffic generated from the neighbors of the target node.

Adaptive Broadcast (ABROAD)

ABROAD protocol [Chlamtac, 2000] divides the frequency channel into frames, each frame consisting of N sub-frames, where N is the number of nodes in the network. Each one of the nodes is assigned a sub-frame made up of five periods; four signal periods and one data period. When a node has a packet to send in its assigned sub-frame, it transmits a request-to-broadcast (RTB) packet in the first period of the sub-frame.

Every surrounding node responds with a clear-to-broadcast (CTB) packet in the second period after they receive the RTB packet. The broadcasting the CTB packet informs all nodes in a two-hop radius not to transmit in that sub-frame. The host node waits for the other two signal periods to pass through and if an idle is observed in these subsequent signaling periods, the host node then broadcasts its packet.

For a node that does not have data to transmit but is allocated an ongoing sub-frame, an idle will be observed in the first two periods. When this occurs, the vacant sub-frame is relocated to a node that wishes to transmit by transmitting an RTB packet in the third period of the sub-frame. However, if its neighbors detect a collision, they will send a negative-CTB (NCTB) packet in the fourth period. The node may only transmit its data packet when no NCTB packets are detected.

ABROAD protocol has resolved the hidden sender problem but does not eliminate the hidden receiver problem. As this protocol focuses on broadcasting algorithms, the exposed sender problem does not pose a problem and the protocol claims to support uni-cast service. However, the author did not provide sufficient evidence to verify this claim.

Receiver-Based Auto Rate (RBAR)

The RBAR [Vaidya, 2000] protocol is based on three-way collision avoidance handshake. In RBAR, the RTS/CTS handshake is modified to allow the target node (receiver) to choose the data rate at which the packet will be transmitted. The RTS/CTS packets consist of two fields: data rate and data packet size. In the RTS dialogue, the rate field carries the data rate that the host node (sender) intends to use for the data packet, whereas the CTS dialogue carries the actual rate that will be used, selected by the receiver.

The protocol is illustrated in Figure 2.4. In this scenario, host node H wants to transmit a data packet of size n to target node T, with nodes A and B in range of H and T, respectively. Using RBAR, node H first chooses a data rate r1 and stores it with the size of data packet n in the RTS packet before sending the RTS to node T. As node A also detects the RTS packet, it then uses r1 and n to calculate the duration that it needs to back off. Once the node T has received the RTS packet, it uses channel quality estimation and rate selection technique to select the best rate r2 for the channel condition. Node T then sends the CTS packet that contains r2 and n to node H. Node B, in this case, has also detected the CTS packet, and will then calculate the duration of its back off period using r2 and n.

Node H responds to the CTS packet by placing r2 into the header of its data packet and transmitting the packet at the selected rate. However, if r1 is not equal to r2, node H will use a unique header to signal the rate change. By detecting the unique header from the data packet, node A then recalculates the back off period accordingly. Node T sends back the ACK packet once the data packet has been successfully received.

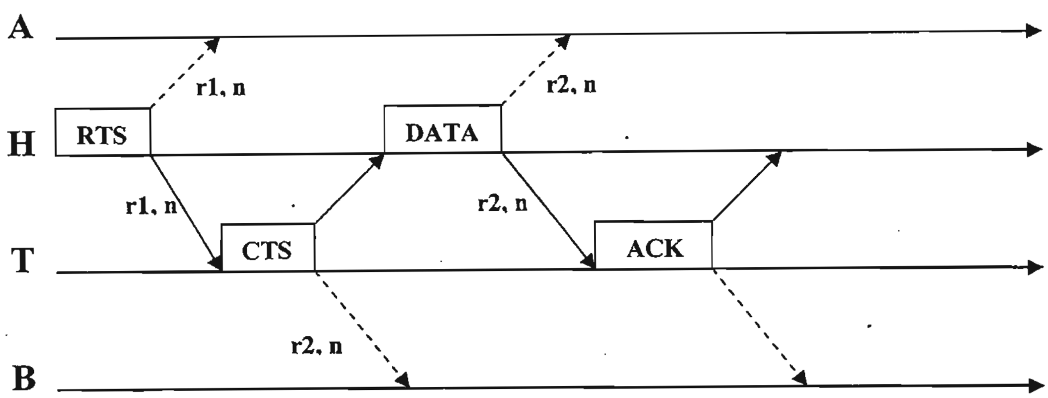


Figure 2.4: Illustration of RBAR protocol

Dynamic Channel Assignment (DCA)

DCA protocol [Wu, 2000] is designed for a multi-channel network. Whereas the single channel protocol uses only one channel for information transmission between all nodes, the multi-channel network employs multiple channels for transmitting information. Depending on the technology used, the channels can be frequency bands or CDMA codes. A channel is normally assigned to several nodes in the network. In order to communicate with a node, the host node has to transmit its information using the channel that is assigned to the target node. Other nodes in the network cannot sense the transmission unless the same channel was assigned to them.

In DCA, the overall bandwidth is divided into one control channel and N data channels. The control channel is used to resolve contention and to grant permission to access the data channels. Each node in the network maintains two lists: the channel usage list monitors information about the neighbors and their channel usage and the free channel list is computed by the nodes formed from the channel usage list.

When a host node wants to transmit to a target node in the neighborhood, it sends the RTS packet together with its free channel list to the target node through the signaling channel. The target node then matches the incoming list with its own channel usage list to identify a data channel to be used. If there is a data channel available, the target node then replies to the host node with a CTS packet. The intention of the CTS packet is to inform the nodes surrounding the target node not to use the allocated data channel. On receiving the CTS packet from the target node, the host node transmits a reservation packet to inhibit other neighbors from using the same channel. The host node then begins to transmit its data packet to the target node. The protocol has a set of complex rules and calculations that are used after a node receives the free channel list that comes with the RTS packet, designed to schedule the transmission and avoid collisions.

Collision Avoidance Transmission Scheduling (CATS)

CATS [Tang, 2000] is also a multi-channel protocol designed for ad-hoc networks and an extension of CATA [Tang, 1999]. In this protocol, the bandwidth is divided into N+2 channels: one signaling channel (SCH), one broadcast data channel (BCH), and N data channels (DCH). The channels are then further divided into time slots, each consisting of five mini-slots (MS1 to MS5) and one data slot. CATS uses seven types of signaling messages called beacons in the SCH; the size of a beacon is the same as that of a mini-slot.

If the node is inactive (not engaged in reserving a channel, or sending or receiving data over a channel), it has to listen to the SCH. A node that needs to transmit or receive data will transmit a link reservation beacon (LRB) in the MS1 over the SCH to inform the surrounding nodes that it is busy. In the scenarios like uni-cast and broadcast, every receiver node has to transmit a LRB in MS2. This is to inform its neighbors not to establish a multicast or uni-cast link over it. MS3 to MS6 are used for data transmission in the BCH and DCH.

In a reservation for uni-cast, the host node listens to MS1 in the SCH to confirm that the slot is available. The host node then listens to the MS2 in the allocated DCH. If the target node is silent, the host transmits a request uni-cast beacon (RUB) over the SCH during MS3. If the target node successfully receives the RUB, and DCH is silent during MS4, the target node then sends a concur-with-uni-cast beacon (CUB) in MS5. The host node initiates its data transmission once it received the CUB in MS5.

A similar procedure is used in a reservation for multicast, the difference being that a request multicast beacon (RMB) is used instead of a RUB during MS3. If the host node observes that the SCH stays clear during MS4 and MS5, it assumes that the reservation is successful and transmits its data in MS6 over the DCH.

For a broadcast reservation, a request broadcast beacon (RBB) is sent from the host node during MS3, after observing a silent MS1. If a node receives the RBB or observes a silent period during MS3, it remains silent during MS4. Otherwise, the node transmits a stop broadcast beacon (SBB). The host node assumes that the reservation is successful and starts data transmission if it does not detect SBB during MS4.

2.4 Wireless MAC protocols for centralized networks

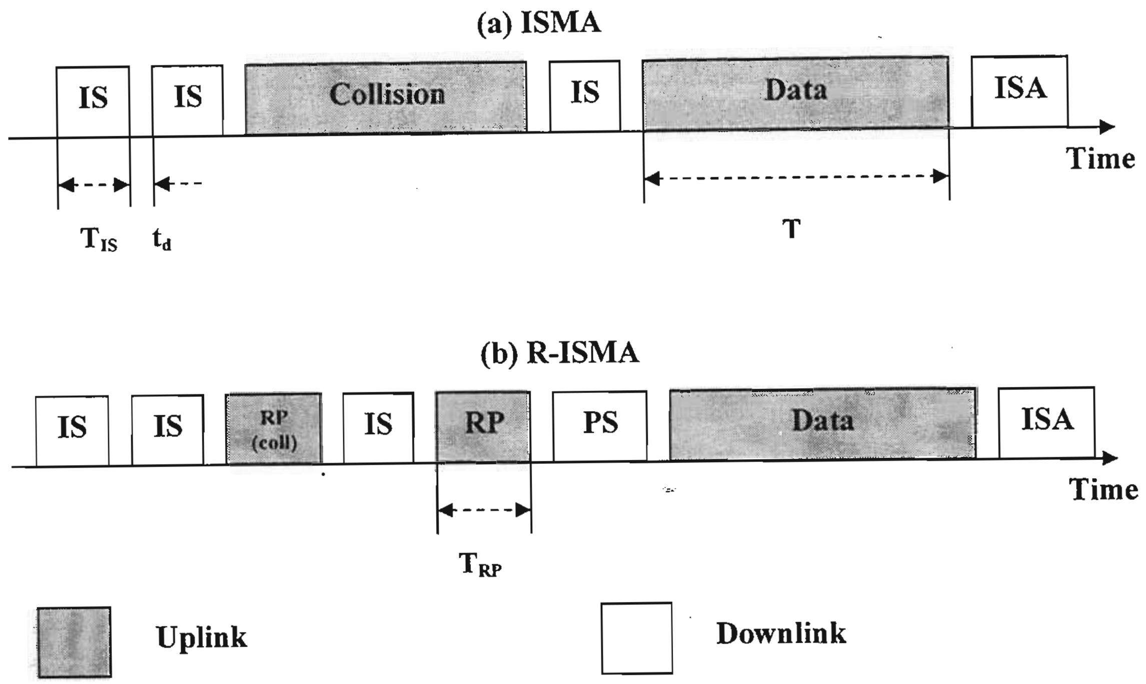
Centralized MAC protocols are designed for networks with base station topology. The base station (BS) has absolute control over which mobile node can access the medium and because the BS is situated in the middle of the network, it is assumed that all nodes can exchange information with the BS. This type of topology allows a highly optimized medium access control as both the hidden and exposed node problems do not exist. These types of protocols can be divided into three categories: random access, guaranteed access, and hybrid access.

2.4.1 Centralized random access protocols

For these types of random access protocols, no regulations or guarantees are provided in the network, so mobile nodes can access the channel freely and randomly. The major disadvantage of the random access protocols is packet collision, where a collision occurs when two or more mobile nodes choose the same slot to transmit their data.

Idle Sense Multiple Access (ISMA)

ISMA [Wu, 1993] is a contention based access protocol in which the carrier sensing and collision detection are performed by the BS. The operation of this protocol is illustrated in Figure 2.5(a). The BS broadcasts an idle signal (IS) when the channel is idle. There is a propagation delay (t_d) between each IS which allows the BS to receive responses from the mobile nodes that intend to transmit in the following period. The mobile node can transmit its data packet to the BS when it has sensed that the channel is idle. If the data packet from the mobile node has been transmitted and received by the BS without any error or collision, the BS then transmits an idle signal with an acknowledgement (ISA). The ISA is used to confirm the successful packet transmission and also to notify other nodes that the channel is available again. If a collision has occurred, the BS can not decode the transmission so it will broadcast an IS again. The transmitting nodes will know their transmissions have failed when they receive an IS instead of an ISA, and they then back off for a random period of time before sensing the channel again.



T_{IS}:	**IS Transmission time**	**T_d:**	**Propagation and processing delay**
T_{RP}:	**RP Transmission time**	**T:**	**Packet transmission time**
IS:	**Idle signal**	**RP:**	**Reservation packet**
ISA:	**Idle signal with acknowledgment**	**PS:**	**Polling signal**

Figure 2.5: Operation of ISMA and r-ISMA protocols

As illustrated in Figure 2.5(a), when a collision occurs in ISMA, the entire data packet is lost, resulting in poor efficiency. An improved version of ISMA, called reservation ISMA (R-ISMA) [Wu, 1996], was proposed in which collisions are avoided by using reservation packets and polling techniques. In R-ISMA (Figure 2.3(b)), the mobile nodes transmit a short reservation packet (RP) instead of a data packet after sensing an idle channel. If a collision occurs while transmitting the RP, the RP is then lost and the mobile node will then retransmit the RP after a random interval. This is illustrated in Figure 2.5, where two reservation attempts are made in part (b), with the same time wasted due to collision as in part (a). If the RP transmission is successfully received by the BS, the BS sends a polling signal (PS) to the mobile node. By receiving the PS from BS, the mobile node is allowed to transmit a data packet. During the data packet transmission, the BS will not transmit an IS, therefore no mobile nodes will attempt a transmission. If the data packet fails to reach the BS, the BS polls the mobile node demanding a retransmission; otherwise the BS broadcasts an ISA notifying the mobile node that it has successfully received the data packet.

Randomly Addressed Polling (RAP)

RAP [Chen, 1993] is a protocol that incorporates contention based random access and polling algorithms into its protocol, with CDMA receiver being used in the contention phase, situated in the beginning of the frame. When mobile nodes need to transmit; each of them randomly chooses a CDMA code and uses the code as a request in the contention phase. Since all the nodes transmit their chosen codes simultaneously, the BS has to decode multiple transmissions and receive all the codes that were transmitted.

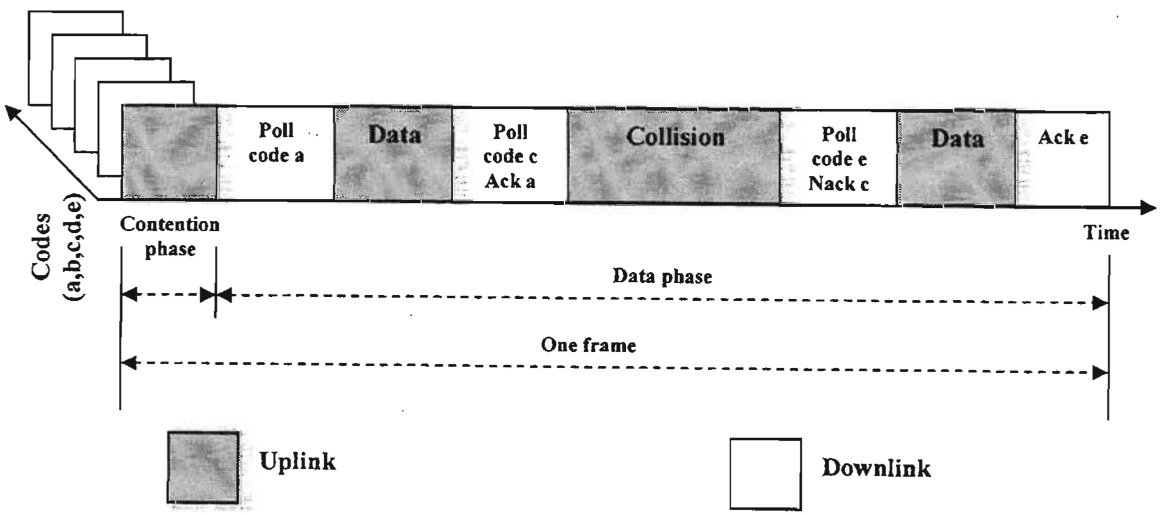


Figure 2.6: Illustration of the RAP protocol

The protocol is illustrated in Figure 2.6. In the contention phase, as mentioned earlier, active mobile nodes chooses a code and transmit it in the same time slot. When the BS receives the codes, it first identifies the codes then sends a poll to the corresponding node. When a mobile node senses the poll of the code that it picked, it transmits its data packet. If the data is successfully received by the BS, the BS sends an ACK to the node. If two or more mobile nodes have used the same code in the contention phase, then a collision occurs and a NACK is sent. The nodes that received NACK retransmit their request in the next contention period. When all the received codes have been polled, the BS starts a new contention phase.

Three codes (**a**, **c**, **e**) are chosen for demonstrating purposes in the example shown in Figure 2.6. Code **a** was chosen by a single node therefore no collision occurred while transmitting the data packet and the BS sends back the ACK. Code **c** was chosen by two or more nodes, which results in a collision, therefore the entire transmission is lost and the BS sends back the NACK packet. Lastly, code **e** was chosen by a single node which results in a successful data transfer.

Resource Auction Multiple Access (RAMA)

RAMA [Amitay, 1992, 1993] is a random access protocol that uses a deterministic access algorithm to achieve resource assignment (Figure 2.7). The protocol has two phases: contention phase and data phase. Each mobile node is assigned an unique binary b-bit ID string. In the contention phase, the node that needs to transmit sends its ID string symbol-by-symbol to the BS. The BS then broadcasts the symbol it has heard to all the nodes. If the broadcast from the BS matches the ID string of the node, the node wins the contest. Only the winning node is allowed to transmit its data in the data phase, all the other nodes retransmit their ID in the next contention phase.

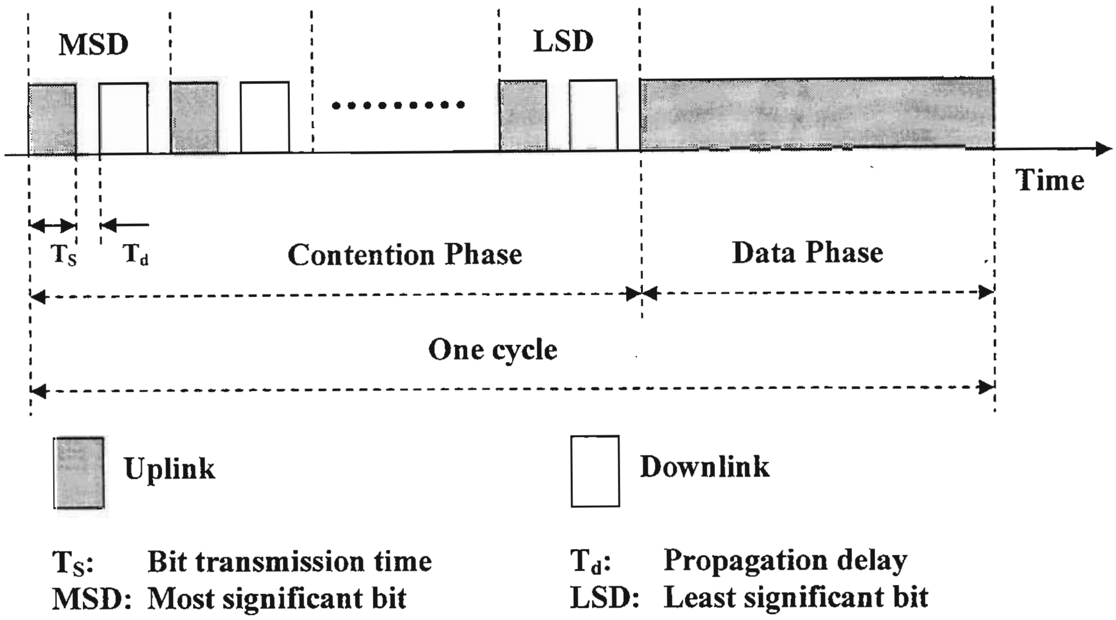


Figure 2.7: Description of RAMA protocol

Consider an example where node A with ID 100 and node B with ID 111 are contending for access to the channel. Both of the nodes transmit symbol '1' and BS acknowledges '1' in the first round. However, in the second round, node A transmits symbol '0' and B transmits symbol '1', the BS performs an OR operation on the symbol transmitted and thereby acknowledges '1'. As a result, node A backs off from contention and the process continues till the complete ID string is sent. The node with highest ID always wins the contention and obtains the permission to transmit in the data phase.

As the node with the highest ID always wins the contention, it may consequently starve other nodes in the network of service. Therefore the RAMA is considered as an unfair protocol, but because of this unfair prioritization scheme, the contest always produces a winner, even if collisions occur. Bandwidth is therefore not wasted due to collision. The issue of fairness is later addressed in the Fair-RAMA (F-RAMA) [Pinheiro, 1996] protocol. In F-RAMA, instead of choosing the highest symbol, the BS selects one of the received symbols randomly before broadcasting the symbol to all the nodes.

2.4.2 Centralized guaranteed access protocols

Polling algorithms are used when designing guaranteed access types of protocols. Polling is defined as a control handshake that is similar to the handshake used in the collision avoidance ad hoc network. It is initiated by the BS using a small packet that carries a message to a specific mobile node. The mobile node responds to the polling packet according to the protocol it adheres to. The BS polls each mobile node in the network for data, one after the other, in a round robin fashion. For these types of protocols, there exists no collision hence they can achieve high utilization when many mobile nodes are accessing the channel. However, bandwidth can be wasted when the polled mobile node has nothing to transmit.

Zhang's proposal

Zhang's protocol [Zhang, 1991] (Figure 2.8) consists of two polling phases. In the first polling phase, the BS polls all the nodes in a round-robin fashion for transmission request. If the polled node has data to transmit, it responds to the poll with the request packet or if its queue is empty, it transmits a "Keep Alive" message to the BS. The "Keep Alive" packet is used by the node to inform the BS that it is still in the network. After polling all the nodes in the network, the BS starts the second polling phase, in which it starts to poll the nodes that had replied with the request packet for data. This protocol is primitive but it guarantees that all mobile nodes are polled and all transmissions are free of collisions.

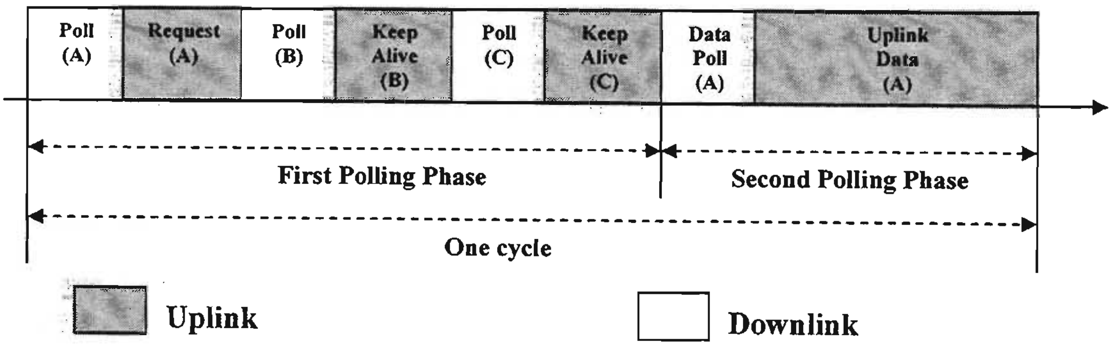


Figure 2.8: Zhang's proposal

Disposable Token MAC Protocol (DTMP)

The DTMP [Haines, 1993] alters Zhang's cyclic format from a poll-request-poll-data, to just a poll-data cycle. The operation of the protocol is shown in Figure 2.9. In DTMP, the data transmissions for both the uplink and the downlink are followed by a single poll. There are two types of poll in the protocol: the normal poll and the data poll. If the BS has no data for the polled node, it sends a normal poll and if the polled node has no data to transmit, it stays silent. After sending the normal poll, the BS waits for the data from polled node for a short period. If no data is transmitted from the polled node in that period, the BS polls the next node. However, if the polled node has data to transmit, it transmits its data immediately after being polled, and the BS replies with an acknowledgement after the data packet has been successfully received.

A data poll is used when the BS has data for the polled node. If the polled node does not have any data for the BS, it transmits a "no data" message to the BS upon being polled. If the polled node has data to send, it sends its data when it is polled. After receiving the "no data" message or the uplink data, the BS transmits its data to the polled node. If the downlink data transmission is successful, the polled node sends an acknowledgement.

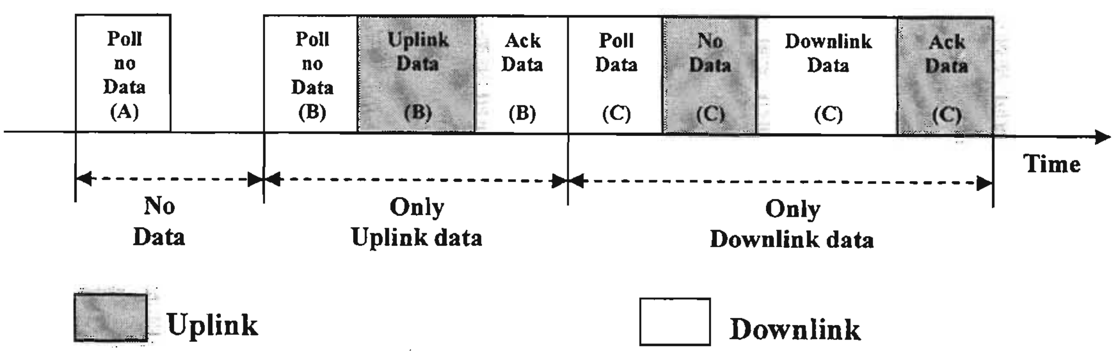


Figure 2.9: Disposable token MAC protocol

Acampora's proposal

Acampora's protocol [Acampora, 1997] operates in three phases: polling phase, request phase, and data phase. Figure 2.10 shows the frame layout of the protocol. Each mobile node in the network is assigned a codeword that is unique to that node. In the polling phase, the BS polls the mobile node with its codeword. If the polled node has data to transmit, it echoes the codeword back. It remains silent if it has no data to transmit. The codeword technique is used to minimize bandwidth wastage when the polled nodes have no data to transmit.

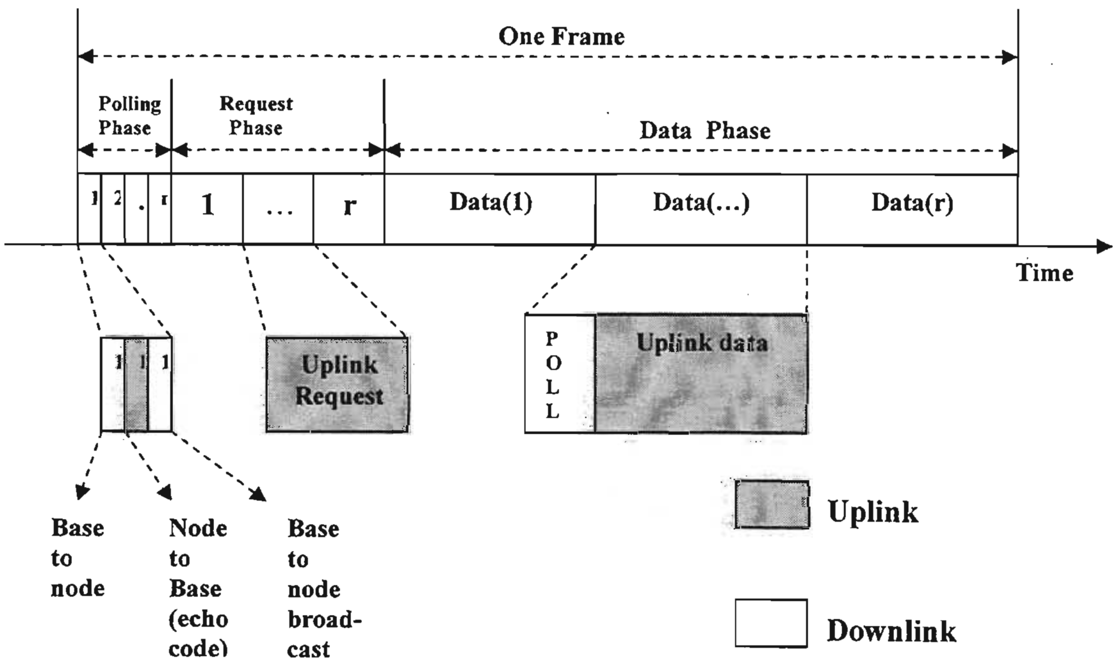


Figure 2.10: Acampora's proposal

Upon receiving the echo, the BS then broadcasts the codeword back to the network, and the nodes use this information to calculate their slots in the request phase. Hence, the nodes know how many slots they have to wait before transmitting their requests in the request phase. In the data phase, the BS polls the nodes that had transmitted in the request phase. The polled node transmits its data packet when it is polled, and the BS transmits the downlink data to the nodes when all the requested nodes have been polled.

2.4.3 Centralized hybrid access with random reservation protocols

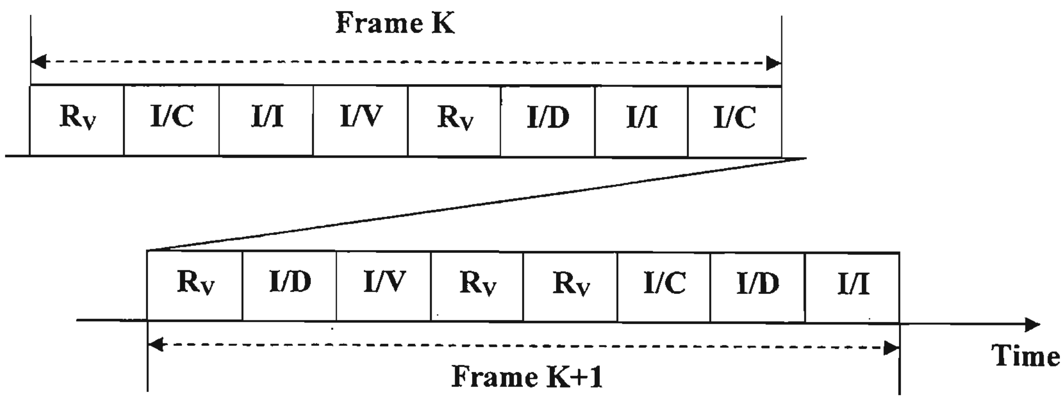
Hybrid access protocols combine the best features of both the random access and guaranteed access protocols. Depending on the scheduling and reservation policies imposed by the BS, hybrid access protocols can be further classified into two classes: random reservation and demand assignment protocols. Random reservation protocols attempt to combine the flexibility of random access with guarantees of polling access. Not many of the protocols can be categorized into this class, as more complex random reservation protocols normally contain a demand assignment access feature and hence are categorized as demand assignment protocols. Random reservation protocols consist of two phases: random access and reservation. All mobile nodes access the channel in a free and random fashion and if access is successful, the nodes then reserve the same slot in the following frames.

Packet Reservation Multiple Access (PRMA)

PRMA [Goodman, 1989] protocol emphasizes reserving bandwidth for voice traffic where it implements a frequency division duplex technique. For PRMA, the uplink stream is divided into frames, each frame containing a number of data slots. The P-persistence slotted-Aloha access scheme is used by the nodes to grant access to the data. If a slot is available, the mobile node has a probability of p of transmitting in the slot.

As PRMA is designed to multiplex speech and data, it defined two types of traffic: voice and data traffic. It also defined that voice traffic has a higher priority than data traffic, with the priority setting achieved by assigning different access probabilities for both voice and data traffic. When a mobile node has voice packets in its buffer, it transmits the first packet in the unreserved slot using the slotted-Aloha access scheme. If the transmission is successful, the same slot is then reserved for the voice traffic in the succeeding frames. Slot assignment is achieved using downlink broadcast, thereby the voice packets are transmitted without any possibilities of collision.

The reserved slots are only released when the talk spurt has ended. With data traffic, data packets are transmitted in a non-reserved fashion, as they have lower priority than the voice traffic. An example is shown in Figure 2.11. In this figure, a successful voice transmission occurs in slot 4 of frame K, and this slot is reserved for this voice traffic in the next K+1 frame. It can be seen that a data node contends successfully in slot 6 of frame K, but no reservation is made in frame K+1.



Notation: a/b: (slot specify by BS)/ (slot used by nodes)

R_V:	**Slot reserved for voice**		
I:	**Idle slot**	**D:**	**Data transmission**
C:	**Collision**	**V:**	**Voice transmission**

Figure 2.11: Upstream frame structure of PRMA and its operation

Integrated Packet Reservation Multiple Access (IPRMA) [Goodman, 1993] is built based on PRMA protocol, with the additional feature of providing reservation for data traffic. IPRMA implements the same rules used by PRMA for voice traffic transmission and also proposes a limitation on the number of slots that can be reserved. Voice traffic reserves slots vertically, that is, the same slots of the following frames are reserved. Data traffic, however, reserved slots horizontally in the remaining empty slots, and the data traffic can reserve fewer slots than those reserved by voice traffic.

2.4.4 Centralized hybrid access with demand assignment protocols

Demand assignment protocols incorporate both random access and guaranteed access protocols, and allocate bandwidth to nodes according to their quality of service (QoS) requirement. Most of these protocols are designed for networks such as wireless ATM networks and wireless voice communication networks, where they required guaranteed QoS for their multimedia traffic.

There are three phases to demand assignment protocols: request, scheduling, and data transmission. In the request phase, a random access scheme is normally used by the nodes to send their QoS requirements to the BS. Once the requests have been gathered, the BS deploys a scheduling protocol to assign data slots to the mobile nodes that have requested it. The notification of the assignment of the uplink data slots is broadcasted through the downlink stream, and the data for the mobile nodes from the BS is usually transmitted after notification. The last phase of the protocol is the data transmission phase, where nodes transmit their data without any collision in the data slots assigned by the BS.

In demand assignment protocols, the BS is required to perform a large amount of processing. It must define the structure of both the uplink and the downlink streams, usually in a frame structure, and also performs the scheduling of data slots, acknowledgement of requests, and time synchronization.

Distributed-Queuing Request Update Multiple Access (DQRUMA)

DQRUMA [Karol, 1995] implements FDD for the uplink and downlink transmissions. Protocol details are illustrated in Figure 2.12. The frames used in the protocol are very small and fixed in size. A standard uplink frame consists of a request mini-slot (reservation phase), a piggybacking field, and a data slot (data transmission phase). A

DQRUMA frame can only carry one data packet. The request mini-slot can be accessed using the slotted-Aloha access protocol. The piggybacking field is for the mobile node that is assigned to the data slot to request more data slots if it has more data packets to transmit.

A standard downlink frame consists of an acknowledgement field, a transmit-permission field, and a downlink data slot. The acknowledgement field provides information on the request mini-slot of the uplink frame. The transmit-permission field contains the ID of the mobile node that will be transmitting data in the data slot of the next uplink frame. The downlink data slot simply carries the downlink data traffic from the BS to a particular mobile node.

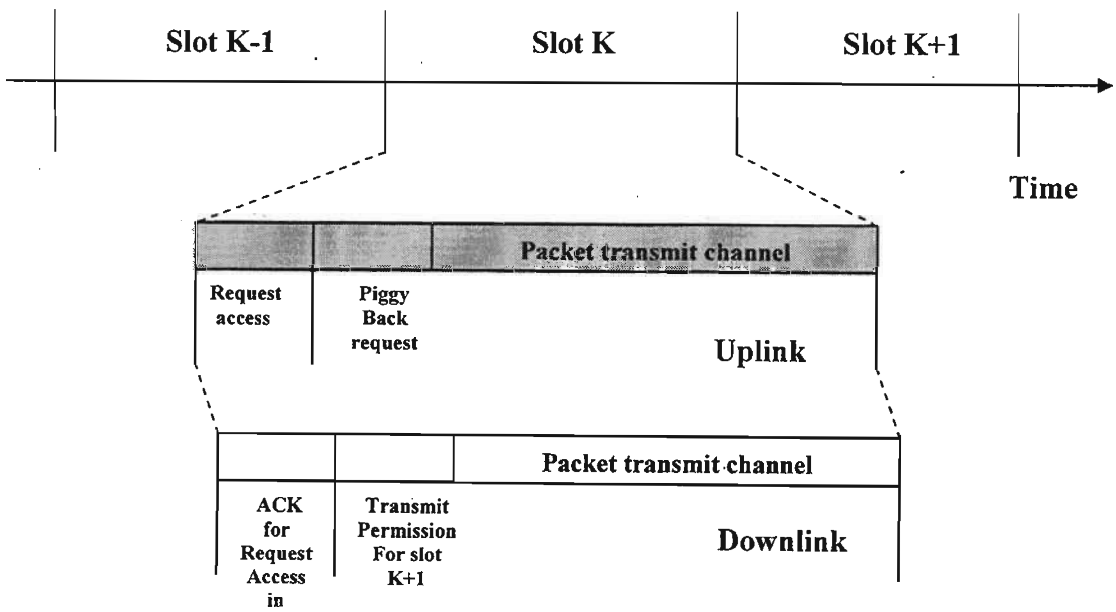


Figure 2.12: Details of the DQRUMA protocol

When a mobile node has data to transmit, it first transmits a request packet through a request mini-slot in the uplink frame using random access protocol. The BS receives the request and acknowledges it by broadcasting the ID of the node in the transmit-permission field of the downlink frame in the immediate downlink slot. The node knows the result of the contention request in the current slot before the beginning of the next slot (Figure 2.10). If a collision has occurred in the uplink request mini-slot, the node retransmits the request packet according to random access protocol. After hearing its own ID in the transmit-permission of the downlink frame, the mobile node knows the data slot of the next uplink belongs to it. The mobile node then transmits its data packet in the next uplink frame. If the node has more than one data packet to transmit, it sends a message in the piggybacking field to request more data slots.

Mobile Access Scheme based on Contention and Reservation (MASCARA)

MASCARA [Mikkonen, 1998] is the MAC protocol designed for the MAGIC Wireless ATM Network Demonstrator project. MASCARA is a TDD based MAC protocol using variable-length time frames. A frame consists of three phases: broadcast, data transmission, and reservation. This is shown in Figure 2.13. The broadcast phase is in the downlink direction and is used to inform all mobile nodes of the structure of the current time frame, the scheduled uplink transmissions, and to acknowledge the requests from the previous frame. The data transmission phase consists of two phases: a downlink data phase and an uplink data phase. In the downlink data phase, the BS transmits the downlink data to the mobile nodes. In the uplink data phase, the mobile nodes transmit packets in the order defined by the BS in the reservation phase.

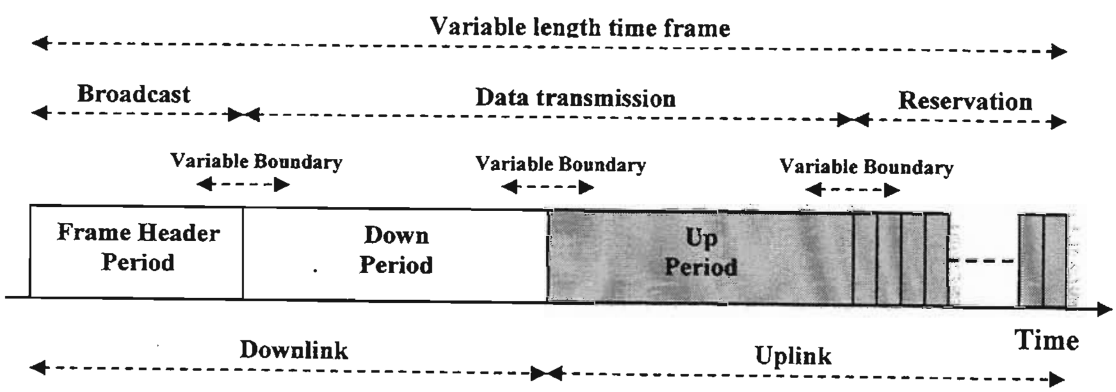


Figure 2.13: Frame structure of the MASCARA protocol

The reservation phase consists of many request mini-slots. It is used by the mobile nodes to send their requests to the BS. All packets that transmit through this phase use the slotted-Aloha access protocol. After the request reception, the BS makes uplink data slot assignments based on a leaky bucket token scheme called Prioritized Regulated Allocation Delay-oriented Scheduling (PRADOS) [Colombo, 1999].

Dynamic Packet Reservation Multiple Access (D-PRMA)

D-PRMA [Alasti, 1999] (Figure 2.14) is the modified version of the PRMA (a random reservation protocol). It uses request mini-slots for reserving data slots for voice traffic. The uplink stream and the downlink stream are divided into frames, however, the author did not specify whether the protocol is a TDD based protocol or a FDD based protocol. The uplink frame consists of three phases: a voice data transmission phase, a data phase, and a voice reservation phase.

The voice data transmission phase consists of data slots that are specifically assigned for voice traffic. If a mobile node needs to transmit a voice packet, it has to send a request packet through the voice reservation phase. The voice reservation phase is the equivalent of the reservation phase in other demand assignment protocols and consists of mini-slots. The BS uses a scheduling algorithm to assign voice data slots to the mobile node, with this procedure dedicated to voice traffic and constant bit rate traffic only. In the data phase, other non-voice types of traffic send their data packets. A contention scheme is implemented for accessing the data phase and the mobile nodes use the slotted-Aloha access protocol to send their data packets in this phase, using no reservation.

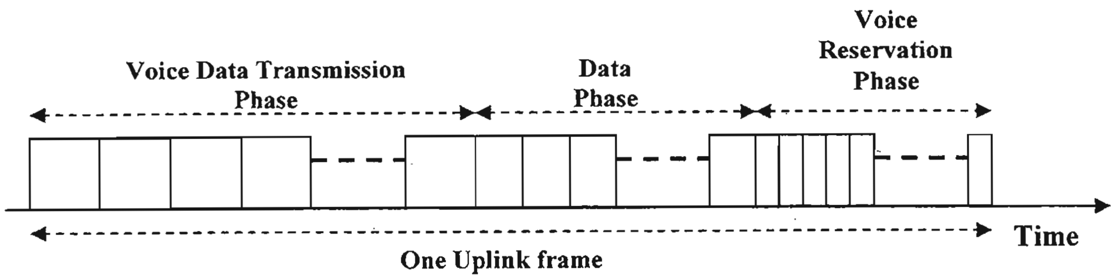


Figure 2.14: Uplink frame structure for D-PRMA protocol

Time Division Multiple Access with Dynamic Reservation (TDMA/DR)

TDMA/DR protocol [Wong, 2000] (Figure 2.15) is built based on TDMA frames. The uplink frame consists of two phases: the reservation phase and the data transmission phase. There are three classes of traffic generated by the mobile nodes: class 1- real-time VBR and time-critical data traffic, class 2- CBR voice with burst switching (on/off), and class 3- non time-critical data.

The reservation phase is divided into two parts. The first phase is called the available request channel and uses slotted-Aloha access protocol to send requests randomly by all classes to the BS. The second phase is called the used request channel; this channel is reserved for class 1 traffic that has successfully transmitted their requests through the available request channel. The purpose of the used request mini-slots is to decrease the delay of real-time VBR traffic by allowing class 1 traffic to send their requests without collisions.

Voice traffic is not considered as real-time VBR traffic in this protocol but as short CBR traffic. Voice traffic (class 2) must request data slots through the available request channel every time a talk spurt occurs. A slot in the data period is dedicated to the voice traffic until the talk spurt ends. When class 1 and class 2 traffic have collisions in the request channel, they retransmit their request immediately in the following frame. Class 3 traffic backs off when a similar situation arises. The data transmission phase is divided into four sections: CBR, CBR with burst switching, real-time VBR, and non-time-critical data. Each section consists of a different number of data slots which are assigned by the BS. The boundaries between each section are variable depending on the traffic load.

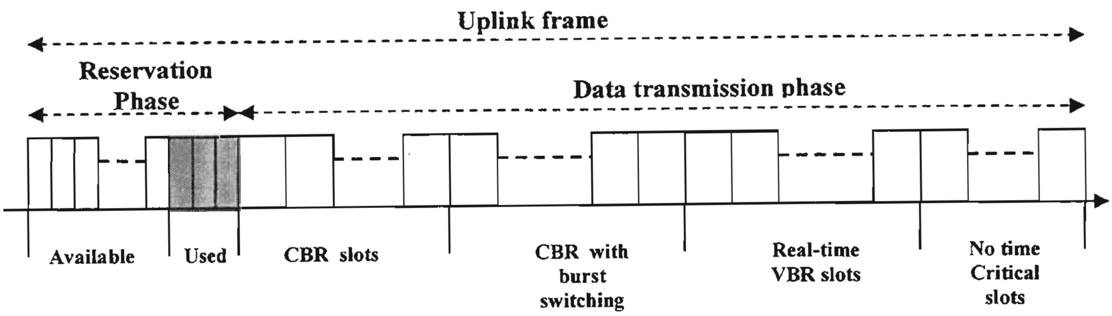


Figure 2.15: Uplink frame structure for TDMA/DR protocol

Collision Resolution and Dynamic Allocation (CRDA)

Based on PRMA protocol, CRDA [Lenzini, 2001] is the successor of the PRMA, but unlike PRMA, it is not a random reservation protocol. CRDA uses a request channel with a non-data carrying request packet to demand a resource. In CRDA, the uplink and downlink are FDD, and both are divided into frames. The frame structure of the CRDA is illustrated in Figure 2.16. Each frame contains N number of slots. There are two types of slots in the uplink frame: request (R) and data slots (D), and they are used in the reservation and data transmission phase respectively. The request slots are designed to have the same size as the data slots.

The downlink frame consists of acknowledgement (A) slots and data (D) slots. Both of the slots have the same duration. Furthermore, the A-slot is shifted with respect to the R-slot by a fixed number of slots which is termed the R/A-slot shift. This allows the requests from the uplink to be acknowledged immediately when the uplink reservation phase is completed. The downlink data slots carry not only the downlink data, but also information on the uplink data slot assignment.

When the mobile node has data to transmit, it first sends a request packet in the reservation phase. If the transmission is successful, it listens to the downlink data slots for the uplink data slot assignment. The result of the transmission is reported in the acknowledgement slots of the next downlink frame. If the node has more than one data packets to transmit, it may obtain the slots by using the piggybacking bit in the uplink data slot.

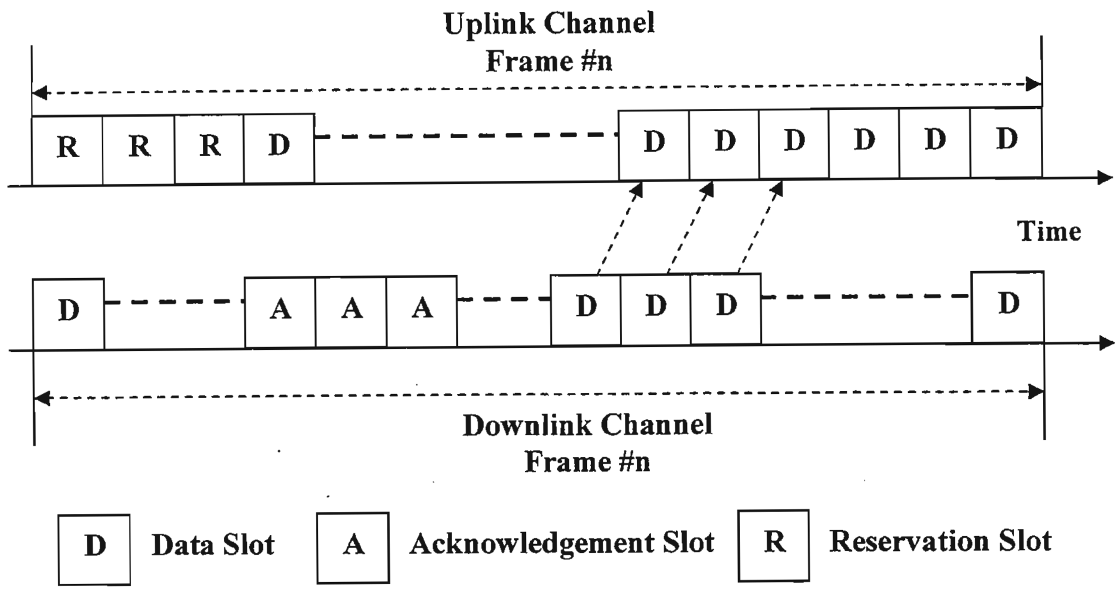


Figure 2.16: Frame structure for CRDA protocol

2.5 Wireless MAC protocols for hybrid networks

Hybrid MAC protocols are designed for networks that are implemented using ad hoc centralized topology, combining both ad hoc and centralized topology. A network is generally defined as an ad hoc network when it is formed without any central administration, resulting in mobile nodes that use a wireless interface to send packet data. However, in an ad hoc centralized network, there is a centralized administrator (e.g. a mobile BS) in the network and so the communication is centralized. The following are some examples of ad hoc centralized MAC protocols.

Bluetooth

Bluetooth [Haartsen, 2000] is a protocol that intends to replace cables connecting household appliances. It is ad hoc in nature and it is based on FH-CDMA. Bluetooth uses the 2.45GHz ISM band (Industrial, Scientific, and Medical band) and has 79 hop carrier channels and full duplex communication is achieved by applying TDD. For a short distance network with a diameter of ten meters, a Bluetooth piconet network can be formed and the multiple piconets with overlapping coverage areas form a scatternet. A Bluetooth piconet consists of eight or fewer wireless nodes. By definition, a node that initiates the connection will be represented as the master node, and the rest as slave nodes.

The master and slave relationship exists only when the piconet exists. All Bluetooth nodes have the same physical capability and can become master nodes or slave nodes. Once the master node is assigned in a piconet, it acts as the BS in the centralized topology and the slave node acts as the mobile node. The topology of the piconet is identical to that of centralized topology with the exception that the BS (master) is mobile instead of stationary. Similar to the centralized topology, slaves can only communicate with the master and not with other slaves. To access a channel, Bluetooth implements a polling based guaranteed access MAC protocol.

Figure 2.17 illustrates the operation of the MAC protocol used in Bluetooth. The master controls the access to the channel and it polls each slave in a round robin fashion. As shown in Figure 2.17, the master polls the slave A for data with a downlink poll packet. Slave A responds to the poll by immediately transmitting its data packet. The master then begins a downlink transmission. This downlink transmission is assumed to contain a packet and is destined for slave E. After successfully received the data, slave E sends an acknowledgement packet to the master. The master then polls the next slave on its list. Since the polled slave B does not have a data packet to send, it responds to the poll with a nothing-to-send packet. Like the BS, the master controls the entire traffic flow. The number of packets that can be transmitted by the slaves in each poll is controlled by the scheduling algorithm of the master.

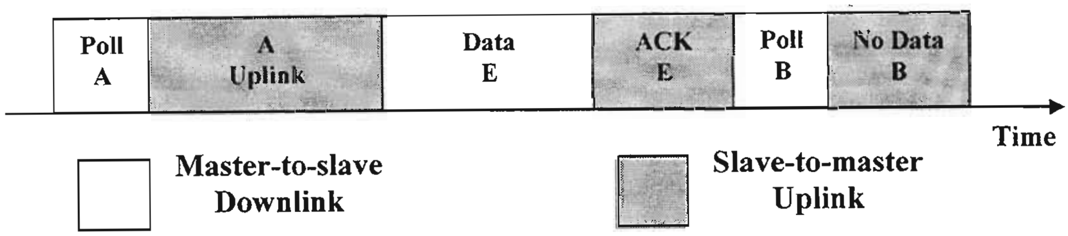


Figure 2.17: Illustration of MAC protocol used in Bluetooth

Centralized intelligent channel assigned multiple access (C-ICAMA)

C-ICAMA [Lihui, 2000] is a packet switching multiple access protocol for centralized ad hoc wireless networks. Within a single area, the protocol perceives the UAV (unmanned aerial vehicle) as the mobile base station. Each UAV provides multiple channels for communications. As illustrated in Figure 2.18, each channel is divided into time slots and the slots are further divided into frames with a fixed N_f slots in each frame. Each frame consists of three subframes: a downlink data subframe with N_d slots, an uplink data subframe with N_u slots, and a reservation subframe with N_r slots. The length of each subframe is adjustable, based on the traffic condition. Each slot in the reservation subframe is subdivided into a further N_m mini-slots. The mini-slots are used by the mobile nodes to store the reservation requests.

If the mobile node wants to transmit a packet or multiple packets, it first transmits a reservation packet on the contention basis based on the slotted ALOHA protocol. The reservation packet contains the number of packets that the node wishes to transmit. The UAV acknowledges the contention by sending back the success or collision packet. If the UAV has received the reservation packet successfully, it will allocate the slots that are needed by the node into the uplink data subframe. For a downlink transmission, the UAV first broadcasts the FRAME-START slot. The FRAME-START slot consists of three fields; the first field contains the number of downlink slots that are assigned in the downlink data subframe. The second field contains the list of nodes who had successfully contended for accessing the channel. The third field consists the number of slots in the reservation subframe. Once the FRAME-START packet is sent, the UAV then begins its downlink transmission. For the uplink transmission, since the mobile nodes know the order of transmission from FRAME-START packet, they transmit their packets accordingly.

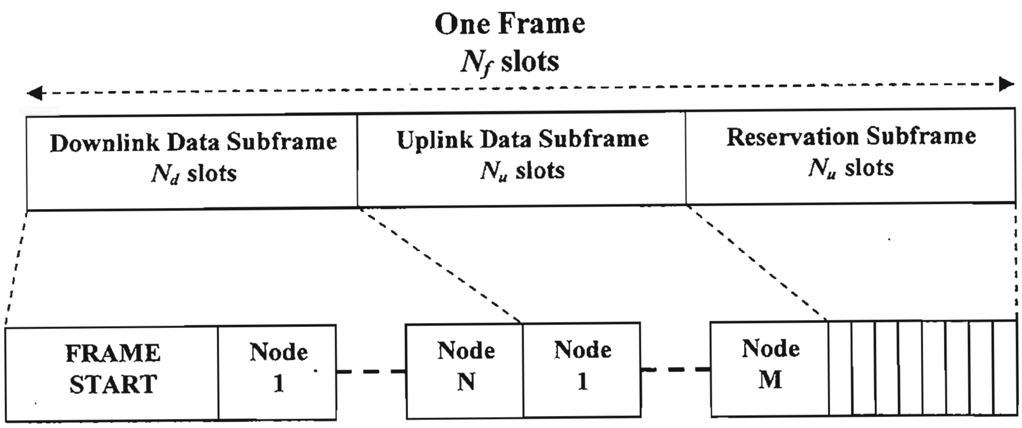


Figure 2.18: Frame structure for C-ICAMA protocol.

2.6 Summary

This chapter reviews existing literature on wireless medium access control protocols. Firstly, wireless MAC protocols are classified into different classes according to their topology and the approach they use. MAC protocols proposed for different classes of wireless networks were then explored. It is observed that the MAC protocols designed for ad hoc networks are very different from those designed for centralized networks. Most of the MAC protocols designed for ad hoc networks use handshakes and busy tones to avoid collisions and the performance of these protocols is not as good as that of the MAC protocols designed for centralized networks. This is mainly due to the propagation delays generated by the handshakes whilst trying to organize collision free transmissions. In the next chapter, a new MAC protocol that implements the advantage of the centralized guaranteed access scheme and applies it in an ad hoc wireless environment is proposed.

CHAPTER 3

THE PROPOSED TOKEN PASSING MAC PROTOCOL

3.1 Introduction

It is known from [Tobagi, 1975] that with the CSMA based scheme, performance is degraded by the hidden terminal problem. In a wireless ad-hoc network there is no guarantee that some terminals may not be hidden from the other terminals. To alleviate or solve the problem, many protocols based on the RTS/CTS handshaking protocol have been proposed. However, it has been shown [Deng, 1998] that when the traffic load is heavy, a packet may still encounter collisions with probability as high as 60%, due to loss of RTS or CTS packets. CSMA type of protocols are also considered to be unfair schemes, as users don't have fair opportunities of transmission. The collision of data packets also degrades the network performance in terms of throughput and delay. Up to now, there exists no protocol that has totally eliminated the problems caused by hidden or exposed terminals. Detailed descriptions of these two phonenomen are discussed in Appendix A.

Guaranteed access types of protocols are known to outperform random access protocols, as the users have fair access to the channel, and they don't generate any packet collisions. Therefore, a protocol that incorporates the advantages of a guaranteed access scheme into a distributed type of wireless network is proposed here. The scheme is based on token passing, with the two most commonly implemented token passing schemes being: the IEEE 802.5 [IEEE, 1989] standard and FDDI [ANSI, 1987]. Both of these are based on token ring protocol, the only difference between the two being the physical implementation. In these schemes, the token is passed to the next node in the polling cycle according to a polling order table. For a wireless network, it is important to note that whenever the token is passed, the address of the next recipient must be explicitly given, since all the nodes will receive the transmission. The proposed protocol is also based on the token passing scheme, with some modifications. The detailed description of the scheme is discussed in section 3.2.

The remainder of this chapter is organized as follows. Section 3.2 describes the proposed MAC protocol in detail. The simulation model and the model parameter are described in section 3.3. Simulation results are in section 3.4 and section 3.5 consists of a summary for this chapter.

3.2 MAC Protocol Description

The protocol description starts with a discussion of the effects of using CDMA code channels within the protocol, and then it follows with a description of the node and token structure. The channel access scheme implemented in the network is then considered. Finally, the quality of service (QoS) guarantees that are incorporated into the protocol are discussed.

3.2.1 CDMA technique

CDMA technology makes use of a spread spectrum known as direct sequence spread spectrum (DS-SS) [Pursley, 1987]. In spread spectrum communication, a transmitter spreads a transmission in a wide frequency spectrum by using a spreading code, which is independent of the data packet being sent. A receiver uses the same code to de-spread the received signal and retrieve the data. The codes themselves are unique to each CDMA channel. CDMA based schemes have been shown to offer improved performance compared to TDMA based schemes; among these are greater channel capacity and graceful degradation.

The hidden or exposed node may cause a collision if it wants to transmit data to the node that is busy transmitting or receiving data, thereby disrupting both transmissions. Due to the fact that each CDMA code channel effectively identifies the other CDMA code channels as noise, the interference caused by collision is minimized. Using this advantage, CDMA code channels are implemented in the proposed protocol to alleviate the interference problem caused by the hidden and exposed terminal phenomenon.

3.2.2 Node structure

In the proposed protocol, each node in the network is assumed to be equipped with two full-duplex transceivers; one set is used specifically for token transmission and reception, and the other set is dedicated to use for data packet transmission and reception. Each node is also assumed to be equipped with a MUD (multiple user detector), in order to receive more than one transmission simultaneously. The nodes maintain an updated list of their neighbors in order to know where the token should be passed to. Every node in the network supports the QoS guarantees that are proposed by the protocol.

3.2.3 Token structure

The token consists of the source address, destination address, number of codes available (*NOC*), and network parameters as shown in Table 3.1.

Source ID	Destination ID	NOC	Network Parameters

Table 3.1: Token structure

- Source ID (8bits): the address of the node who passed the token (predecessor's address).
- Destination ID (8bits): the address of the node that is currently holding the token (successor's address).
- Number of codes available (*NOC*) (4bits): this parameter is used to limit number of simultaneous transmissions in the network. A detailed explanation of this parameter is given in section 3.2.4.

- Network parameters (20bits): this field is used to pass other network management information to other nodes.

3.2.4 Quality of service (QoS) incorporation

Quality of service is normally defined as a measure of the satisfaction experienced by a user using the service. In a wireless communication system, there are numerous methods to formulate the QoS objectives quantitatively. It has been decided to incorporate two quality of service guarantees: data rate and delay level QoS guarantees. The description of the services is given below.

3.2.4.1 Data rate QoS guarantee

Figure 3.1 shows the node model used when incorporating data rate QoS. In order to provide this service, a permit generation system is implemented. Each node is equipped with a permit buffer for storing the generated permits. The permit generation rate (α) is proportional to the data rate (λ),

$$\alpha_i = k \cdot \lambda_i \tag{3.1}$$

The subscript i denotes the node number and k is the proportional constant which is the same for all the nodes. As shown in (3.1), the parameter k is used to control the permit generation rate. The main purpose of the parameter k is to provide fairness in the network by ensuring that all nodes receive sufficient access to the network. This is achieved by constraining the node-transmit capacity, limiting the maximum number of packets that can be transmitted and by varying the parameter k. If the token has been captured by node i, then the packet buffer is emptied such that the number of packets removed is equal to the number of permits (α_i) in the permit buffer (the scheme is gated). The maximum number of packets that may receive services is dependent on the quantity of the permits in the permit buffer. If the permit buffer is empty when the token arrives, the transmission request is then denied. The objective of implementing this QoS guarantee is to provide fairness into the network by ensuring that all nodes receive sufficient access to the network. This is achieved by constraining the node-transmit capacity, limiting the maximum number of packets that can be transmitted when the token visits the node.

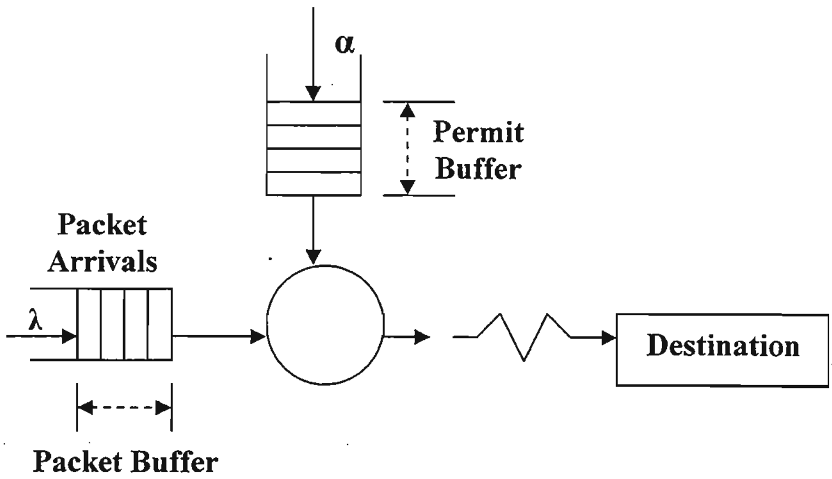


Figure 3.1: Node model with data rate QoS guarantee

3.2.4.2 Delay level QoS guarantees

The delay level QoS guarantee is implemented using a channel reservation technique as illustrated in Figure 3.2. With channel reservation, delay classes are formed in the network and each delay class, i, has associated with it a delay level parameter N_i where $0 \le N_i \le C$; C is the maximum number of codes used in the network. Each node in the network is assigned to a delay class.

The proposed MAC scheme incorporates both QoS guarantees. In order to satisfy both QoS guarantees, the node first has to satisfy the delay level QoS. For example, if a node with class i receives the token, it may only capture the token if the number of codes (NOC) in the token is greater or equal to the delay level parameter N_i ($N_i \le NOC$). Once the token is captured, the data rate QoS is applied to determine the number of packets that may receive service. The purpose of implementing delay level QoS is to assign priorities to different traffic classes. Traffic classes with higher priority have the advantage of attaining access to the data channel, thereby generating lower packet delay than low priority classes.

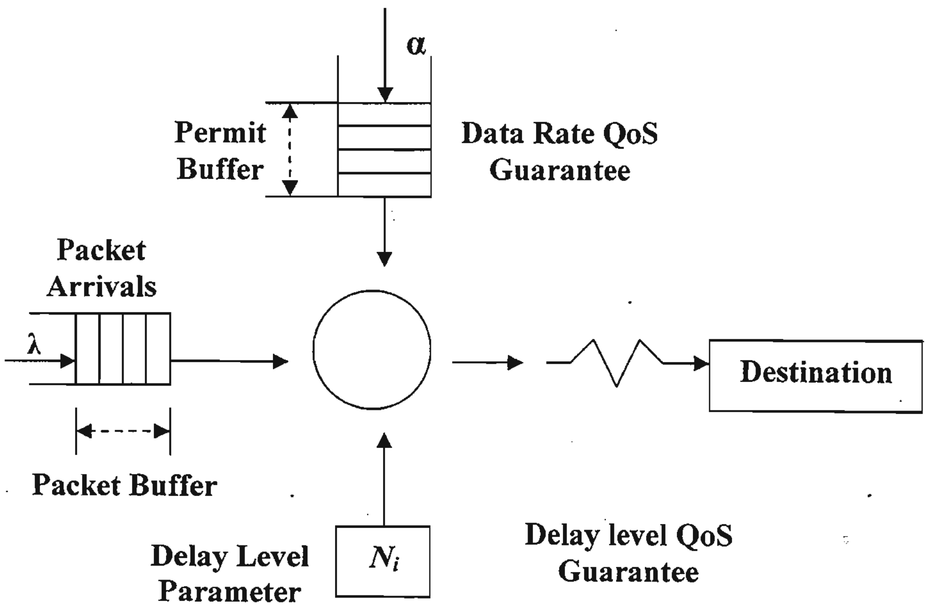


Figure 3.2: Node model for the MAC scheme with data rate and delay level QoS guarantees

3.2.5 Channel access control

Channel access control is implemented using a token passing technique. Once the token is generated in the network, it continuously circulates within the network, following a predetermined order. As discussed in section 3.2.4, the proposed scheme accommodates two different QoS guarantees. The access scheme is described as follows:

3.2.5.1 Access control with data rate and delay level QoS guarantees

A node may only transmit its data packets if both the data rate and delay level QoS are satisfied. As the token constantly circulates within the network, it uses the parameter *NOC* to control the amount of traffic flow in the network.

If the node has data packets that it wishes to transmit, it has to wait for the arrival of the token. When a node is visited by a token, it forwards the token to the successor node without capturing it under two circumstances. In the first scenario, the node forwards the token if it is busy transmitting data packets. In the second scenario, the node forwards the token if it has finished transmitting data packets and also has a code channel. In the later scenario, the node has to release the code to the token irrespective of whether or not it has packets that it wishes to transmit. This action is to avoid the node constantly withholding the code for its own transmission, thereby disrupting the fairness of the network. Once the code is released, the token increments its *NOC* value by one.

If the node is neither transmitting nor has a code channel when the token arrives, it may capture the token if the delay constraint of the delay level QoS requirement is met. If the node fails to satisfy this QoS guarantee, it will forward the token to its successor node according to the pre-defined order. Once the node captures the token, data rate QoS guarantee is applied to limit the number of data packets the node is allowed to transmit. If the node fails to satisfy the data rate QoS guarantee (either the permit or packet buffer is empty), it then forwards the token to its successor node. After satisfying the data rate QoS, the node attains the code channel by decrementing the *NOC* value by one, indicating that it has occupied a code channel. Using that specified code channel, up to σ packets will be transmitted, where σ is the number of permits inside the permit buffer at the time when the token is captured by the node. The node forwards the token to the successor node in the pre-defined order before it begins with its data transmission.

Once the intended sender node captures the code from the token, it decrements the *NOC* by one and scans through the code status table inside the token, in order to find the code that is available for transmission. The code status table is used to monitor the availability of the code channels assigned in the token. An example is shown in table 3.2, where in this case node 1 has captured code 1, and nodes 3 and 4 have captured code 2 and 3 respectively. Both node 2 and node 5 have not captured any codes.

Node 1	**Node 2**	**Node 3**	**Node 4**	**Node 5**
1	0	2	3	0

Table 3.2: Code status table

After selecting the available code channel, the sender node broadcasts the channel in which it is going to transmit its packets. The nodes in the network reply to the broadcast message by listening to that code channel to monitor if there are any data packets that are destined to them. For the transmission and reception of data packets, the sender node inserts the destination node's address into the data packet before transmission. Each data packet has a header that stores the address of the destination node. For the reception of the packet, once the sender node transmits the packet, the surrounding nodes in the network receive it and check the address that is stored in the header. If the address matches that of a surrounding node, the node knows it is the destination node and receives the packet. However, if the address does not match, the node ignores the packet and regards the transmission as noise.

3.2.5.2 Token lost and node out of network scenarios

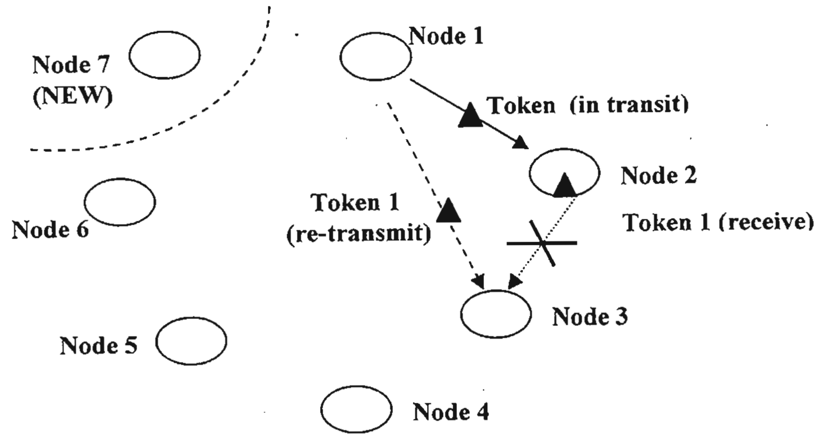


Figure 3.3: Graphical display of the described scenarios

Figure 3.3 displays the graphical illustration of the proposal on tackling the issues of a token being lost and nodes entering and leaving the networks. As the token starts circulating in the network from one node to another, a token detection scheme is created for monitoring the token's activity.

Considered the example shown in Figure 3.3. In this network, node 1 transmits the token to successor node 2. After forwarding the token to node 2, node 1 monitors the channel to observe whether node 2 has forwarded the token to node 3 or not. The monitoring is achieved by detecting token transmission activity. After a predefined time-out period, if node 1 has not detected token transmission activity, it will assume that the token is lost. For the "token lost" scenario, node 1 regenerates the token and forwards it to node 2 again.

However, if four attempts are made to retransmit the token to node 2, and node 1 still detects no token transmission, node 1 will assume that the "node leaves network" scenario had occurred. Node 1 will then update the node list in the token and regenerate the token, forwarding it to node 3 instead of sending it to node 2 again. This action is performed in order to avoid the generation of the looping effect. The looping effect is defined as the situation where the predecessor node repeatedly forwards the token to the successor node, which has already departed from the network. The proposed token detection scheme is considered as more efficient than using the acknowledgment (ACK) packet method. By sending an ACK packet to notify the successful token reception, more errors may be created, as ACK packets may be lost due to the wireless nature of the medium.

3.2.5.3 New nodes join the network scenario

As illustrated in Figure 3.3, if the new node 7 wishes to join the network, it has to wait for n token cycles. The value of n is a predefined value stored in each node once the network is initialized. A token cycle is denoted as the time it takes for a token to circulate through all the active nodes in the network. Each node in the network is equipped with a token-visit counter to record the number of times that the token has visited it. Once the value of n is reached, the node that is currently visited by the token becomes the new-cycle-initialisation node. The new-cycle-initialisation node then broadcasts a welcome message. Responding to the welcome message, all active nodes in the network send back their own address, and reset their token-visit counter to zero. The new-cycle-initialisation node then sorts all the node addresses in the node list, and forwards the token to the next successor node in the node list. A more detailed discussion of the scenarios can be found in chapter 4, where the proposed protocols are simulated in a Bluetooth piconet wireless environment.

3.3 Simulation Model

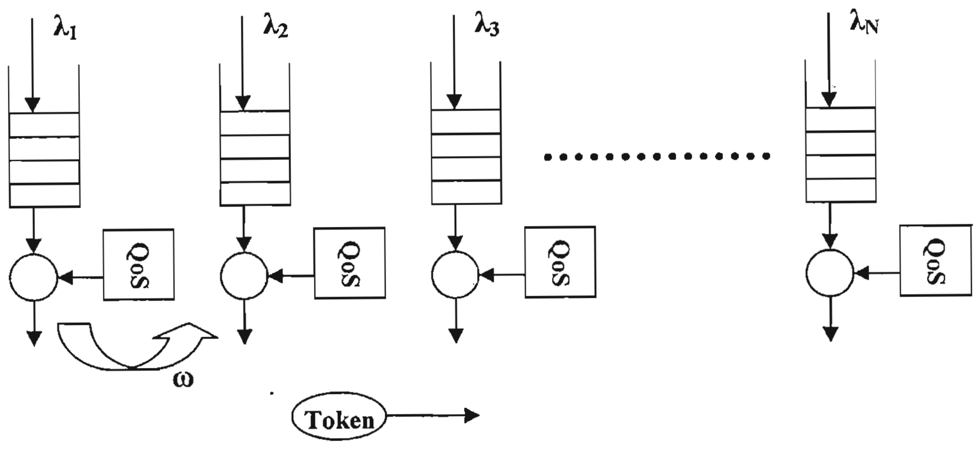


Figure 3.4: System model for the proposed MAC protocol

A simulation model that allows for the direct investigation of the performance of the proposed MAC scheme has been created. The simulation model is shown in Figure 3.4 and the assumptions can be summarised as follows:

A1. The network is distributed with nodes scattered in the network.

A2. Each node has no problem forwarding the token to the adjacent nodes in the network.

A3. A perfect environment where no interference exists is assumed. Each node is equipped with a transceiver that has perfect power control. Transmission errors do not exist.

A4. Markov Modulated Poisson Process (MMPP) [Fishcher, 1992] models the traffic generated by each node with two states: ON and OFF. The ON and OFF times are set to have means of 10 ms and 90 ms respectively.

A5. Steady-state simulation is used where it is assumed that the token and nodes will not drop out of the network.

A6. The token walk time from one node to the subsequent node is determined by the time needed to transmit the token (token transmit time) and transfer the token to the subsequent node (token transfer time). For the simulation, the token is assumed to be travelling at 66% ($\frac{2}{3}$) the speed of the light and, since the nodes are mobile in the network, the token walk time is not a constant parameter.

A7. It is assumed the data packet only consists of a single category of data with no header or trailer. In addition, the simulation assumed perfect transmission with no errors, where the error checking mechanism was not necessarily incorporated into the data packet.

A8. For the simulation purposes, it is assumed that the system would have equal number of nodes in different traffic classes in order to clearly distinguish the difference in delay and throughput performance between different classes.

A9. In order to constrain the activity of the mobile nodes, the simulation is conducted in the area of 1km by 1km network [Lihui, 2000]. This constraint is important as the simulation also takes the distance factor into consideration when calculating the token walk time, which contributes to the packet delay. If the distance between the successor and predecessor nodes is infinite (without boundary), then the token walk time would be infinite, resulting in an infinite delay.

3.3.1 Simulation model for a MAC scheme with data rate QoS guarantee

In order to monitor and evaluate the performance of the proposed QoS guarantees, in the first stage, a simulation model is constructed to simulate the proposed MAC protocol with only the data rate QoS guarantee incorporated. In the second stage, another simulation model is proposed that will accommodate both data rate and delay level QoS guarantees. The simulation model for the first stage can be modelled as shown in Figure 3.4 by disabling the delay level QoS function. Simulation parameters used are shown below:

- A single token in a network
- The token has a capacity of M CDMA codes, $M = 8$
- Each wireless code channel has a link speed rate of 128 kbits/s

- The area of the network is set to 1[km] by 1[km]
- There are 20 nodes in a network
- Each node is moving at a constant velocity of 1meter/minute within the network
- There are two traffic types with data rates λ_1,λ_2 (kbits/s)
- λ_2 is set to be twofold of λ_1 ($\lambda_2 = 2\lambda_1$)
- Half of the nodes in the network use data rate 1 (λ_1)
- The other half of the nodes use data rate 2 (λ_2)
- The token length is set to 40 bits
- The packet length is set to 160 bits
- The token walk time (ω) between two nodes is determined by the token transmit and token transfer time as discussed in assumption A6.

3.3.2 A simulation model for a MAC scheme with data rate and delay level QoS guarantees

For the MAC scheme that incorporates two QoS guarantees, the system model is the same as in Figure 3.4. However, the simulation parameters used for this model are not entirely identical to those in 3.2.1. In order to evaluate this more complicated model, additional parameters have been proposed. Simulation parameters used for this model are shown as follows:

- A single token network
- The token holds M CDMA codes, $M = 8$
- Each wireless code channel has a link speed rate of 128 kbits/s
- The area of the network is set to 1[km] by 1[km]
- There are 20 nodes in the network
- The permit generation parameter k for the data rate QoS is set to 0.5, $k = 0.5$
- Each node is moving at a constant velocity of 1meter/minute within the network
- The token length is set to 40 bits
- The packet length is set to 160 bits

- The token walk time (ω) between two nodes is determined by the token transmit and token transfer time as discussed in assumption A6.
- Four traffic types are introduced
- There are two different sets of data rates (λ_1,λ_2)
- λ_2 is set to be twofold of λ_1 ($\lambda_2 = 2\lambda_1$)
- Two delay levels are specified in terms of (N_1, N_2)
- There are four classes of traffic types, defined by the pair {λ_i, N_i}
- Each class contains a quarter of 20 nodes with the same data rate and delay level
- Class 1 (Low Data Rate, High Priority)
- Class 2 (Low Data Rate, Low Priority)
- Class 3 (High Data Rate, High Priority)
- Class 4 (High Data Rate, Low Priority)

3.4 Simulation Results

The MAC protocol has been simulated with the above system models by an event-driven simulation. C++ Builder software package was used to develop the simulation program. The purpose of the simulation was to determine the effect of different parameter settings on the performance of the network. In particular, the system delay and throughput was evaluated as a function of the number of nodes in the system, number of codes available for the network (*NOC*), node data rate (λ), and the system loading per code. The system load displayed in the results is defined as,

$$System\ load = \frac{System\ Traffic}{System\ capacity} = \frac{\sum_{i=0}^{9} \lambda_{i(LD)} + \sum_{i=10}^{19} \lambda_{i(HD)}}{M \cdot Channel\ capacity} \quad (3.2)$$

Both the node data rate (λ_i) and channel capacity in (3.2) are measured in kbits/s, and channel capacity is assumed to be a fixed parameter of 128 kbits/s. The node data rate is assumed to be a variable that changes between high (HD) and low (LD) traffic classes.

3.4.1 Simulation results for a MAC scheme with data rate QoS guarantee

For this MAC scheme, the results attained from monitoring the throughput and delay for two of traffic classes (high and low data rate) as well as entire network are examined. The term throughput is defined as the traffic exiting the monitoring entity per unit time. Two types of entity are monitored in this simulation; throughputs for the complete network, and traffic classes. The delay is defined as the packet delay and is the duration of time that a packet stays in the node's data buffer. The results of the simulation are shown in Figure 3.5-3.7. Figure 3.5 plots the packet delay for the entire network under different traffic conditions by varying the data rate QoS guarantee's proportional constant *k*. Figure 3.6 plots the packet delay for two different traffic classes and the overall delay in the network. Figure 3.7 plots the throughput for the entire network and the traffic classes under different network traffic parameters.

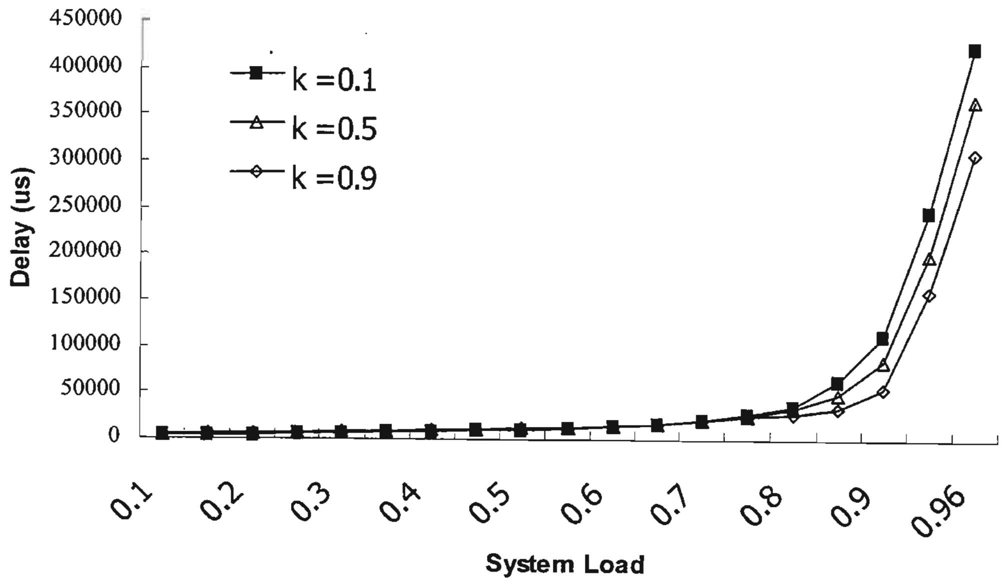


Figure 3.5: Overall packet delay when varying data rate QoS's parameter *k* under different traffic conditions

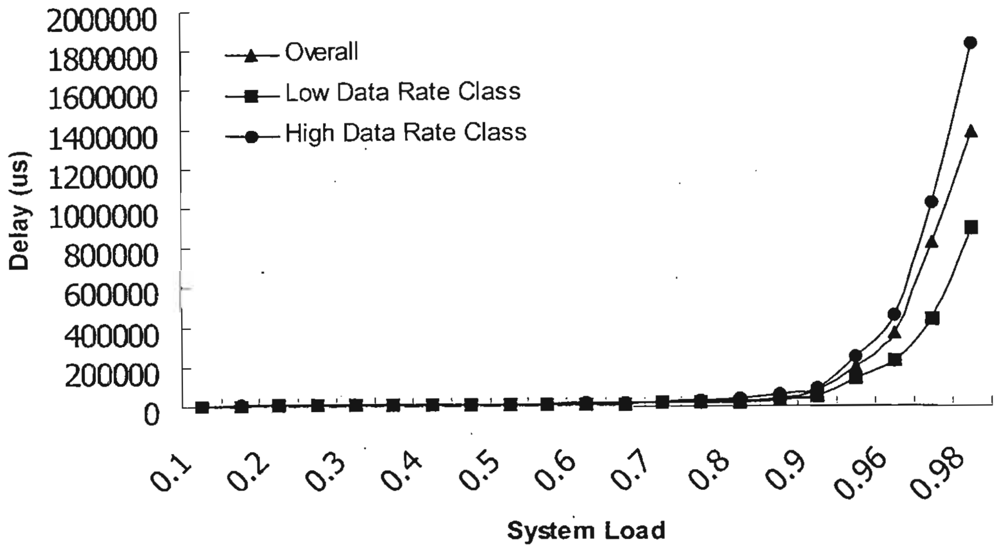


Figure 3.6: Comparison of packet delay in two different traffic classes and overall delay in the network that consists of 20 nodes and 8 CDMA code channels with k = 0.5

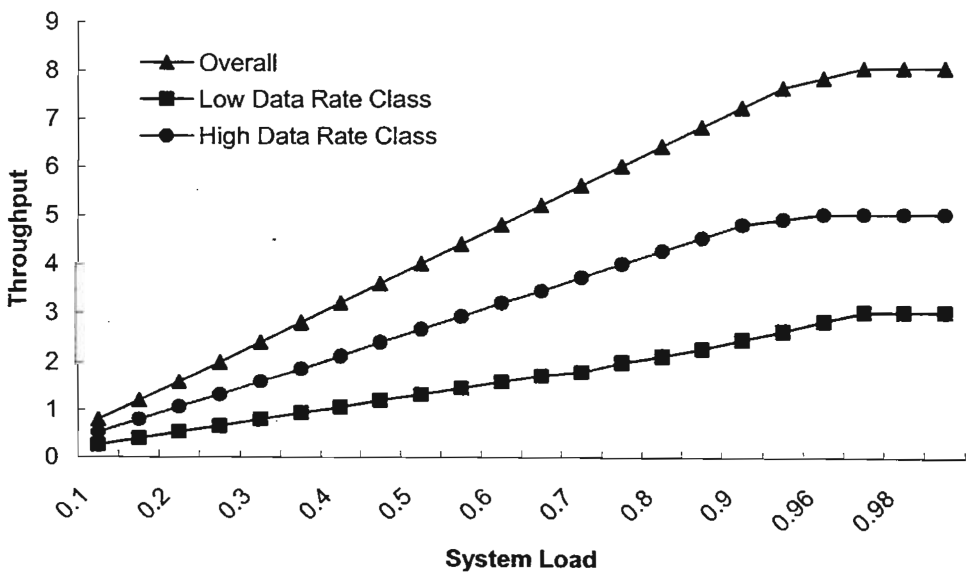


Figure 3.7: Comparison of throughput for two traffic classes and whole network that consists of 20 nodes and 8 CDMA code channels with k = 0.5

To observe how data rate QoS's permit generation proportional constant parameter, *k*, affects the system performance. A simulation was conducted by varying the values of *k* whilst using one system load condition. The simulation was then repeated for numerous system load settings. Figure 3.5 was then plotted, by applying different *k* constraints, it can be observed from the result that the QoS constraint, *k*, had provided regulations to the system performance. From (3.1), it is known that as *k* increases, more packets will be allowed to transmit, thereby decreasing the packet queuing delay as less packets queue to receive service. This can be seen by the increase in the delay performance of the system as the value of *k* increases, validating the prediction. Data rate QoS is therefore efficiently applied, as the permits effectively control the traffic flow of the network.

Figure 3.6 demonstrates that the network is efficient, as the packet delay is minimal at light to medium load and deteriorates at heavy load. It can be seen that the packet delay increases exponentially and deteriorates quickly when a heavy load is applied. This is as predicted, since the overall channel capacity is assigned as a fixed parameter, and as the network traffic generated in the network increases; the channel capacity reaches a maximum and it is unable to manage the network traffic, consequently leading to the exponential increase in packet delay. The token passing strategy also leads to the effect of exponential increase in delay under heavy load condition since it is known that in the token passing system, a packet suffers exponential increased in delay when the system experiences heavy load. It is illustrated in Figure 3.6 that the packet delay experienced by the high data rate class is higher than that of the low data rate class. This is as predicted since the high data rate class generates twofold the amount of packets than low data rate class, therefore the time needed for the packets in the high data rate class to receive service is longer than that of the low data rate class.

Figure 3.7 displays the code utilization for both the high and low data rate classes, as well as the network. It can be clearly seen that a maximum value of 8 is reached for the network throughput performance. This is as predicted since 8 CDMA channels are distributed in the network for the simulation. Since the high data rate class generates more packets than low data one, the throughput performance for the high data rate class would than be higher than its counterpart. The assumption is verified from the throughput attained from Figure 3.7.

3.4.2 Simulation results for a MAC scheme with data rate and delay level QoS guarantees

In this section, the proposed delay level QoS mechanism is expected to maintain the different levels of delay priority between classes. The two priority classes have the same number of nodes. Three different types of results are examined for this MAC scheme: the throughput, delay and the effectiveness of the delay level QoS guarantee for each class and the entire network.

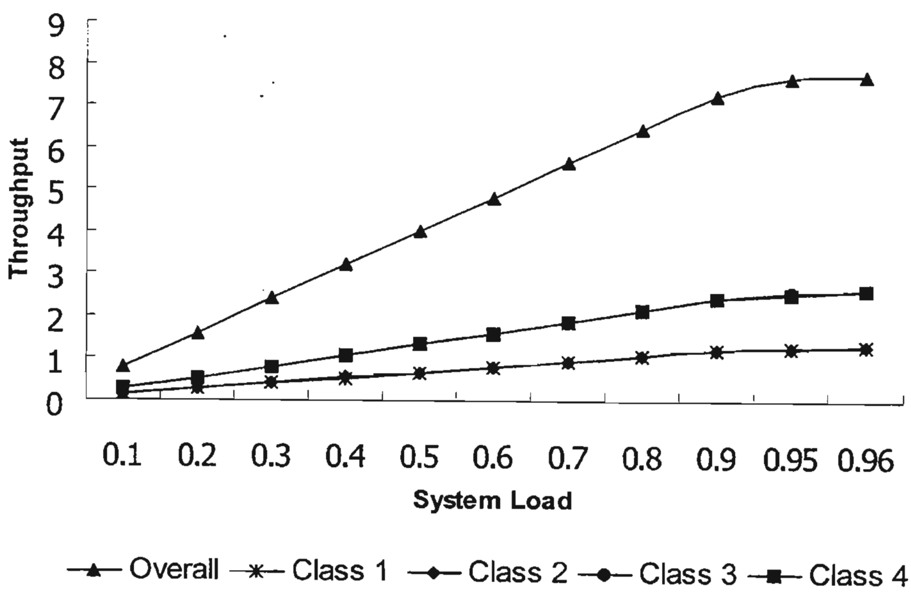


Figure 3.8: Average code utilization for the network and all classes with the channel reservation parameter, *N*, set to half of the codes thereby reserving half of the codes for the high priority classes

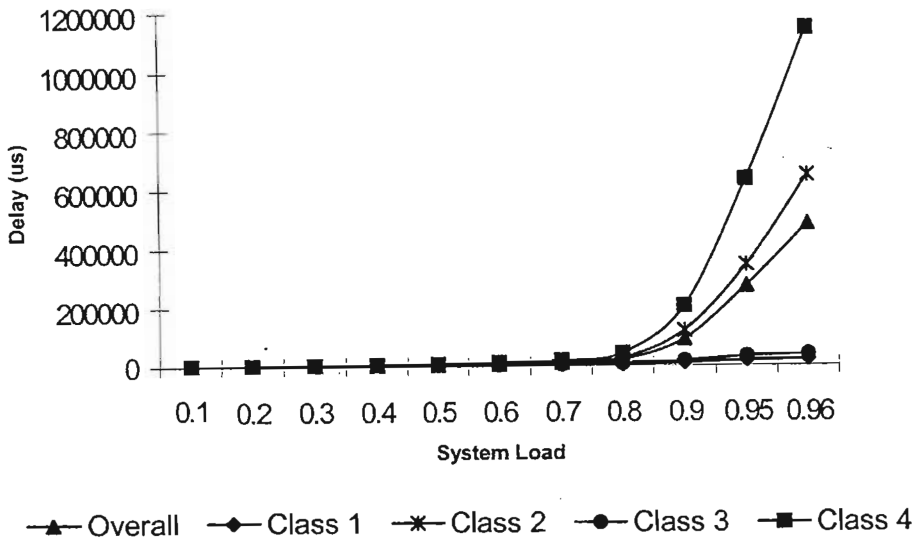


Figure 3.9: Packet delay performance for the network and each class with the channel reservation parameter, *N,* set to half of the codes thereby reserving half of the codes for the high priority classes

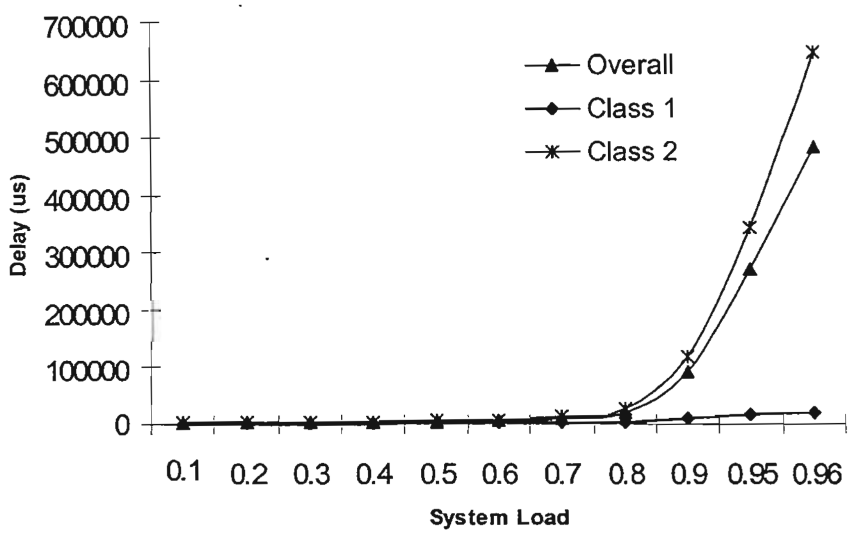


Figure 3.10: Packet delay performance for the network and both classes with a low data rate (classes 1 and 2) and the channel reservation parameter, *N,* set to half the number of codes, thereby reserving half of the codes for the high priority classes

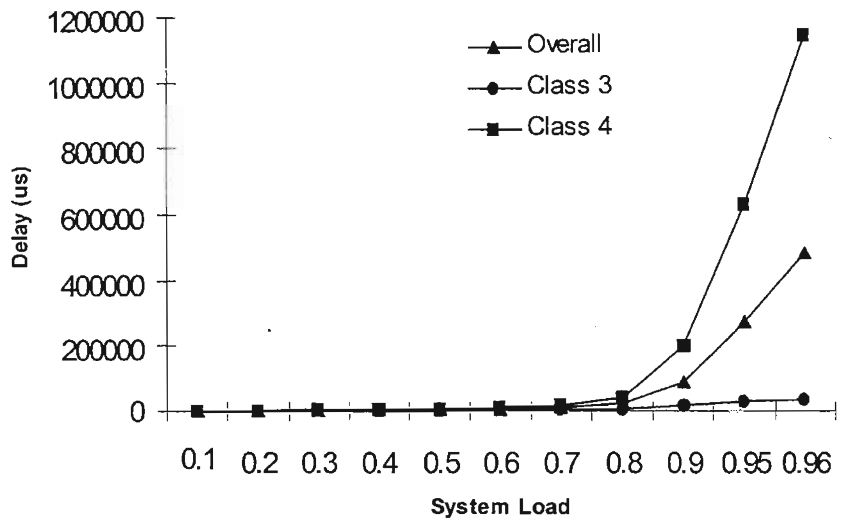


Figure 3.11: Packet delay performance for the network and both classes with a high data rates (classes 3 and 4) and the channel reservation parameter, N, set to half the number of codes thereby reserving half of the codes for the high priority classes

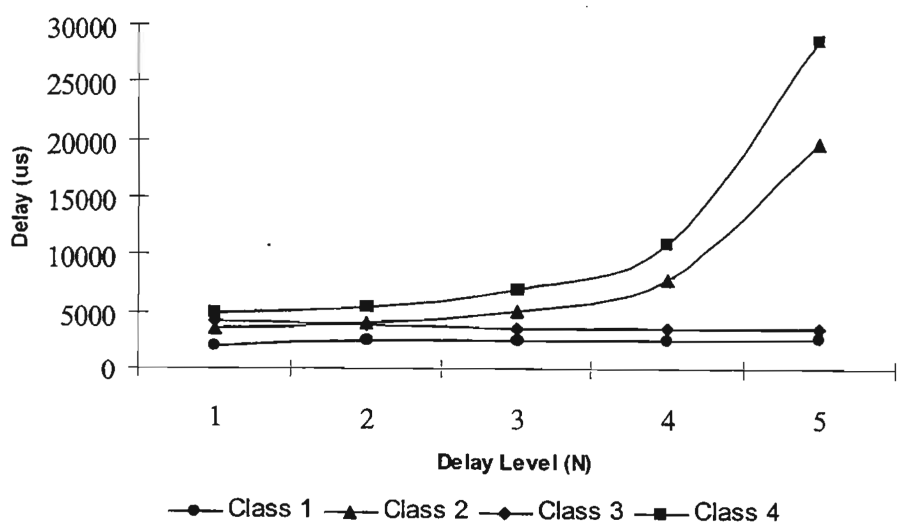


Figure 3.12: Monitoring the effectiveness of the delay level QoS by observing the packet delay performance of each class with fixed system load and different channel reservation parameter (N) values

Figure 3.8 displays the code utilization for the entire network and all classes. It can be seen that for the overall network throughput performance, the maximum value of 8 is reached as predicted, since 8 CDMA code channels are distributed in the network for the simulation. The throughput performance of class 3 and 4 are identical; this is as predicted since they generate the same amount of data packets. Similar throughput performance are also found for the low data rate classes 1 and 2. Both classes 3 and 4 have higher throughput than classes 1 and 2; this is as predicted as classes 3 and 4 generate twofold the data packets of classes 1 and 2.

Figures 3.9, 3.10 and 3.11 show the delay performance of the proposed scheme under the condition that half of the codes are reserved for the high priority classes. The difference in the delay experienced between the four classes is due to the different nature of the QoS mechanisms that are employed by the proposed protocol. It can be seen that the network is stable as the delay for each class is relatively low under medium-to-light traffic loads. However, as a heavy load is applied, the delay increased exponentially; this is due to the fact that the pre-defined channel rate cannot provide sufficient channel capacity. It can be clearly seen from Figures 3.10 and 3.11, that the delay level QoS guarantee is efficiently applied in the network. This is demonstrated as the delay for the low priority classes (2 and 4) increases exponentially when a heavy load is applied, whereas the delay for the high priority classes (1 and 3) remains stable. This is to be expected since when the load is high in the network, the delay level QoS is applied to ensure the high priority classes (1 and 3) receive sufficient access.

In order to monitor the effectiveness of the delay level QoS guarantee, the network load traffic was fixed, and the delay level parameter, N, was varied and the outcome observed. As discussed in section 3.2.5, as the value of the delay level, N, increases, more codes are reserved for the higher priority classes. Figure 3.12 shows a clear separation between the plots of the different classes. Nodes of class 4 (low priority, high data rate) experience the longest delay whereas nodes of class 1 (high priority, low data rate) experience the shortest delay.

It can be seen from Figure 3.14 that when N is low, the delay of the high and low priority classes merge and the priority scheme disappears. However, when N increases, the difference in the delay in packet transmission increases exponentially between the high and low priority classes. A priority scheme is clearly established. The delay for the low priority classes (2 and 4) is therefore increased whereas the delay for the higher priority classes (1 and 3) remains unaffected. As predicted, this occurs when the delay level QoS guarantee gets more stringent, and hence the lower priority classes have fewer opportunities for transmission.

3.5 Summary

In chapter three, a CDMA medium access scheme using token passing protocol was proposed for a distributed wireless ad hoc network. The aim was to propose a protocol that incorporates the advantages of a guaranteed access scheme into a distributed type of wireless network.

A MAC scheme that incorporates a data rate QoS guarantee was first proposed, with the objective of providing fairness within the network. It was shown from the simulation result that the proposed MAC scheme had a low average packet transfer delay, as well as a high maximum throughput. This system stays stable under varying traffic conditions.

Next, an additional delay level QoS guarantee was introduced providing priority access to the network. By incorporating both data rate and delay level QoS guarantees into the proposed MAC, it can be seen from the simulation result that under heavy traffic load, the QoS requirements are still maintained. Both delay and throughput performances are optimised at light to medium loads.

CHAPTER 4

THE PROPOSED MAC SCHEME WITHIN THE BLUETOOTH PAN

4.1 Introduction

In this chapter, we adapt the proposed MAC protocol to the Bluetooth PAN (personal area network) to observe the performance and efficiency of the proposed MAC. The PAN is defined as the collection of devices carried by a mobile, networked individual. The devices include any form of cell phone, laptop, palm pilot, etc. These devices form his/her PAN. The connectivity within the PAN is wireless and the PAN can expand and contract dynamically depending on needs, and may be used for ad hoc communication or as internet access technology. In section 4.2, the Bluetooth system is discussed in detail. In section 4.3, the complexities and techniques involved in adapting the proposed MAC to a Bluetooth PAN environment are discussed. Section 4.4 presents the simulation model and discusses the results.

4.2 Bluetooth System

Bluetooth is a short-range radio technology operating on the 2.4 GHz unlicensed ISM band. It uses frequency hopping and Time Division Duplex (TDD) techniques to transmit data. The nominal communication range is set to 10 meters, but can be extended to 100m. Bluetooth devices are organized in an ad hoc, piconet structure. There is one Bluetooth device in each piconet that acts as a master, and can admit up to seven active slaves. A network that consists of several overlapping piconets is called a scatternet, as depicted in Figure 4.1.

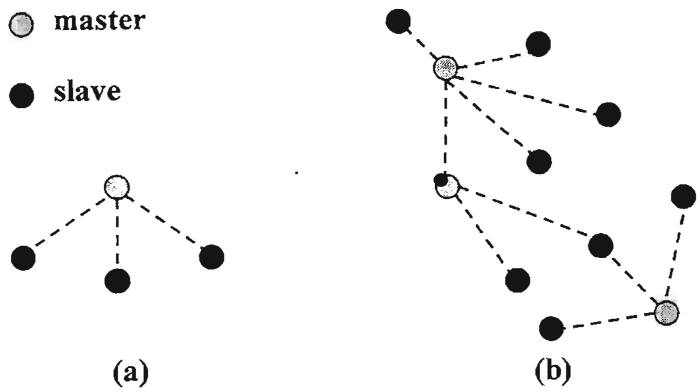


Figure 4.1: Bluetooth network scenario: (a) piconet and (b) scatternet

4.2.1 Master/Slave definition and network topology

It is defined in the Bluetooth network that each node has identical hardware and software interfaces and is distinguished by a unique 48-bit address. When the network is setup, the initializing node is assigned as the master; however such assignment is valid only during this connection. The role of the master node is to initiate the connection and control the traffic of the connection. Each slave node is assigned a temporary 3-bit member address to reduce the number of addressing bits required for active communication.

As shown in Figure 4.1, the Bluetooth network supports both point-to-point and point-to-multipoint connections. A different frequency-hopping channel defines each piconet. All nodes participating in the same piconet are synchronized to this channel.

4.2.2 Establishing network connections

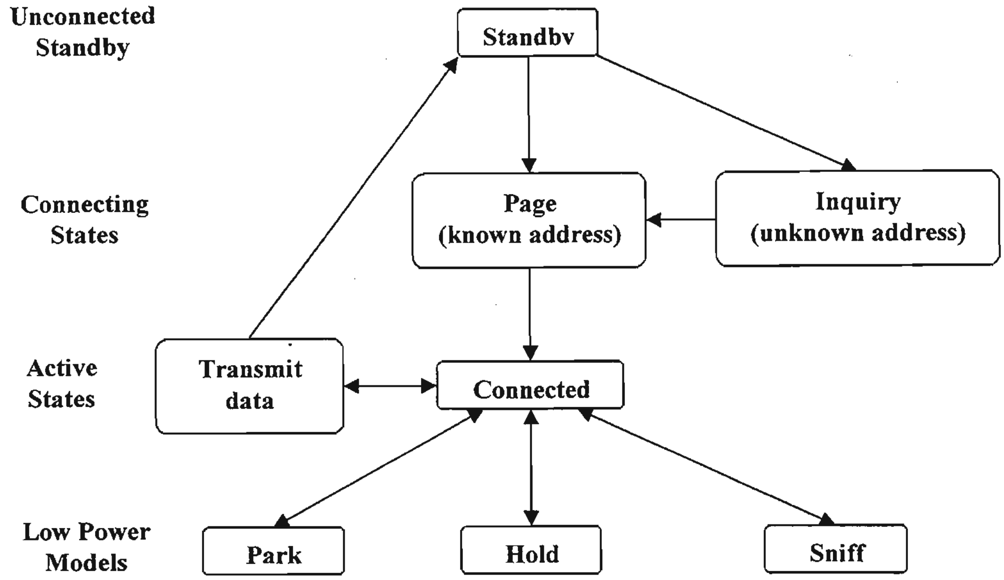


Figure 4.2: Connection state machine

To establish a network or add nodes to a piconet, the nodes must first be identified. Nodes can by dynamically connected and disconnected from the piconet at anytime. Before any connections are made, all nodes are in standby mode. Idle nodes periodically wake up and listen for packets destined for them, and if they receive one, a connection is made. There are two different scenarios (Page, and Inquiry) for establishing connections. One is where the master knows the address of the slave (Page scenario), and the other is where the master does not know the address of the slave (Inquiry scenario) as depicted in Figure 4.2.

- Master node knows the slave node's address

If the master knows the idle node's address, it sends a PAGE message to the slave node's access code. The access code is a number based on the slave node's unique address. The idle node periodically wakes up and scans on a hop carrier for 10 ms, looking for PAGE messages that are destined for its access code. During the idle node's sleep period, the master node transmits PAGE messages on 16 different hop carriers. It cycles through the carriers so that a PAGE message is transmitted on each frequency. The master node alternates between transmitting on 2 carriers and listening for responses. Since it does not know when the idle node will wake up, it cycles through the carriers repetitively for 1.28s. If no response is received after this time, the master node transmits a train of 16 identical PAGE messages on the remaining 16 hop frequencies in the wake up sequence.

- Master node does not know the slave node's address

Similar procedures are applied to the scenario where the master does not know the slave's address, with some modifications. In this scenario, the idle node still wakes up and scans the same way, however it will also respond to INQUIRY messages sent to a reserved access code. The slave nodes implement a random back off approach in order to prevent multiple idle nodes from responding to the INQUIRY message and causing collisions. The master node uses the same frequency hopping technique as discussed earlier. When the idle node receives the INQUIRY message, it replies back by sending its own access code to the master node. Therefore, after the INQUIRY phase, the master can send a PAGE message since it now knows the slave's address. The procedures that follow are the same as those discussed previously.

Once the connection is established and if there is no data to transmit, the nodes will stay in the HOLD state. However, data transmission can be restarted instantaneously when the nodes wake up from the HOLD state. This keeps nodes connected in a low power mode to save battery life. There are two more low power states available, the SNIFF state and the PARK state respectively. The SNIFF state is a power saving mode where the master and slave have negotiated intervals when they can sleep. The PARK state is also a power saving mode where the node releases its 3 bit member address, but remains synchronized to the piconet's hopping pattern. The SNIFF, HOLD, and PARK states are listed in increasing power-saving order.

4.2.3 Link types

Once a Bluetooth unit has been connected to a piconet it may communicate by using one of two link types. These links are:

- Synchronous Connection Oriented (SCO) link
- Asynchronous Connectionless (ACL) link

The SCO link is a point-to-point full duplex link between the master and slave. The master generates this link and keeps it alive. The SCO link is typically used for voice traffic. The master reserves slots used for the SCO link on the channel. The ACL link makes a momentary connection between the master and any of the slaves for the duration of one frame (master to slave slot and slave to master slot). No slots are reserved for this link. The master can freely decide which slave to address and in which order. A polling scheme is used to control the traffic from the slave to the master. This link is intended for asynchronous traffic.

4.2.4 MAC protocol

Bluetooth nodes use a polling MAC scheme as described in section 2.5, where the node that first made PAGE messages to other nodes is always defined as the master node. However, this master/slave designation is only valid for the life of the piconet. All nodes have the capability of being the master or slaves. With the functionalities that all the nodes are synchronized and implement the same hop sequence, Bluetooth uses a polling technique to allow multiple nodes access to the medium.

Since a polling technique is implemented, communication exists only between the master and the slave. Transmissions are categorized in either one of two conditions. The first condition is that the master has no data for the slave. The master then sends a poll message to the slave in one time slot, and the slave respond to the poll with data in the following time slot. The second condition is when the master has data for slave. In this case, the master sends the slave the data in one time slot, and the slave can respond to the master with data in the following time slot.

4.3 Adapting the Proposed MAC in a Bluetooth Environment

Since the Bluetooth system is a master driven time division duplex (TDD) system, it can be seen that there could be wastage of slots within the scheme, as when only the master or slave has data to send, a slot gets wasted. Kalia [Kalia, 2000] also suggests that the fairness issue is complicated for the Bluetooth system since service may be given to a master-slave connection even if the master and slave do not have data to send at the same time (backlogged). Due to the above reasons, Kalia mentioned and proved that for the TDD based MAC protocols implemented in Bluetooth, the system yields low throughput and may not ensure fairness. In this section we suggest an alternative approach by adapting our proposed MAC scheme in the Bluetooth system within a piconet network. The aim is to observe the performance of the proposed MAC when it is applied under the wireless Bluetooth environment.

4.3.1 Wireless Bluetooth environment

Due to the ad hoc wireless nature of the Bluetooth network, the nodes and token may potentially drop out of the network and become lost. Section 3.2.5.3 has briefly discussed how the proposed protocol would react in the wireless environment. We now have to take these scenarios into consideration: when a node or the token drops out of the network, and when new nodes joins the network.

4.3.1.1 The node or token drops out scenario

As previously described in section 3.2.5.3, the nodes use the token detection scheme to monitor the token's activity. By modifying the token structure, we assume that the token has a node list that depicts the status of the node and a code list to display which node has a code. Each list has a capacity of one byte. If a node is in the network, its status bit on the node list is "1", and if the node drops out the network, its status bit changes to "0". If a node has captured a code for transmitting data packets, its code list will be depicted as "1", otherwise it will depict as "0". Figure 4.1 and 4.2 demonstrate the node list and code list respectively. In figure 4.3, we can observe that all nodes are active except node 3 and in figure 4.4; we may see that node 1 and node 4 currently have the code.

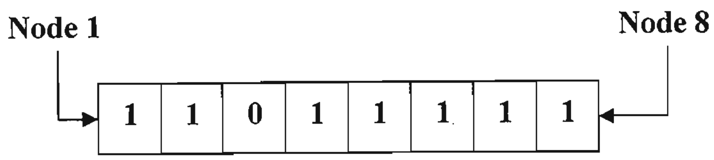


Figure 4.3: Node list in the token

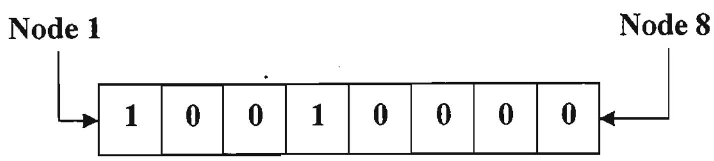


Figure 4.4: Code list in the token

During token transmission, if the predecessor node does not detect any token transmission activity from the intended successor node after the predefined time-out period, it will assume either the token or the node has dropped out the network. The predecessor node will then regenerate the token and change the successor node's status bit to "0" on the node list. The predecessor node will also examine its successor's status bit in the code list. If the successor node drops out while it is transmitting the data packet, the predecessor node will release the code by incrementing the *NOC* value by one, and change the successor's status bit to "0" in the code list. The predecessor node then searches for the next available node in the node list and forwards the token to that node. Figure 4.5(a) displays the flow chart for this scenario.

4.3.1.2 New nodes join the network

Once the network is initialized, new nodes that wish to join the network have to wait for n token cycles, as mentioned in chapter 3, n is a predetermined value that is assigned to each node when the network is initialized. A token cycle is denoted as the time it takes for a token to circulate through all the active nodes in the network. Each node in the network is equipped with a token-visit counter and each time the token visits the node, the counter is incremented by one. When any one of the nodes in the network reaches the value n while the token visits it, it initiates a checking procedure. This procedure checks if there are still nodes busy transmitting data by determining if the token has all the codes. If the token does not have all the codes, the node forwards the token to the next node in the node list without capturing any remaining codes for transmission. A new cycle begins only when the visited node observes that the token has all the codes and the value n is reached.

When a new cycle is initiated, the node that currently has the token broadcasts the welcome message; the active nodes reply to it by sending back its own address and reset their token-visit counter to zero. The node then sorts all the node addresses in the node list and forwards the token to the next node in the node list. Figure 4.5(b) displays the flow chart for initializing a new cycle in the network.

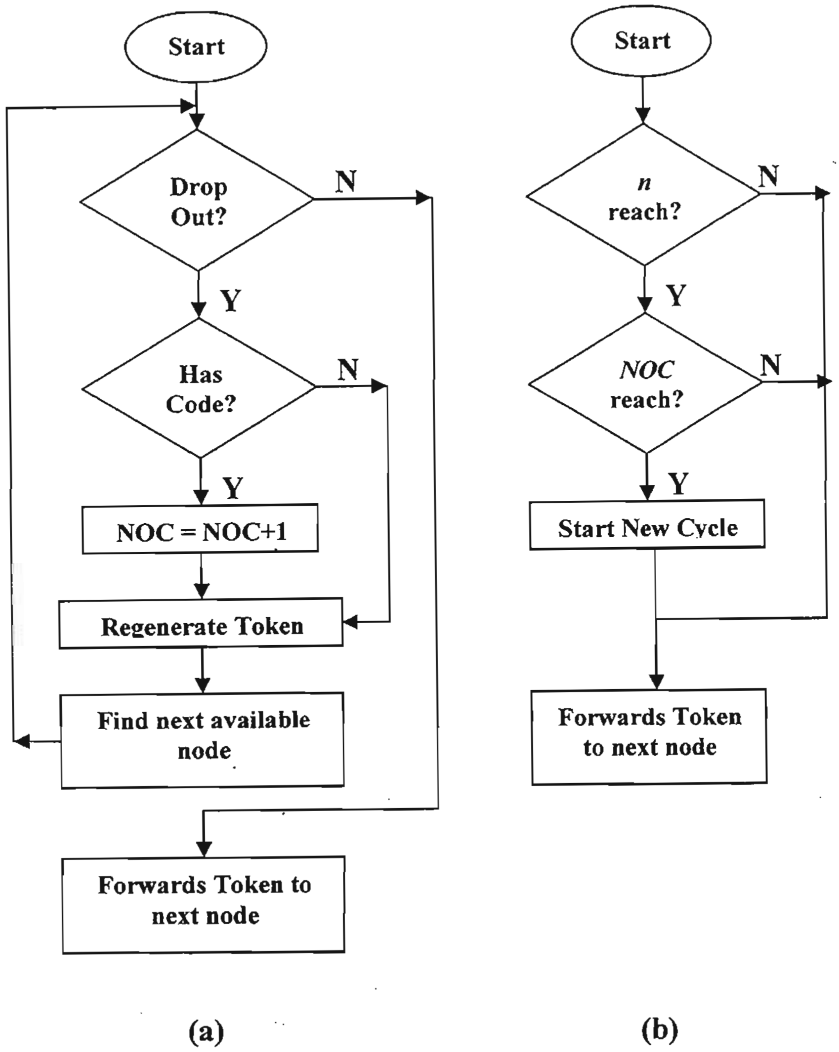


Figure 4.5: Flow charts for both (a) a node or token lost and (b) new cycle scenarios

4.3.2 Node and token structure

- **Node structure**

Nodes in the Bluetooth piconet network are assumed to all have the same functionality, as proposed in section 3.2.2. In this case, every node is equipped with a MUD (multiple user detector) and two full-duplex transceivers. Each node supports two QoS guarantees as proposed in chapter 3; they are data rate and delay level QoS respectively.

- **Token structure**

Based on section 3.2.3, the token structure is basically the same, except that a few more parameters are augmented. The token now consists of the source address, destination address, number of codes available (*NOC*), a node list, and a code list. Table 4.1 shows the token structure proposed for a Bluetooth network.

Source ID	Destination ID	NOC	Node List	Code List

Table 4.1: Proposed token structure for Bluetooth network

- Source ID (3bits): the address of the node who passed the token (predecessor's address).
- Destination ID (3bits): the address of the node that is currently holding the token.
- *NOC* (3bits): this parameter is used to limit the number of simultaneous transmissions in the network. For the nodes, the higher the priority, the smaller the *NOC* value.
- Node list (8bits): this field is used to monitor the status of the nodes in the network. If the node is active, the status bit is set to "1" else it is set to "0".
- Code list (8bits): this list is implemented to record which node is currently holding a code for data transmission. If a code is captured by a node, the node's status bit will be set to "1" in the code list.

4.4 Simulation Model and Parameters

The simulation model proposed is also a token-passing model as depicted in Figure 3.7. The assumptions used are as follows:

A1. The analysed network is distributed, with nodes scattered in a piconet network.

A2. A perfect environment where no interference exists is assumed. Transmission errors do not exist.

A3. Traffic models are generated using Markov Modulated Poisson Process (MMPP) [Fishcher, 1992] with two states: ON and OFF. The ON time and OFF time are set to have means of 10 ms and 90 ms respectively.

A4. A dropout model is created for the token and nodes and a dropout probability is assigned to decide whether the node or token will leave the network. Each time the dropout model is activated by a node or token, it generates a value between 0 and 1. If the generated value is less than or equal to the pre-defined dropout probability, then the dedicated node or token leaves the network.

A5. The token walk time from one node to the subsequent node is determined by the time needed to transmit the token (token transmit time) and to transfer the token to subsequent node (token transfer time). The token is assumed to be travelling at 66% ($\frac{2}{3}$) the speed of the light, and since the nodes are mobile in the network, the token walk time is not a constant parameter.

A6. It is assumed the data packet only consists of a single category of data with no header or trailer. In addition, the simulation assumed perfect transmission with no errors, where the error checking mechanism was not necessarily incorporated into the data packet.

Both the data rate and delay level QoS guarantees are incorporated in the MAC scheme. The simulation parameters used for this model are shown as follows:

- A single token network
- The token holds M CDMA codes, $M = 3$
- Each wireless code channel considered has a link speed rate of 128 kbits/s
- The area is set to 100[m] by 100[m]
- There are 8 nodes in a network
- Each node is moving at constant velocity of 1meter/minute within the network
- The permit generation parameter k for data rate QoS is set to 0.5, $k = 0.5$
- The token length is set to 25 bits
- The packet length is set to 160 bits

- The token walk time (ω) between two nodes is determined by the token transmit and token transfer time as discussed in assumption A5.
- Four traffic types are introduced
- There are two different sets of data rate (λ_1,λ_2)
- λ_2 is set to be twofold of λ_1 ($\lambda_2 = 2\,\lambda_1$)
- Two delay levels are specified in terms of (N_1, N_2)
- There are thus four classes of traffic types, defined by the pair $\{\lambda_i, N_i\}$
- Each class contains a quarter of the 8 nodes, with the same data rate and delay level
- Class 1 (Low Data Rate, High Priority)
- Class 2 (Low Data Rate, Low Priority)
- Class 3 (High Data Rate, High Priority)
- Class 4 (High Data Rate, Low Priority)

4.5 Simulation Results

To simulate the proposed MAC scheme in a Bluetooth environment, an event-driven simulation program was used as mentioned in chapter three. As the nodes or token may now drop out of the network, the complexities in the designing of the program are increased. In order to evaluate the performance of the implementation, the system delay and throughput has been evaluated as a function of the number of nodes in the system, number of codes available for the network (*NOC*), and node data rate (λ). Since the packet length is fixed at 160 bits, the MMPP is used as the traffic model with mean of 10ms ON state. By dividing the data rate (λ) to the packet length, the packet generation rate is obtained. The mean number of packets generated within the ON state can then be determined by dividing the duration of ON time by the packet generation rate. Four different types of results are examined: the throughput, delay, the effectiveness of the delay level QoS guarantee for each class and the entire network as well as and the effect of changing the dropout probability on delay and throughput performance.

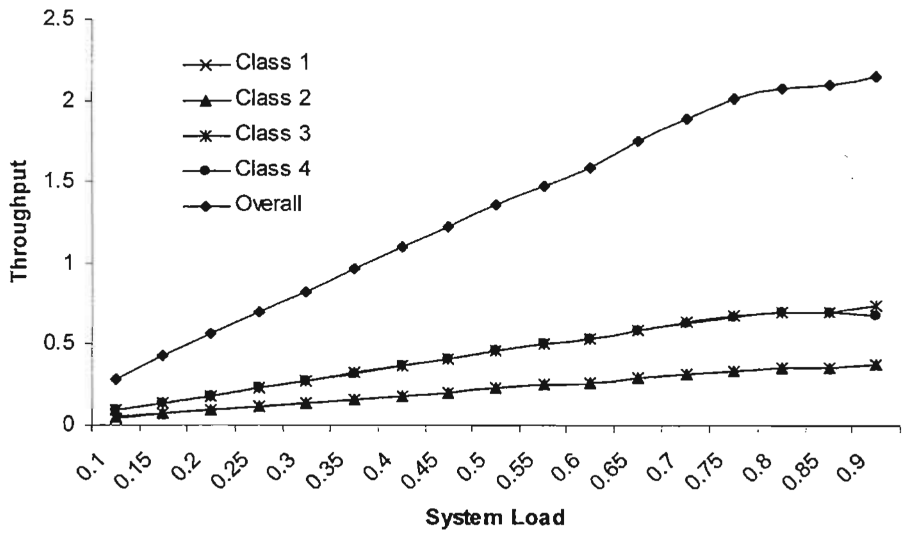


Figure 4.6: Average code utilization for the piconet Bluetooth wireless network and all classes with 5% dropout probability, 3 code channels (*NOC*=3) and channel reservation parameter, *N*, set to the value of one

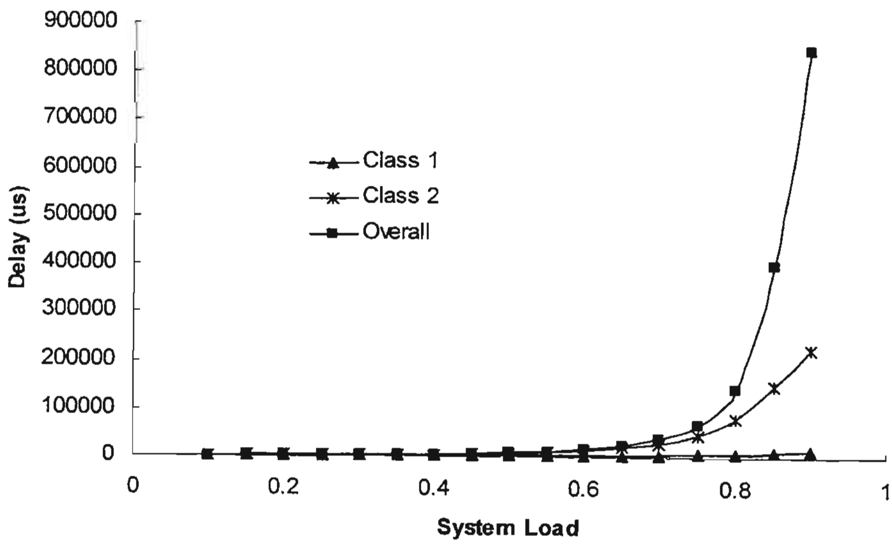


Figure 4.7: Packet delay performance for the piconet network and two classes with low data rate (1 and 2) under conditions that 5% dropout probability is assigned, 3 code channels (*NOC*=3) are used and channel reservation parameter, *N*, set to the value of one

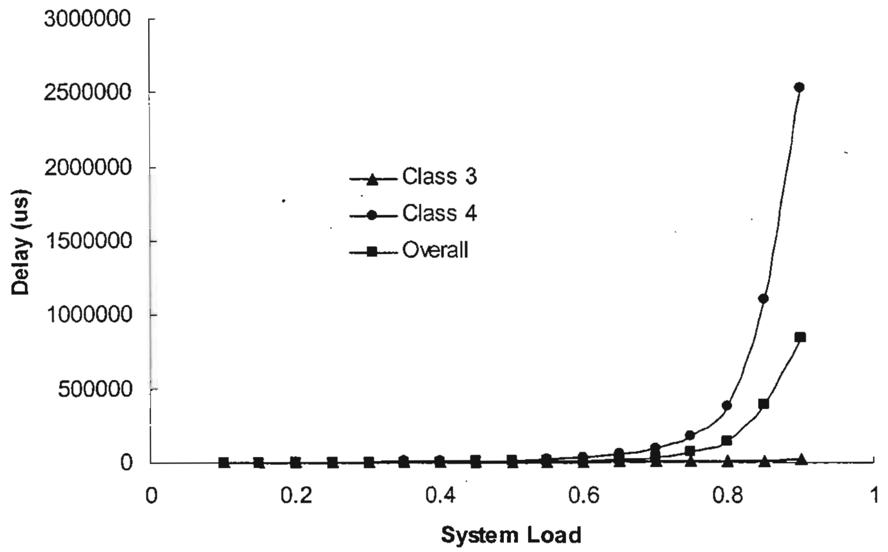


Figure 4.8: Packet delay performance for the piconet network and two classes with high data rates (classes 3 and 4) with 5% dropout probability, 3 code channels (*NOC*=3) and channel reservation parameter, *N*, set to the value of one

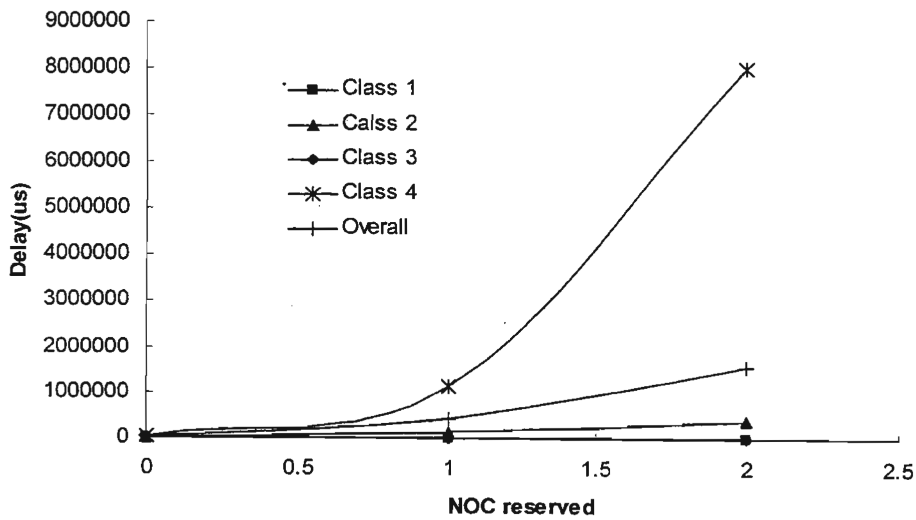


Figure 4.9: Monitoring the effectiveness of the delay level QoS by observing the packet delay performance of each class, achieved by fixing the *NOC* value to three and varying channel reservation parameter (*N*)

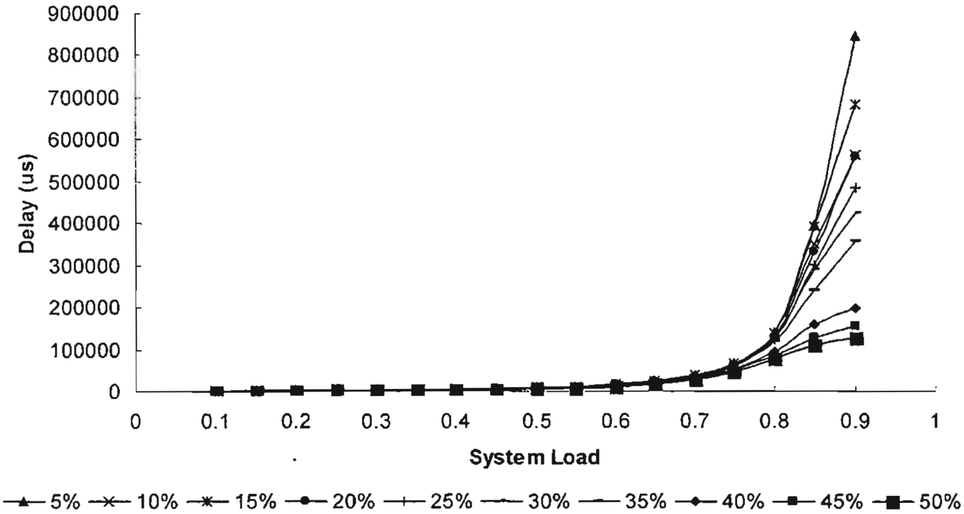


Figure 4.10: Monitoring the effect on delay performance when adjusting the dropout probability of the token and nodes from 5 % to 50% under the conditions that $NOC = 3$ and channel reservation parameter, $N = 1$

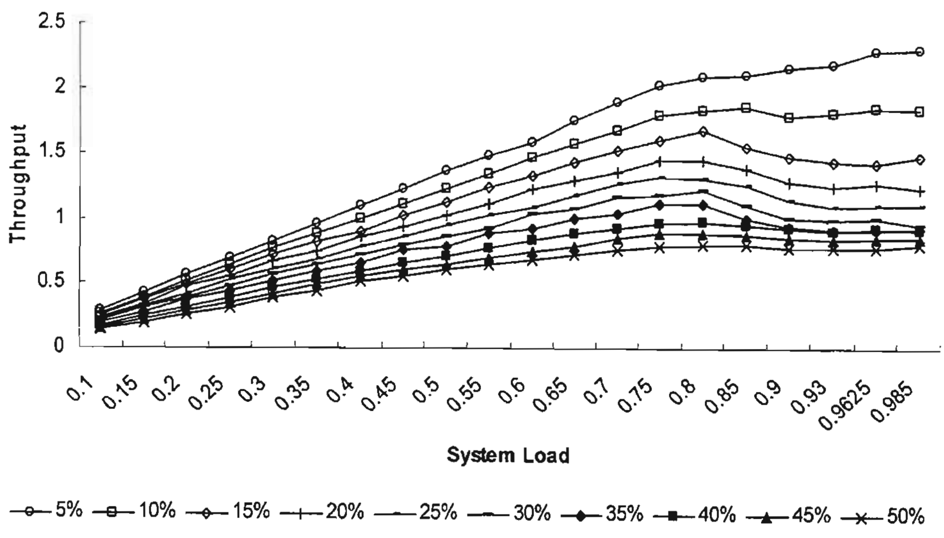


Figure 4.11: Monitoring the effect on throughput performance when adjusting the dropout probability of the token and nodes from 5 % to 50% with $NOC = 3$ and the channel reservation parameter, $N = 1$

Figure 4.6 depicts the numerical results obtained for code utilization for the entire network and all classes under different system load conditions where the system load is previously defined in Chapter 3. With the *NOC* parameter set to three, it was noticed that for overall network throughput performance, the maximum value of 3 is not reached although 3 CDMA code channels (*NOC* = 3) are distributed in the network for the simulation. This is to be expected since the token and nodes drop out the network, thereby decreasing the number of packets that can receive service. Consequently, the throughput can therefore not reach the maximized value of three. Considering the different classes, the throughput performance on the high data rate classes 3 and 4 are identical. This is expected since they generate the same amount of data packets. Similar throughput performance is also found for the low data rate classes 1 and 2.

The simulation results for packet delay also follow similar steps to those presented in chapter 3. Figures 4.7 and 4.8 display the delay performance of the proposed scheme under the condition that one third of the codes are reserved for high priority classes (N=1). From both figures, it can be seen that the network is stable under medium-to-light traffic loads as the delay for each class is relatively low. As soon as a heavier load is applied, the delay increases exponentially. The delay level QoS guarantee is then efficiently applied in the network, illustrated in that the delay for low priority classes (2 and 4) increase when the heavy load is applied whereas the delay for high priority classes (1 and 3) remains stable.

Similarly to chapter 3, the channel reservation parameter N is varied and the load is fixed in order to monitor the effectiveness of the delay level QoS guarantee. As previously mentioned in section 3.2.5, as the value of the channel reservation parameter, N, increases, more codes are reserved for higher priority classes. Figure 4.9 depicts a clear separation between the plots of different priority classes. Nodes of low priority classes 2 and 4, experience the longest delay, where nodes of high priorities classes 1 and 3 experience the shortest delay. It can be seen from Figure 4.9 that when N is low, the delay of high and low priority classes merge and the priority scheme disappears. The condition changes as soon as *N* increases; the difference in delay in packet transmission increases exponentially between high and low priority classes. This is as predicted.

In order to observe the effect of adjusting the dropout probability on delay and throughput performance, simulation results were obtained. It was assumed that the dropout probability determines the probability of the node or the token dropping out of the network. The higher the dropout probability, the higher the possibility that the token or node drops out. Fixing the NOC to 3 and N to 1 but varying the dropout probability between 5 to 50%, the simulation was conducted. The overall packet and throughput performance was then recorded.

Figure 4.10 displays the delay performance for the entire network. It can be seen that the overall packet delay for different dropout probabilities merges at light-to-medium loads. And, as the load increases, packet delay increases exponentially. However, it was also observed that if the dropout probability was increased, the packet delay decreased. This is to be expected since when nodes drop out of the network, the packets that reside in the node's buffer drop out together with the node. Packets generated from the newly joined node have a shorter arrival time; therefore the delay for packet delivery time is shortened. This packet loss effect is illustrated in Figure 4.11, which shows that as the dropout probability increases, the overall network throughput decreases.

4.6 Summary

In chapter four, a modified version of the proposed MAC protocol was introduced to adapt the proposed MAC protocol in the wireless environment. The modified protocol uses the same concept of token passing to maximize performance. The simulation model was proposed and simulated in the Bluetooth piconet wireless environment. Numerous simulations were performed with the simulation model with various network parameters. The goal was to observe whether the modified MAC protocol could still efficiently support the incorporated QoS guarantees within a wireless environment.

From the presented simulation results, it is clearly shown that when considering delay performance, the system remains stable under light-to-medium traffic load conditions. Once a heavy load is applied, delay performance for low priorities deteriorates gracefully in order to support delay level QoS guarantees.

Furthermore, the effect on network performance of varying the dropout probability was simulated. The results were found to be as predicted, where the higher the probability, the fewer the packets that receive service, thereby degrading both the delay and the throughput performance.

CHAPTER 5

APPROXIMATE DELAY ANALYSIS ON THE PROPOSED MAC PROTOCOL WITH DATA RATE QoS GUARANTEE

5.1 Introduction

In this chapter an analytical model for the MAC scheme described in chapter 3 is proposed. The performance of the analytical model is examined by conducting an approximate packet delay analysis. For the proposed CDMA MAC scheme, a token is circulating in the network distributing codes; nodes need to capture the code to attain access to the channel. This model is similar to the multiple server model, where the nodes in this type of network also need to capture the server to be granted access to the channel. Therefore, the proposed analytical model is built based on a multiple server scheme.

It has been stated in numerous papers in the literature, for example [Borst, 1997], that multiple server systems are extraordinarily hard to analyze, and no exact results are derived for models with independent servers, apart from some mean-value results for global performance measures like cycle times, e.g., [Qing, 1989], [Hamacher, 1989], [Kamal, 1994], [Borst, 1997] and [Ho, 2000]. Most of the proposed analyses used exhaustive, gated or 1-limited service policy to service the packets, and none of these papers incorporate any QoS guarantees into the analysis. The following analysis is based on deriving the sojourn-time approximations for a symmetric multiple server system that incorporates data rate QoS guarantee. In this analysis the delay level QoS guarantee is not incorporated.

The remainder of the chapter is organized as follows. A detailed analytical model description is presented in section 5.2. Section 5.3 describes the approximate analysis conducted to derive the mean packet delay. Sections 5.4 and 5.5 present the expressions needed to find the mean packet delay. In section 5.4, the mean queue length expressions are derived for the proposed model using the Markov chain approach. Following this, sections 5.5 presents derivations for token inter arrival time and the token rotation time. In section 5.6, results obtained from the analytical expressions are presented.

5.2 Analytical Model Description

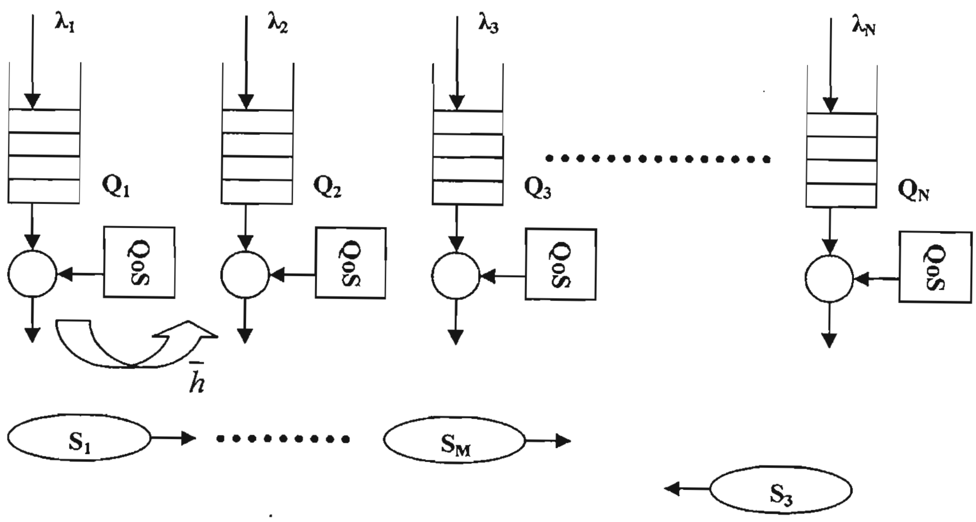


Figure 5.1: A multiple queue system with M multiple cyclic servers

As discussed in section 5.1, the analytical model proposed is considered as a multi-queue system, consisting of M cyclic servers $S_1,\ldots,S_M$ and with N queues $Q_1,\ldots,Q_N$, as shown in Figure 5.1. It is assumed that each server represents a code channel and each queue resembles a node. Each queue, i, is assumed to have an infinite capacity, into which packets arrive according to a Poisson process of rate λ_i. Packets in the system have a fixed size of s bits. Since the code channel rate is assumed to be constant in the analysis, the service time of a packet is then assumed to be a constant of T_p seconds.

When a server visits a queue, the packets are serviced only if the data rate QoS guarantee is satisfied. A detailed description of the proposed QoS is discussed in the next section. It is assumed that each of the M servers moves independently from one queue to another according to a predefined and fixed schedule. The walk time of any server from queue i to the subsequent queue, $i+1$, is assumed to have a mean of $\bar{h}$. In addition, since the servers move in the network according to a fixed schedule, it is assumed that one may not overtake the other. The result is that if a server, k, visits a queue, i, that is servicing packets using server j, then the transmission process will be interrupted. Consequently, server j forwards itself to the successor queue, $i+1$, and server k thereby starts a new transmission cycle. The analytical model is assumed to be symmetrical. In this case we assume that all servers are identical; travel at the same speed, follow the same route and carry the same load. The entire network is considered to be operated under steady state.

5.2.1 Data rate QoS incorporation

The proposed analytical model incorporates data rate guarantee quality of service. A description of the service is shown below.

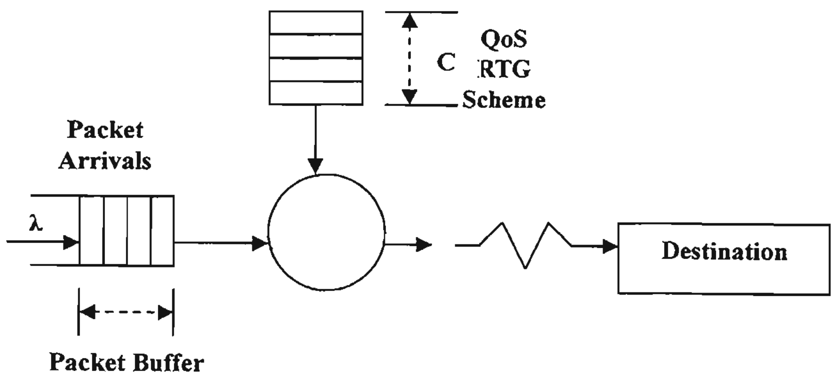


Figure 5.2: Queue model of the proposed system

Figure 5.2 displays the queue model for the proposed system. The system was perceived as a randomly timed gated, multiple server, multiple queue network. [Yechiali, 1998]

In Yechiali's approach, a single server network that incorporates a RTG scheme was analyzed. The proposed model is different from his proposal, as a multiple server network is analysed instead of a single server network.

To access the channel, the queue needs to capture one of the servers that are circulating in the network. If the server has been captured and there is more than one data packet in the queue, the RTG scheme is then activated to provide a constraint parameter, *C*, where C is an exponentially distributed random variable with mean θ. θ is set to be proportional to the packet arrival rate, λ. The queue then empties a maximum of *C* packets in order to satisfy the data rate QoS. The purpose of the constraint parameter is to model the permit buffer effect, as proposed in Chapter 3. The parameter *C* is used to provide fairness and dynamic service discipline into the model, thereby ensuring that all queues receive sufficient access to the network.

5.3 Mean Packet Delay Analysis

Numerous papers have attempted to derive the expression for the mean packet transfer delay for multiple server networks, for example [Qing, 1989], [Hamacher, 1989], [Borst, 1997], [Kamal, 1994], [Kuhl, 1989], and [Wang, 1984]. However, the delay expressions derived are mainly used for networks that implement exhaustive, gated, or 1-limited service policies.

Using a similar approach as [Wang, 1984] for determining the sojourn time for the *k*-limited service policy, the analysis begins by focusing on one of the queues, queue *i* in this case. The mean sojourn time ($\overline{S_i}$) of a tagged packet arriving at queue *i* consists of two components. The first component is the mean waiting time ($\overline{W}$), this is the duration between the packet's arrival time and the time the same packet is removed from the queue by a server. The second component is the total mean service time of the packets (including itself), which are picked up together with the tagged packet. Thus,

$$\overline{S_i} = \overline{W} + \frac{\overline{B^2}}{\overline{B}} \cdot T_p \tag{5.1}$$

where the expression for $\overline{W}$, also from [Wang, 1984], is equal to,

$$\overline{W} = \frac{\overline{\alpha^2}}{2 \cdot \overline{\alpha}} + \frac{1}{\lambda_i} \cdot \overline{X} \tag{5.2}$$

The first term in (5.2) represents the time from when a packet arrives at queue i, to the time the next server arrives at queue i. This is also known as residual token inter-arrival time. The second term in (5.2) is, by Little's law ($\overline{X} = \lambda_i \cdot \overline{W}$), the mean time from the first service at queue i witnessed by a packet, until that packet is removed. The second term in (5.1) gives the product of the mean service time (T_p) of a packet and the number of packets in a batch removed by a server ($\frac{\overline{B^2}}{\overline{B}}$), where B is defined as the number of packets receiving service in a service epoch.

In order to compute the mean packet delay, it is imperative to derive the expressions for all the variables (X, B, and α) given in (5.1) and (5.2). Subsequently section 5.4 and 5.5 present the detailed derivation processes for each variable.

5.4 Approximate Mean Queue Length Analysis

The proposed model is built on a multiple server system, with the packets arriving at the queue according to a Poisson process, and the servicing policy is based on a RTG service time generation scheme. Using Kendall notation [Schwartz, 1987], each queue in the network may be modelled as an M/G/C/RTG-type queue. This type of system has been analysed in works such as [Kleinrock, 1975] and [Gross and Harris, 1985]. For the analysis, the arrival process and the service duration are assumed to be mutually independent, and the system is analysed in steady state.

The model is solved by finding the stationary distribution of the imbedded Markov chains. Firstly, cycles are indexed as service epochs at a queue with an integer n, $1 \leq n < \infty$. A service epoch cycle starts when a server visits a queue and ends when the next server visits the same queue. The activity of the queue between service cycles can be model as,

$$X_n = X_{n-1} - B_{n-1} + A_{n-1} \tag{5.3}$$

$$= \begin{cases} X_{n-1} - B_{n-1} + A_{n-1}\ ;\ B_{n-1} = X_{n-1},\ X_{n-1} \leq \frac{\alpha}{s} \leq C_i \ \text{ or } \ X_{n-1} \leq C_i \leq \frac{\alpha}{s} \\ X_{n-1} - B_{n-1} + A_{n-1}\ ;\ B_{n-1} = C_i,\ C_i < X_{n-1} < \frac{\alpha}{s} \ \text{ or } \ C_i < \frac{\alpha}{s} \leq X_{n-1} \\ X_{n-1} - B_{n-1} + A_{n-1}\ ;\ B_{n-1} = \frac{\alpha}{s},\ \frac{\alpha}{s} < X_{n-1} < C_i \ \text{ or } \ \frac{\alpha}{s} < C_i \leq X_{n-1} \end{cases}$$

Where,

X_n number of packets in queue i just before the *nth* service cycle

B_{n-1} number of packets in queue i that received service in the *(n-1)th* service cycle

C_i number of packets that can be served by queue i at the *(n-1)th* cycle

A_{n-1} number of packet arrivals to queue i between two consecutive *(n-1)*th and *(n)*th service cycles.

α server inter-arrival time

s packet length (fixed)

With the properties of the discrete time functions, as shown in (5.3), it is assumed that the number of packets in queue i, just before the *nth* service cycle (X_n), consist of two components. The first component, B_{n-1}, represents the number of packets that received service in *(n-1)th* service cycle. Secondly, A_{n-1}, are the number of packets that arrived between the *(n-1)th* and *nth* service cycles.

As displayed in equation (5.3), the number of packets that may receive service (B) is determined by three factors. They are the queue length (X), the QoS constraint (C) and the maximum number of serviceable packets during the token interarrival time ($\frac{\alpha}{s}$). In the first circumstance, the packets in the queue before service cycle (X_n) are less than the QoS constraint parameter C_i and maximum serviceable packets $\frac{\alpha}{s}$, therefore the queue is emptied i.e. $B_{n-1} = X_{n-1}$. For the second circumstance, $B_{n-1} = C_i$, which occurs when X_{n-1} meets the QoS constraint ($X_{n-1} > C_i$), in order to satisfy the QoS guarantee, only C_i packets may receive service. For the last circumstance, $B_{n-1} = \alpha/s$, both C_i and X_n are greater than $\frac{\alpha}{s}$. Consequently only $\frac{\alpha}{s}$ amount of packets may receive service.

Since packet arrivals (A) are Poissonian, we assume that this random variable, A, depends only on server interarrival time (α). It therefore does not depend on the queue or on the time of service initiation, with the result that

$$\begin{aligned}\Pr\{A=a\} &= \int_0^\infty \Pr\{A=a \mid \alpha=t\}\, dF_\alpha(t) \\ &= \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^a}{a!}\, dF_\alpha(t)\end{aligned} \tag{5.4}$$

where $F_\alpha(\cdot)$ denotes the cumulative probability distribution (CDF) function of the server inter-arrival time α. Next, we define the steady state probability of queue i being in state j as π_j, and assume that the state probabilities are identical at the beginning of each interval. Note that the j represents the queue size in queue i at the *(n-1)th* service cycle ($X_{n-1}^i = j$). Figure 5.3 illustrates the steady state assumption where j ranges from 0 to infinity.

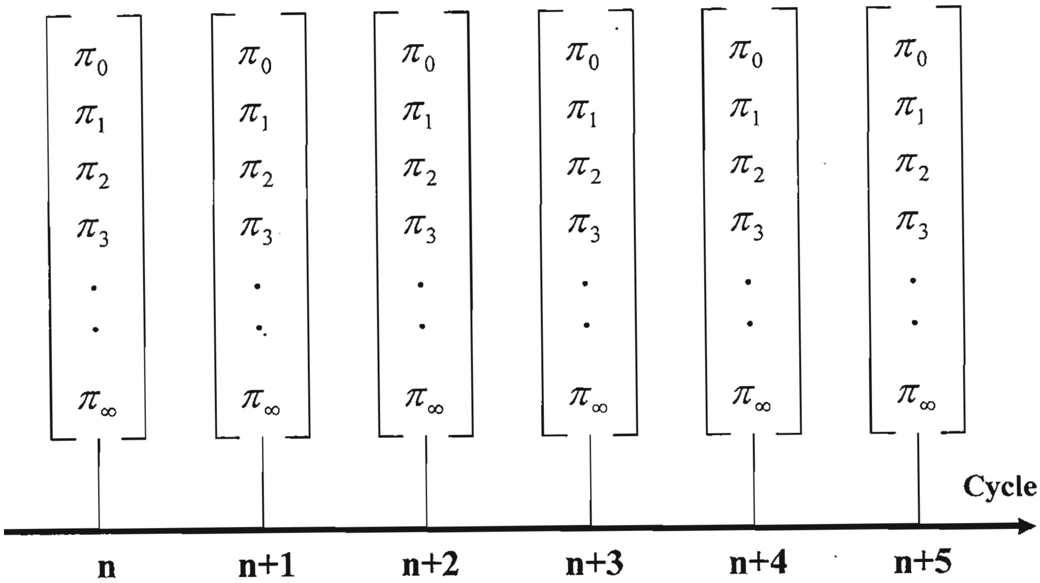


Figure 5.3: Steady state probabilities

We then define the steady state probability vector, $\prod$ as:

$$\Theta\Pi = \Pi \tag{5.5}$$

where $\prod$ is given by:

$$\Pi = \left[\pi_0, \pi_0, \pi_0, ...\pi_j, ...\pi_\infty\right] \tag{5.6}$$

The variable Θ is a state transition probability matrix; it is imperative that the expression for Θ is defined in order to solve the stationary equation (5.5). Therefore p_{jk} is denoted as the probability that the queue size is k immediately before the *nth* service cycle, given that the queue size was j before the *(n-1)th* service cycle. From the queuing model, no constraint is set for the queue capacity, therefore any j and k may therefore rise to infinity. The transition probability matrix Θ is therefore an infinite matrix with elements $k = 0,1,.....\infty$ and $j = 0,1,....\ \infty$. In order to solve the system state, an analytical expression for p_{jk} is needed.

5.4.1 Derivation of p_{jk}

p_{jk} is solved using Markov Chains and we re-write equation (5.5) into the following standard equation.

$$(\Theta - I)\Pi = 0 \tag{5.7}$$

where I is the identity matrix. We now represent a state change given that j is the queue length at the start of the *nth* service cycle and k is the queue length at the start of the *(n+1)th* service cycle. From equation (5.3), we may then say,

$$\begin{aligned} p_{jk} &= \Pr\{X_n = k | X_{n-1} = j\} \\ &= \Pr\{k = j + A - B\} \\ &= \Pr\{A = k - j + B\} \end{aligned} \tag{5.8}$$

As defined earlier, A represents the number of arrivals during the service cycle and B the number of packets that received service during the service cycle. We then solve for p_{jk}, the probability that the state jumps from j to k using the following two scenarios.

Scenario 1: Initial State $j \leq$ Final State k

$$p_{jk} = \Pr\{A = k - j + B\} \tag{5.9}$$

$$= \begin{cases} \Pr\{A = k\} \ldots\ldots\ldots\ldots\ldots\ldots j \leq \frac{\alpha}{s} \leq C_i \quad or \quad j \leq C_i \leq \frac{\alpha}{s} \\ \Pr\{A = k - j + C_i\} \ldots\ldots C_i < j < \frac{\alpha}{s} \quad or \quad C_i < \frac{\alpha}{s} \leq j \\ \Pr\left\{A = k - j + \frac{\alpha}{s}\right\} \ldots\ldots \frac{\alpha}{s} < C_i \leq j \quad or \quad \frac{\alpha}{s} < j < C_i \end{cases}$$

Consider the first scenario where initial state $j \leq$ final state k. There are three possible ways for state j to jump to k during the service cycle. In the first case, there would be a jump from state j to k with k arrivals and j services. The boundary condition for this case is that state j must be smaller than constraints, C_i and α/s. In the second case, state j jumps to state k with $(k\text{-}j+C_i)$ arrivals and C_i services, with the boundary condition that C_i must be smaller than both j and α/s. In the last case, there are $(k\text{-}j+\alpha/s)$ arrivals and α/s services during the state jump from j to k, with the boundary condition that α/s must smaller than both j and C_i.

Scenario 2: Initial State j > Final State k

$$p_{jk} = \Pr\{A = k - j + B\} \tag{5.10}$$

$$= \begin{cases} \Pr\{A=k\}..................j \le \frac{\alpha}{s} \le C_i \;\; or \;\; j \le C_i \le \frac{\alpha}{s} \\ \Pr\{A=k-j+C_i\}......C_i < j < \frac{\alpha}{s} \;\; or \;\; C_i < \frac{\alpha}{s} \le j \;\; and \;\; C_i \ge j-k \\ \Pr\left\{A=k-j+\frac{\alpha}{s}\right\}......\frac{\alpha}{s} < C_i \le j \;\; or \;\; \frac{\alpha}{s} < j < C_i \;\; and \;\; \frac{\alpha}{s} \ge j-k \end{cases}$$

The second scenario considers a situation where the initial state j is higher than the final state k. There are also three possible state jumps from j to k. However, the boundary conditions for the second and third case are different from the first scenario. In both cases, an additional boundary condition has been added, for which the services (C_i, α/s) must be bigger than the $(j\text{-}k)$ to avoid negative arrivals.

5.4.2 Average expressions of p_{jk}

To compute equations (5.9) and (5.10), an average expression is derived for both of them using switching functions [Davio, 1978]. With the help of a switching function, the boundary condition may clearly be separated. Switching functions $G_1\left(\frac{\alpha}{s}-C\right)$ and

$G_2\left(C-\frac{\alpha}{s}\right)$ are implemented, where G_1 is used for boundary condition $j \le C_i \le \frac{\alpha}{s}$ and G_2 is used for boundary condition $j \le \frac{\alpha}{s} \le C_i$. Equations (5.11) and (5.12) are then derived for the first case of equation (5.9).

$$G_1\left(\frac{\alpha}{s}-C\right)=\begin{Bmatrix} 1 & if & \frac{\alpha}{s}>C \\ \frac{1}{2} & if & \frac{\alpha}{s}=C \\ 0 & if & \frac{\alpha}{s}<C \end{Bmatrix} \tag{5.11}$$

and

$$G_2\left(C-\frac{\alpha}{s}\right)=\begin{Bmatrix} 1 & if & C>\frac{\alpha}{s} \\ \frac{1}{2} & if & \frac{\alpha}{s}=C \\ 0 & if & C<\frac{\alpha}{s} \end{Bmatrix} \tag{5.12}$$

The average expression for first case ($\Pr\{A=k\}$) of (5.9) can now be defined as,

$$\begin{aligned} Case\ 1 = & \sum_{\frac{\alpha}{s}\ge C}^{\infty}\sum_{C\ge j}^{\infty} G_1\left(\frac{\alpha}{s}-C\right)\cdot P_A(k)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\ & \sum_{C\ge\frac{\alpha}{s}}^{\infty}\sum_{\frac{\alpha}{s}\ge j}^{\infty} G_2\left(C-\frac{\alpha}{s}\right)\cdot P_A(k)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right) \end{aligned} \tag{5.13}$$

$P_A(k)$, $P_c(C)$ and $P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)$ are the probability distribution functions (pdf) used for the computation. $P_A(k)$ generates the probability that k packets arrive in the queue during a service cycle. As discussed in equation (5.4), $P_A(k)$ is a Poisson distributed probability distribution function since the arrivals are assumed to be Poisson random variables.

$P_c(C)$ is the pdf used for the QoS guarantee. Since C is defined as an exponential random variable, $P_c(C)$ is an exponential pdf. As mentioned earlier, a token inter-arrival time, α, is assumed to be a random variable with gamma distribution, and s is a constant. Therefore α/s is also a gamma distributed random variable and $P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)$ is a Gamma pdf.

Switching functions (5.14) and (5.15) are implemented for the second case ($\Pr\{A = k - j + C_i\}$) in equation (5.9). G_3 is implemented for the boundary condition $C_i < \frac{\alpha}{s} \le j$ and G_4 for $C_i < j < \frac{\alpha}{s}$, where G_3 and G_4 are given by

$$G_3\left(j-\frac{\alpha}{s}\right)=\begin{Bmatrix} 1 & if & j>\frac{\alpha}{s} \\ \frac{1}{2} & if & \frac{\alpha}{s}=j \\ 0 & if & j<\frac{\alpha}{s} \end{Bmatrix} \tag{5.14}$$

and,

$$G_4\left(\frac{\alpha}{s}-j\right)=\begin{Bmatrix} 1 & if & \frac{\alpha}{s}>j \\ \frac{1}{2} & if & \frac{\alpha}{s}=j \\ 0 & if & \frac{\alpha}{s}<j \end{Bmatrix} \tag{5.15}$$

The average expression for the second case ($\Pr\{A=k-j+C_i\}$) of (5.9) is then defined as,

$$Case\ 2=\sum_{C=0}^{j-1}\sum_{\frac{\alpha}{s}\geq j}^{\infty}G_4\left(\frac{\alpha}{s}-j\right)\cdot P_A\left(k-j+C\right)\cdot P_C\left(C\right)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+$$
$$\sum_{\frac{\alpha}{s}=0}^{j}\sum_{C=0}^{\frac{\alpha}{s}-1}G_3\left(j-\frac{\alpha}{s}\right)\cdot P_A\left(k-j+C\right)\cdot P_C\left(C\right)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right) \tag{5.16}$$

Switching functions G_5 and G_6 are introduced for the last case of (5.9) i.e.($\Pr\left\{A=k-j+\frac{\alpha}{s}\right\}$), where G_5 is for the boundary condition $\frac{\alpha}{s}<C_i\leq j$ and G_6 for $\frac{\alpha}{s}<j<C_i$,

$$G_5\left(j-C\right)=\begin{Bmatrix}1 & if\ j>C\\ \frac{1}{2} & if\ C=j\\ 0 & if\ j<C\end{Bmatrix} \tag{5.17}$$

and,

$$G_6\left(C-j\right)=\begin{Bmatrix}1 & if\ C>j\\ \frac{1}{2} & if\ C=j\\ 0 & if\ C<j\end{Bmatrix} \tag{5.18}$$

Lastly, the average expression for the third case of (5.9) is defined,

$$Case\ 3=\sum_{C=0}^{j}\sum_{\frac{\alpha}{s}=0}^{c-1}G_5\left(j-C\right)\cdot P_A\left(k-j+\frac{\alpha}{s}\right)\cdot P_C\left(C\right)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+$$
$$\sum_{\frac{\alpha}{s}=0}^{j-1}\sum_{C\geq j}^{\infty}G_6\left(C-j\right)\cdot P_A\left(k-j+\frac{\alpha}{s}\right)\cdot P_C\left(C\right)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right) \tag{5.19}$$

By deriving the average expressions for three different cases, the average expression for scenario 1 may now be computed,

$$
\begin{aligned}
p_{jk} &= Case\ 1 + Case\ 2 + Case\ 3 \\
&= \sum_{\frac{\alpha}{s}\geq C}^{\infty}\sum_{C\geq j}^{\infty} G_1\left(\frac{\alpha}{s}-C\right)\cdot P_A(k)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
&\sum_{C\geq\frac{\alpha}{s}}^{\infty}\sum_{\frac{\alpha}{s}\geq j}^{\infty} G_2\left(C-\frac{\alpha}{s}\right)\cdot P_A(k)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
&\sum_{C=0}^{j-1}\sum_{\frac{\alpha}{s}\geq j}^{\infty} G_4\left(\frac{\alpha}{s}-j\right)\cdot P_A(k-j+C)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
&\sum_{\frac{\alpha}{s}=0}^{j}\sum_{C=0}^{\frac{\alpha}{s}-1} G_3\left(j-\frac{\alpha}{s}\right)\cdot P_A(k-j+C)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
&\sum_{C=0}^{j}\sum_{\frac{\alpha}{s}=0}^{c-1} G_5(j-C)\cdot P_A\left(k-j+\frac{\alpha}{s}\right)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
&\sum_{\frac{\alpha}{s}=0}^{j-1}\sum_{C\geq j}^{\infty} G_6(C-j)\cdot P_A\left(k-j+\frac{\alpha}{s}\right)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)
\end{aligned}
\qquad (5.20)
$$

For scenario 2, where initial state, j, is greater than final state, k, the same method is applied to attain the average expression. In addition, two special conditions must be taken into consideration. Firstly, in the second case of (5.10), the QoS constraint, C, must be greater than $(j-k)$ and secondly, the maximum serviceable packets during the token inter-arrival cycle, $\frac{\alpha}{s}$, must also be greater than $(j-k)$. This is to avoid negative arrivals. This additional boundary condition affects the expressions by changing the lower limits of the average expressions for cases 2 and 3. The average expression for scenario 2 is derived as shown in (5.21).

$$\begin{aligned}
p_{jk} = & \sum_{\frac{\alpha}{s}\geq C}^{\infty}\sum_{C\geq j}^{\infty} G_1\left(\frac{\alpha}{s}-C\right)\cdot P_A(k)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
& \sum_{C\geq\frac{\alpha}{s}}^{\infty}\sum_{\frac{\alpha}{s}\geq j}^{\infty} G_2\left(C-\frac{\alpha}{s}\right)\cdot P_A(k)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
& \sum_{C\geq(j-k)}^{j-1}\sum_{\frac{\alpha}{s}\geq j}^{\infty} G_4\left(\frac{\alpha}{s}-j\right)\cdot P_A(k-j+C)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
& \sum_{\frac{\alpha}{s}=0}^{j}\sum_{C\geq(j-k)}^{\frac{\alpha}{s}-1} G_3\left(j-\frac{\alpha}{s}\right)\cdot P_A(k-j+C)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
& \sum_{C=0}^{j}\sum_{\frac{\alpha}{s}\geq(j-k)}^{c-1} G_5(j-C)\cdot P_A\left(k-j+\frac{\alpha}{s}\right)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)+ \\
& \sum_{\frac{\alpha}{s}\geq(j-k)}^{j-1}\sum_{C\geq j}^{\infty} G_6(C-j)\cdot P_A\left(k-j+\frac{\alpha}{s}\right)\cdot P_C(C)\cdot P_{\frac{\alpha}{s}}\left(\frac{\alpha}{s}\right)
\end{aligned} \tag{5.21}$$

Focus is now placed on obtaining an expression for the steady state probability vector, Π. Since there may theoretically be an infinite number of packet arrivals during a service cycle interval, π_k is simply expressed as:

$$\pi_k = \sum_{j=0}^{\infty} \pi_j \cdot p_{jk} \tag{5.22}$$

Equation (5.22) is then solved subject to the following steady state condition

$$\sum_{j=0}^{\infty} \pi_j = 1 \tag{5.23}$$

The pdf of the queue length (X) is then defined as

The pdf of the queue length (X) is then defined as

$$\Pr\{X=b\}=\Pi_b=\sum_{j=0}^{\infty}\pi_j\cdot p_{jb} \tag{5.24}$$

Equation (5.24) is then solved to obtain the pdf of the queue length subject to the conditions in (5.23). Once the pdf of the queue length is obtained, the mean queue length can be derived using statistic theory since,

$$E[X]=\sum_{x=0}^{\infty}x\cdot\Pr\{X=x\}=\sum_{x=0}^{\infty}x\cdot\sum_{j=0}^{\infty}\pi_j p_{jx} \tag{5.25}$$

5.5 Derivation for Token Rotation Time (*TRT*), Interarrival Time (α)

A similar approach to [Wang, 1984] was used to derive the expressions for token rotation time and interarrival time. However, [Wang, 1984] only derived the expressions for multiple server networks that implemented gated and k-limited service disciplines. His approach is therefore modified and an expression derived for our model. Note that for the derivation, each server is visualized as a single token, since the definition for the server and token are identical.

5.5.1 Mean token rotation time ($\overline{TRT}$)

The random variable token rotation time, $TRT_{i,j}$, is defined as the time between successive visits of sever j to queue i. The random variable token interarrival time, α_i, is also defined as the time between successive visits by any token to queue i.

To obtain the mean of the token rotation time, $TRT_{i,j}$, it is first assumed that the model is server-symmetric. h is defined to be the random variable denoting the total walk time of a token in a token rotation cycle. The proportion of time that server j spends walking on the first n cycles is then

$$\frac{\frac{1}{n}\sum_{k=1}^{n} h^k}{\frac{1}{n}\sum_{k=1}^{n} TRT_{i,j}^k} \tag{5.26}$$

The superscript k denotes a value that occurs on the *k-th* token rotation cycle of a server. As $n \to \infty$, equation (5.26) then tends to a long-run limit of mean value $\frac{\overline{h}}{\overline{TRT_{i,j}}}$. Next it is assumed that the rate of completion of work is equal to the service rate of work in the system, which is the ratio of packet servicing time of all N queues ($\sum_{i=1}^{N} M \cdot \overline{B_i} \cdot T_p$) over the mean token rotation time ($\overline{TRT_{i,j}}$). Note that $\overline{B_i}$ is the mean number of packets serviced from queue i during the token interarrival time and T_p is the service time of a single packet. With the previously defined assumption, which states that each of the M tokens do not overtake one another, the expression $M \cdot \overline{B_i} \cdot T_p$ gives the total number of packets serviced from queue i during the token rotation cycle.

From the assumption of server-symmetry, it may be said that server j completes a $\frac{1}{M}$ portion of the work. By conservation of work we get that

$$1 - \frac{\overline{h}}{\overline{TRT_{i,j}}} = \frac{1}{M} \frac{\sum_{i=1}^{N} M \cdot \overline{B_i} \cdot T_p}{\overline{TRT_{i,j}}} \tag{5.27}$$

And with simple manipulation,

$$\overline{TRT} = \frac{\sum_{i=1}^{N} M \cdot \overline{B_i} \cdot T_p}{M} + \overline{h} \tag{5.28}$$

where $\overline{TRT_{i,j}}$ is now written as $\overline{TRT}$, since it does not depend on j or i, and $\overline{h} = \sum_{i=1}^{N} \overline{h_i}$.

5.5.2 Mean token interarrival time ($\overline{\alpha}$)

The mean token interarrival time ($\overline{\alpha}$), is assumed to be the average time that each server spends in the token rotation cycle, and is therefore defined from [Wang, 1984] as,

$$\overline{\alpha} = \frac{\overline{TRT}}{M} \tag{5.29}$$

From equation (5.28), and with the assumption that the mean number of packets ($\overline{B}$) that received service in a cycle for each queue is the same, the general expression for mean TRT is derived,

$$\overline{TRT} \approx \frac{N \cdot M \cdot T_p \cdot \overline{B}}{M} + \overline{h} \tag{5.30}$$

The mean token inter-arrival time is dependent on B, and therefore,

$$\begin{aligned} \overline{\alpha} = \frac{\overline{TRT}}{M} &\approx \frac{N \cdot M \cdot T_p \cdot \overline{B}}{M^2} + \frac{\overline{h}}{M} \\ &\approx \frac{N \cdot T_p \cdot \overline{B}}{M} + \frac{\overline{h}}{M} \end{aligned} \tag{5.31}$$

5.5.3 Variance of token interarrival time (Var[α])

From equations of (5.28) and (5.29), the general expression of α can be derived as,

$$\begin{aligned}\alpha = \frac{TRT}{M} &= \frac{\sum_{i=1}^{N} M \cdot B_i \cdot T_p}{M^2} + \frac{h}{M} \\ &= a \cdot \sum_{i=1}^{N} B_i + c\end{aligned} \tag{5.32}$$

where $a = \frac{T_p}{M}$ and $c = \frac{h}{M}$

From general statistics, the variance of α is equal to

$$Var[\alpha] = E\left[\alpha^2\right] - E[\alpha]^2 \tag{5.33}$$

To derive the relationship between the token interarrival time and the serviced packet, (5.33) has to be expanded, where the 2nd moment of α is now equalled to,

$$\begin{aligned}E\left[\alpha^2\right] &= E\left[\left(a \cdot \sum_{i=1}^{N} B_i + c\right)^2\right] \\ &= E\left[a^2 \cdot \sum_i \sum_j B_i B_j + 2ac \sum_i B_i + c^2\right] \\ &= E\left[a^2 \cdot \sum_i \sum_j B_i B_j\right] + 2acN \cdot E[B] + c^2 \\ &= \sum_{i=1}^{N} a^2 E\left[B_i^2\right] + \sum_i \sum_{j \neq i} a^2 E[B_i] E\left[B_j\right] \\ &\quad + 2acN \cdot E[B] + c^2 \\ &= \sum_{i=1}^{N} a^2 E\left[B_i^2\right] + a^2 N(N-1)(E[B])^2 \\ &\quad + 2acN \cdot E[B] + c^2\end{aligned} \tag{5.34}$$

and,

$$\begin{aligned}E[\alpha]^2 &= \left(N \cdot a \cdot E[B] + c\right)^2 \\ &= a^2 N^2 E[B]^2 + 2acN \cdot E[B] + c^2\end{aligned} \tag{5.35}$$

Now the variance of α can be computed as,

$$
\begin{aligned}
Var[\alpha] &= E[\alpha^2] - E[\alpha]^2 \\
&= E\left[a^2 \cdot \sum_i \sum_j B_i B_j\right] + 2acN \cdot E[B] + c^2 - a^2N^2E[B]^2 - 2acN \cdot E[B] - c^2 \\
&= E\left[a^2 \cdot \sum_i \sum_j B_i B_j\right] - a^2N^2E[B]^2 \\
&= \sum_{i=1}^{N} a^2 E[B_i^2] + \sum_i \sum_{j \neq i} a^2 E[B_i]E[B_j] - a^2N^2(E[B])^2 \\
&= \sum_{i=1}^{N} a^2 E[B_i^2] + a^2N(N-1)(E[B])^2 - a^2N^2(E[B])^2 \\
&= \sum_{i=1}^{N} a^2 E[B_i^2] + a^2NE[B]^2[(N-1) - N] \\
&= Na^2E[B^2] - Na^2E[B]^2 \\
&= Na^2(E[B^2] - E[B]^2) \\
&= Na^2Var[B]
\end{aligned}
\tag{5.36}
$$

Based on [Wang, 1984], the token interarrival time (α) is assumed to be a gamma distributed random variable with mean and coefficient of variation of $\frac{\overline{TRT}}{M}$ and $\sqrt{\frac{M-1}{M+1}}$ respectively, i.e. $\gamma \cdot \beta = \frac{\overline{TRT}}{M}$ and $\sqrt{\frac{1}{\gamma}} = \sqrt{\frac{M-1}{M+1}}$. With the parameters γ and β derived, the $Var[\alpha]$ can therefore be solved, i.e. $Var[\alpha] = \gamma \cdot \beta^2 = \frac{M-1}{M+1}\left(\frac{\overline{TRT}}{M}\right)^2$.

5.5.4 Mean and variance of serviced packets (*B*)

The mean value of service packet B can be derived from equation (5.3), where

$$X_n = X_{n-1} - B_{n-1} + A_{n-1} \tag{5.37}$$

Using statistical theory, the expected value for equation (5.37) is derived as,

$$E[X_n] = E[X_{n-1}] - E[B_{n-1}] + E[A_{n-1}] \tag{5.38}$$

Assuming the system is in steady state, the queue lengths between two adjacent service cycles are the same,

$$E[X_n] = E[X_{n-1}] \tag{5.39}$$

By substituting (5.39) into (5.38), it may then be shown that the mean service packet is equal to the mean packet arrivals under steady state conditions,

$$E[B] = E[A] = \lambda \tag{5.40}$$

To find the second moment of B, equation (5.37) is first squared,

$$\begin{aligned}(X_n)^2 &= (X_{n-1} - B_{n-1} + A_{n-1})^2 \\ &= (X_{n-1})^2 - 2(X_{n-1} \cdot B_{n-1}) + 2(X_{n-1} \cdot A_{n-1}) \\ &\quad -2(A_{n-1} \cdot B_{n-1}) + (A_{n-1})^2 + (B_{n-1})^2\end{aligned} \tag{5.41}$$

Next, the expected value for (5.41) is found,

$$\begin{aligned}E[X_n^2] = E[X_{n-1}^2] - 2E[X_{n-1} \cdot B_{n-1}] + 2E[X_{n-1}]E[A_{n-1}] \\ -2E[A_{n-1}]E[B_{n-1}] + E[A_{n-1}^2] + E[B_{n-1}^2]\end{aligned} \tag{5.42}$$

It is known from (5.37) that the number of arrivals *(A_{n-1})* during the *(n-1)th* service cycle is independent of either the service packet during cycle *(B_{n-1})* or the queue length in cycle *(X_{n-1})*. Since they are independent of each other, we may separate their expectation as shown in the 3rd and 4th terms of (5.42). From (5.39) and (5.40), It is also known that $E[X_{n-1}] = E[X_n]$ and $E[B] = E[A] = \lambda$, where the subscript n has been ignored as we assume the system is under steady state. (5.42) may then be evaluated as

$$E[B^2] = 2\lambda \cdot \lambda + 2E[XB] - 2\lambda E[X] - (\lambda + \lambda^2) \tag{5.43}$$

To solve for $E[XB]$, it is assumed that X and B are correlated with each other with the correlation variable ρ i.e. $E[XB] = \rho E[X]E[B]$. Now the second moment of B is equal to,

$$E\left[B^2\right] = \lambda^2 - \lambda + 2\rho \cdot \lambda E[X] - 2\lambda E[X] \tag{5.44}$$

It is known from (5.25) that the mean queue length is equal to

$$E[X] = \sum_{x=0}^{\infty} x \cdot \Pr\{X = x\} = \sum_{x=0}^{\infty} x \cdot \sum_{j=0}^{\infty} \pi_j p_{jx} \tag{5.45}$$

Therefore the second moment of B may then be derived as

$$E\left[B^2\right] = \lambda^2 - \lambda + 2\lambda(\rho - 1)\sum_{x=0}^{\infty} x \cdot \sum_{j=0}^{\infty} \pi_j p_{jx} \tag{5.46}$$

Variance of B is then derived from statistical theory,

$$\begin{aligned} Var[B] &= E\left[B^2\right] - E[B]^2 \\ &= \lambda^2 - \lambda + 2\rho \cdot \lambda E[X] - 2\lambda E[X] - \lambda^2 \\ &= -\lambda + 2\lambda E[X](\rho - 1) \\ &= -\lambda + 2\lambda\left(\sum_{x=0}^{\infty} x \cdot \sum_{j=0}^{\infty} \pi_j p_{jx}\right)(\rho - 1) \end{aligned} \tag{5.47}$$

The correlation variable ρ is therefore solved by substituting (5.36) into (5.47). The equations needed to obtain the mean packet delay in the system are shown below.

$$\pi_k = \sum_{j=0}^{\infty} \pi_j \cdot p_{jk} \tag{5.22}$$

$$\sum_{j=0}^{\infty} \pi_j = 1 \tag{5.23}$$

$$\overline{X} = E[X] = \sum_{x=0}^{\infty} x \cdot \Pr\{X = x\} = \sum_{x=0}^{\infty} x \cdot \sum_{j=0}^{\infty} \pi_j p_{jx} \tag{5.25}$$

$$\overline{\alpha} = \frac{\overline{TRT}}{M} \tag{5.29}$$

$$\begin{aligned}\overline{\alpha^2} = E\left[\alpha^2\right] = \sum_{i=1}^{N} a^2 E\left[B_i^2\right] + a^2 N(N-1)\left(E[B]\right)^2 \\ + 2acN \cdot E[B] + c^2\end{aligned} \tag{5.34}$$

$$\overline{B} = E[B] = E[A] = \lambda \tag{5.40}$$

$$\overline{B^2} = E\left[B^2\right] = \lambda^2 - \lambda + 2\lambda(\rho - 1)\sum_{x=0}^{\infty} x \cdot \sum_{j=0}^{\infty} \pi_j p_{jx} \tag{5.46}$$

5.6 Computation Parameters and Results

To verify the analysis, the analytical model is computed and compared with the computer simulations. Two examples are considered. In the first example, a 20-queue network is simulated. The packet length is fixed at 160 bits and there are 8 tokens (16 bits in length) circulated in the network. Each code channel has a capacity of 128kbits/s. Figure 5.4 shows the mean packet transfer delay for this network.

Figure 5.5 shows the second example, where a 30-queue network was simulated. The packet length is fixed at 160 bits and 8 tokens with a token length of 16 bits. The results in both cases clearly show that the analysis closely agrees with the simulation results, with some deviation at high loads. The deviation is caused by simplifying assumptions.

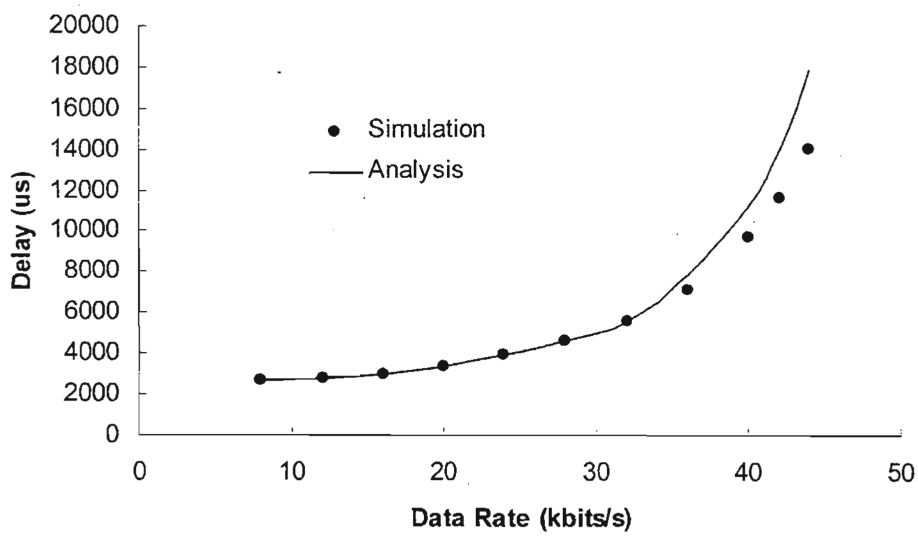


Figure 5.4: Comparison of approximate packet delay analysis to simulation, $N = 20, M = 8$

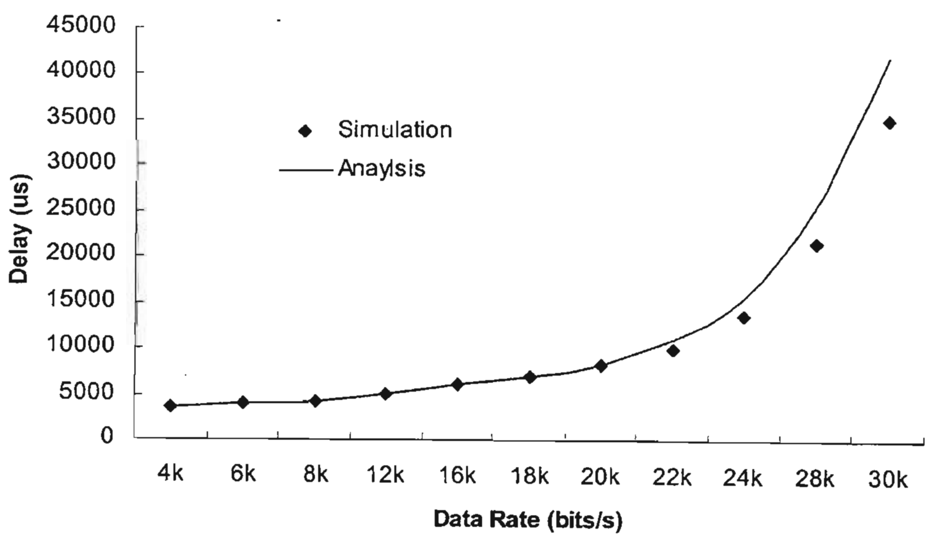


Figure 5.5: Comparison of approximate packet delay analysis to simulation, $N = 30, M = 8$

5.7 Summary

In chapter five, an analytical model for the MAC scheme proposed in chapter three was presented. Since code-sharing behaviour in the proposed CDMA MAC scheme resembled the token-sharing behaviour in the multiple token scheme, the proposed analytical model is built based on a multiple token scheme. An approximate delay analysis of a multiple token network that incorporated a data rate QoS guarantee was then presented.

The analysis started with the derivation of the expression for the queuing delay that a packet suffered whilst waiting to be serviced. The same approach that is used by [Wang, 1984] was used. The equation for packet delay consisted of two parts; the duration that the tagged packet waited to be service and the duration needed to serviced all the packets that queued before the tagged packet. The analysis continued by deriving the mean value of the queue length. As discussed previously, the model is a multiple token system, and the packets arrived at the queue according to a Poisson process. Each queue in the network is modelled as an M/G/C/RTG-type queue. The data rate QoS guaranteed is provided by implementing a randomly timed gated scheme as proposed by [Yechiali, 1998]. The model is then solved by finding the stationary distribution of the imbedded Markov chains and the expressions were derived for the mean queue length.

The expressions for token rotation time and token interarrival time were then derived using a similar approach to that proposed by [Wang, 1984]. The analysis started by defining the random variable token rotation time as the time between successive visits of a server to a queue. The analysis continued by finding the relationship between token rotation time and token interarrival time. The serviced packets within a service cycle were then derived as the serviced packets correlated to the token interarrival time. The analysis concluded by deriving the relationship between the serviced packet and the token interarrival time. From the analysis conducted, the complexities involved in analysing multiple token network protocols can clearly be seen, even under steady state conditions. The main contribution of chapter five consists of the derivations of the mean queue length and the mean and variance of serviced packets of the proposed analytical model. Since the queue length and the serviced packets constitute the major portion of the formation of mean packet transfer delay, two examples were given that validated the analysis. It was observed that the analytical results closely agreed with the simulation results under light-to-medium load conditions. However, discrepancies can be seen under high load conditions caused by the simplifying assumptions.

CHAPTER 6

CONCLUSION

This dissertation began by introducing the structure of the wireless network, followed by a description of the concepts of both centralized and distributed wireless networks. Chapter two reviewed relevant existing literature on wireless MAC protocols. Firstly, the protocols were classified into three different categories, based on the network topology they were implemented on.

The first category examined the protocols that are used in distributed ad hoc wireless networks. Ad hoc MAC protocols use an ad hoc topology, in which each device in the network has the same functionality and is free to manoeuvre in the network. Collision avoidance algorithms are comprehensively used in these types of networks. The focus of this dissertation is on distributed wireless networks, and hence more time was spent investigating the various protocols that were proposed.

The second category discussed was the protocols implemented in centralized networks. Centralized MAC protocols use a centralized topology in which a base station controls and organizes the transmission between the mobile devices. This type of topology allows a highly optimized medium access control, as both the hidden and exposed node problems do not exist.

Next, the protocols proposed, the two topologies discussed above are combined; known as hybrid networks was examined. Normally, a network is denoted as an ad hoc network when it is formed without any central administration and it consists of mobile nodes that use a wireless interface to send packet data. However, in an hybrid network, there is a centralized administrator (e.g. a mobile BS) in the network.

In chapter three, building upon a token passing strategy, we introduced the new concept of data transmission assignment. This concept allows the token to be constantly circulating in the distributed network, with transmission being granted by distributing CDMA codes to the mobile device, eliminating the problem of hidden and exposed terminals. Based on this concept, the token passing based code assignment protocol was developed. With the increasing popularity of multimedia applications, quality of service (QoS) is an important part of MAC protocol design, therefore two types of quality of service guarantees were incorporated into the proposed MAC scheme. They are data rate and delay level QoS guarantees. Data rate QoS is implemented to ensure fairness, and delay level QoS is used to provide a priority mechanism. The proposed protocol was then evaluated through event driven computer simulations. The results obtained suggest that the protocol can provide priority access and at the same time maintain fairness to all traffic classes.

Chapter four goes a step further; implementing the proposed MAC protocol in the wireless Bluetooth piconet network. This chapter began by providing a comprehensive overview of the Bluetooth network and its topology. The approach used to adapt the proposed MAC scheme into a Bluetooth piconet system was then discussed. In this scenario, due to the wireless nature and the non-uniform manoeuvrability of the nodes, the issues of token or nodes dropping out the network were taken into consideration. The additions of the token and node list to the token structure were introduced to assist in tracking down the active nodes in the network. The network also reinitialized after a pre-assigned number of token cycles, in order to allow new nodes to join the network.

The simulation model and parameters were then presented, and the model was simulated in a Bluetooth piconet environment using an event driven program. The results have shown that the proposed scheme remains stable even under the influence of the wireless aspect. Delay performance remained at optimal level at a light to medium load and deviated exponentially at heavy loads. The throughput performance, however, was not optimized. As predicted, when nodes drop out the network, packets are also lost thereby decreasing the throughput. By increasing the dropout probability, it was also observed that the throughput was significantly lower on all the nodes, with a corresponding reduction in packet delay at heavy loads for all traffic classes. This is as expected, since the packets generated from the newly joined node have a shorter arrival time, and therefore the delay for packet delivery time is shortened.

In chapter five, some approximating methods for analyzing the proposed MAC scheme were presented with an approximate mean value analysis incorporating data rate QoS. In the analysis, a network with a single token with multiple codes was considered as a multiple token network. Two examples were given that validated the analysis, although the analysis presented is approximate and the simplifying assumptions cause deviations from the simulation results at heavy loads. It may, however, give insight into a more accurate approximate analysis.

During the course of this research, there were issues that had the potential to further improve the overall performance of the system, but have fallen outside the research scope. One of them is taking of the packet transmission into consideration. In one of our assumptions, it was assumed that all packet transmissions are interference free; therefore it would be valuable to investigate the effect of introducing a packet loss model and monitoring the packet transmission activity. Secondly, although the proposed MAC scheme was investigated in a wireless Bluetooth piconet environment, it would be worth discovering how the scheme would behave in the more complicated scatternet system.

A protocol based on token passing code assignment has the potential of providing a flexible and intelligent accessing control to the wireless channel in distributed networks. Although the proposed token passing MAC protocol shows reasonable performance, it could be further improved by incorporating several other factors that have been described earlier. This is left for further research.

APPENDIX A

HIDDEN AND EXPOSED TERMINALS

A.1 Hidden terminal

The problem caused by the hidden terminal [Tobagi, 1975] constitutes one of the hidden sender and the hidden receiver problems. When a node wants to transmit to another adjacent node that is currently receiving data, this is defined as hidden sender problem. This problem is illustrated in Figure A.1, node A transmits a packet to node B, and node C as the hidden sender, also wants to transmit to node B. However, node C does not hear the ongoing transmission from node A; a collision will be resulted if node C transmits to node B.

To eliminate the hidden sender problem, a control handshake mechanism is commonly implemented. Using the example mentioned earlier, as node A transmits to node B, node B will then broadcast a signal (or packet) before or while it is receiving a transmission from node A. This mechanism is useful to avoid the hidden sender problem, however, control handshakes can generate a hidden receiver problem. Node B uses a control handshake to warn other surrounding nodes that it is receiving a transmission, node C hears the control handshake and defers its transmission. At the same instance, if node D wishes to transmit data to node C and node D is too far away from node B. Consequently, node D does not hear the control handshake generated by node B. Node D then transmits a packet to node C. Node C receives the packet successfully, but cannot acknowledge node D because transmitting an acknowledgement packet would result a collision in node B. Node C is a hidden receiver.

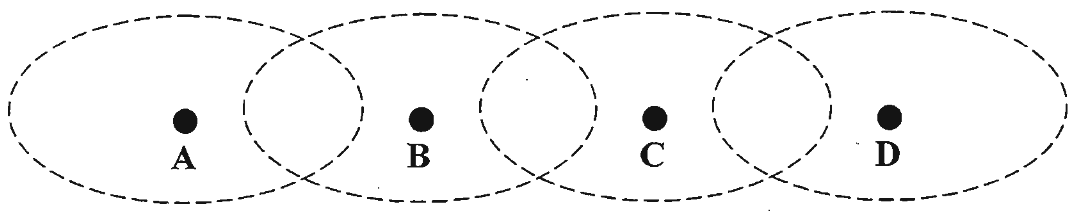


Figure A.1 Example of hidden and exposed terminal problems

A.2 Exposed terminal

The exposed node problem [Tobagi, 1975] is similar to the hidden node problem in that it also consists of two problems, the exposed sender and the exposed receiver problems respectively. The exposed sender problem occurs when a node that has data packets to transmit is exposed by an ongoing transmission. As illustrated in Figure A.1, node C is transmitting a packet to node D. Node B has a packet for node A but cannot initiate a transmission to node A as the transmission from node B has an opportunity to collide with the transmission from node C. Node B is therefore an exposed sender.

For the exposed receiver problem, as shown in Figure A.1, if node C transmits a packet to node D and node A has a packet destined for node B. A collision will occur if node A transmits its packet to node B. The collision is caused by the transmission from node C colliding with the transmission from node A. Node B is therefore an exposed receiver.

REFERENCES

Acampora, AS and Krishnamurthy, SV (1997). "A New Adaptive MAC Layer Protocol for Wireless ATM Networks in Harsh Fading and Interference Environments," *Proc. ICUPC '97*, vol. 2, Oct. 1997, pp. 410-415.

Alasti, M and Farvardin, N (1999). "D-PRMA: A Dynamic Packet Reservation Multiple Access Protocol for Wireless Communications," *Proceedings of the 2 nd ACM international workshop on modeling analysis and simulation of wireless and mobile systems*, 1999, pp. 41-49.

Amitay, N (1992). "Resource Auction Multiple Access (RAMA): Efficient Method for Fast Resource Assignment for Decentralized Wireless PCS," *Electronic Letters*, vol. 28, 1992, pp. 799-801.

Amitay, N (1993). "Distributed Switching and Control with Fast Resource Assignment/Handoff for Personal Communication Systems," *IEEE JSAC*, SAC-11, 1993, pp. 842-49.

ANSI Standard X3T9.5 (1987). Fiber Distributed Data Interface (FDDI) – Token Ring Medium Access Control (MAC), May, 1987.

Borst, SC and Van, DM (1997). "Analysis of Multiple-Server Polling Systems by Means of the Power-Series Algorithm," *Stochastic Models,* vol. 13, 1997, pp 339-369.

Chandraa, A, Gummalla, V, and Limb, JO (2000). "Wireless Medium Access Control Protocols," *IEEE Communications Surveys*, Second Quarter, 2000.

Chen, KC and Lee, CH (1993). "RAP – A Novel Medium Access Protocol for Wireless Data Networks," *Proc. IEEE GLOBECOM,* 1993, pp. 1713-1717.

Chlamtac, I, Myers A, Syrotiuk, V, and Zaruba, G (2000). "An Adaptive Medium Access Control (MAC) Protocol for Reliable Broadcast in Wireless Networks," *IEEE International Conference on Communications*, 2000, vol. 3, 1692-1696.

Colombo, G, Lenzini, L, Mingozzi, E, Cornaglia, B, and Santaniello R (1999). "Performance Evaluation of PRADOS: A Scheduling Algorithm for Traffic Integration in a Wireless ATM Network," *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking* (MobiCom'99), Aug. 1999, pp. 143-150.

Crow, B. et al (1997). "IEEE 802.11: Wireless Local Area Networks," *IEEE Commun. Mag.*, vol. 35, no. 9, Sept. 1997, pp. 116-126.

Davio, M, Deschamps, JP and Thayse, A (1978). *Discrete and Switching Functions*, McGraw-Hill International 1978.

De Moraes, LFM and Rubin, I (1984). "Analysis and comparison of message queueing delays in token-rings and token-buses local area networks," in *Proc. IEEE Int. Conf. Commun.*, May. 1984, pp 130-135.

Deng, J and Hass, ZJ (1998). "Dual busy tone multiple access (DBTMA): A new medium access control for packet radio networks," in *Proc. Int. Conf. Universal Personal Commun.*, Florence, Italy, Oct. 5, 1998.

Everitt, D (1986). "Simple Approximations for Token Rings," *IEEE Trans. On Commun*, vol. com-34, no. 7, July 1986, pp 719-721.

Fishcher, W and Hellstern, KM (1992). "The Markov- modulated Poisson Process (MMPP) cookbook", *Performance Evaluation*, vol.18, 1992, pp 149-171.

Fullmer, CL and Garcia-Luna-Aceves, JJ (1995). "Floor Acquisition Multiple Access (FAMA) for Packet-radio Networks," *Proc. ACM SIGCOMM 95,* 1995.

Fullmer, CL and Garcia-Luna-Aceves JJ (1999). "Floor Acquistion Multiple access (FAMA) in Single-channel Wireless Networks," *Mobile Networks and Applications,* vol. 4, 1999, pp. 157-174.

Goodman, DJ *et al.* (1989). "Packet Reservation Multiple Access for Local Wireless Communications," *IEEE Trans. Commun.*, vol. 37, no. 8, July 1989, pp. 885-890.

Goodman, DJ and Wong WC (1993). "Integrated Data and Speech Transmission using Packet Reservation Multiple Access," *Proc. IEEE ICC '93,* May 1993, pp. 172-176.

Gross, D and Harris, CM (1985). *Fundamentals of Queuing Theory*, Second Edition, John Wiley and Sons 1985.

Gummalla, A and Limb, J (2000). "Wireless Collision Detection (WCD): Multiple Access with Receiver Initiated Feedback and Carrier Detect Signal," *IEEE International Conference on Communications*, vol. 1, 2000, pp. 397-401

Haartsen, J (2000). "The Bluetooth Radio System," *IEEE Personal Communications,* Feb. 2000, pp. 28-36.

Haines, RJ and Aghvami, AH (1994). "Indoor Radio Environment Considerations in Selecting a Media Access Control Protocol for Wideband Radio Data Communications," *Proc. IEEE ICC'94*, vol. 3, 1994, pp. 1306-1311.

Hamacher, VC and Kamal, EA (1989). "Approximate analysis of non-exhaustive multiserver polling systems with applications to LAN," *Computer Networks and ISDN Systems*, vol 17, 1989, pp 15-27.

Huo, TC and Tsai, TJ (2001). "An access-based clustering protocol for multihop wireless Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 7, 2001, pp 1201-1210.

IEEE Standards 802.5 (1989). Token Ring Access Method and Physical Layer Specifications. The Institute of Electrical and Electronic Engineers, Inc., 1989.

ITU Telecommunication Indicator Update (2001). July-August-Sepetmeber-October, 2001.

Kalia, M, Bansal, D and Shorey, R (2000). "Data scheduling and SAR for Bluetooth MAC," *In Proceedings VTC2000-Spring IEEE 51st Vehicular Technology Conference*, vol. 2, Tokyo, Japan, May, 2000, pp 716-720,.

Kamal, EA (1994). "The multi-token ring network protocol," *Computer Networks and ISDN Systems*, vol 26, 1994, pp.1477-1494.

Karn, P (1990). "MACA – A New Channel Access Method for Packet Radio," *ARRL/CRRL Amateur Radio 9 th Computer Networking Conference*, 1990, pp. 134-140.

Karol, MJ, Liu, Z and Eng, KY (1995). "An Efficient Demand-Assignment Multiple Access Protocol for Wireless Packet (ATM) Networks," *Wireless Networks, ACM-Press*, Baltzer Science Publishers, vol. 1.3, 1995, pp. 267-279.

Kleinrock, L (1975). *Queuing Systems Volume I: Theory,* John wiley and Sons, Wiley-Interscience Publication 1975.

Kleinrock, L and Tobagi, FA (1975). "Packet switching in radio channels: Part II – the hidden terminal problem in carrier sense multiple-access modes and the busy-tone solution," *IEEE Trans. Commun.*, vol. 4, COM-23, no. 12, 1975, pp. 1417-1433.

Kuhl, JG and Karmarkar, VV (1989). "An integrated to distributed demand assignment in multiple bus local networks," *IEEE Trans. On Computers*, vol. 38, no.5, MAY 1989.

Lee, D, Attias, R, Puri, A, Sengupta, R, Tripakis, S and Varaiya P (2001). "A wireless token ring protocol for intelligent transportation systems," *IEEE Intelligent Transportation Systems Conference Proceedings*, Aug. 2001, pp. 1152-1157.

Lenzini, L, Luise, M and Reggiannini R (2001). "CRDA: A Collision Resolution and Dynamic Allocation MAC Protocol to Integrate Data and Voice in Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 6, 2001, pp. 1153-1163.

Lihui, DG, Ly, H, Hong, X, Gerla, M, Pei, M and Lee YZ (2000). "C-ICAMA, a centralized intelligent channel assigned multiple access for multi-layer ad-hoc wireless networks with UAVs," *WCNC 2000 - IEEE Wireless Communications and Networking Conference*, no. 1, Sept. 2000, pp. 879 – 884.

Mikkonen, J, Aldis, J, Awater, G, Lunn, A and Hutchison D (1998). "The Magic WAND – Functional Overview," *IEEE Journal on Selected Areas in Communications*, v16, no. 6, Aug. 1998, pp. 953-972.

Pinheiro, ALA and De Marca JRB (1996). "Fair Deterministic Packet Access Protocol: F-RAMA," *Electronic Letters*, vol. 32, no. 25, Dec. 5, 1996.

Prasad R and Ojanpera T. (1998). "An Overview of CDMA Evolution toward Wideband CDMA," *IEEE Communications Surveys*, vol. 1, No. 1, Fourth Quarter, 1998.

Pursley, MB (1987). "The role of spread-spectrum in packet radio networks," *Proc. IEEE, Vol. 75, No. 1,* Jan. 1987, pp. 116-143.

Qing, Y and Bhuyan, LN (1989). "Approximate analysis of single and multiple ring networks," *IEEE Trans. On Computers*, vol. 38, no.7, July 1989.

Rubin, I and DeMoraes, LFM (1983). "Message delay analysis for polling and token multiple-access schemes for local computer networks," *IEEE J. Select. Areas Commun.*, vol. SAC-1, Nov. 1983, pp. 935-947.

Saadawi, TN, Ammar, MH and Hakeem AE (1994). *Fundamentals of Telecommunication Networks*. A Wiley-Interscience Publication, 1994

Schwartz, M (1987). *Telecommunication Networks: Protocols, Modeling and Analysis.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1987.

Sethi, AS and Saydam, T (1985). "Performance Analysis of Token Ring Local Area Networks," *Computer Networks and ISDN Systems*, no 9, 1985, pp. 191-200.

Shieh, CK, Hwang, WS and Ho, JH (2000). "A new polling architecture for network with non-exhaustive multiple token mechanism," *IEEE Proceedings of ICS '00*, Taiwan, Dec, 2000, pp. 46-50.

Spragins, JD (1991). *Telecommunications Protocols and Design*, Addison-Wesley 1991.

Takagi, H (1986). *Analysis of Polling Systems*, MIT Press, Cambridge, MA, 1986.

Tang, K and Gerla, M (2000). "Random Access MAC for Efficient Broadcast Support in Ad Hoc Networks," *IEEE Wireless Communications and Networking Conference*, vol. 1, 2000, pp. 454-459.

Tang, Z and Garcia-Luna-Aceves, JJ (1999). "A Protocol for Topology-dependent Transmission Scheduling in Wireless Networks," *Proceedings IEEE WCNC*, 1999.

Tang, Z and Garcia-Luna-Aceves, JJ (2000). "Collision-Avoidance Transmission Scheduling for Ad-Hoc Networks," *IEEE International Conference on Communications*, 2000, vol. 3, 2000, pp. 1788-1794.

Tobagi, FA and Kleeinrock, L (1975). "Packet switching in radio channels: Part II – the hidden terminal problem in carrier sense multiple-access modes and the busy-tone solution," *IEEE Trans. Commun.*, vol. COM-23, no. 12, 1975, pp. 1417-1433.

Vaidya, N, Holland, G, and Bahl, P (2000). "A rate-adaptive MAC protocol for wireless networks," *Technical Report TR00-019, Texas A&M University,* August, 2000.

Wang, YT and Morris, RJT (1984). "Some Results For Multi-Queue Systems With Multiple Cyclic Servers," *Perform. Computer-Communication Systems*. Amsterdam, The Netherlands. Elsevier Science Publisher B. V., North Holland, IFIP, 1984, pp. 245-258.

Wang, YT and Morris, RJT (1985). "Load sharing in distributed systems," *IEEE Trans. Comput.*, vol. C-34, Mar. 1985, pp. 204-217.

Wu, C and Li, V (1988). "Receiver- initiated Busy-tone Multiple Access in Packet Radio Networks," *Proceedings of the ACM Workshop on Frontiers in Computer Communications Technology*, 1988, pp. 336-342.

Wu, G, Mukumoto, K and Fukuda, A (1993). "An Integrated Voice and Data Transmission System with Idle Signal Multiple Access–Dynamic Analysis," *ICICE Trans. Communication*. vol. E76-B, no. 11, Nov., 1993.

Wu, G et al (1996). "An R-ISMA Integrated Voice/Data Wireless Information System with Different Packet Generation Rates," *Proceedings of IEEE ICC '96*, vol. 3, 1996, pp. 1263-1269.

Wu, S, Lin, C, Tseng, Y and Sheu, J (2000). "A New Multi-channel MAC Protocol with On-demand Channel Assignment for Multi- hop Mobile Ad Hoc Networks," *International Symposium on Parallel Architectures, Algorithms and Networks*, 2000, pp. 232 –237.

Wong, T, Mark, J and Chua, K (2000). "Access and Control in a Cellular Wireless ATM Network," *IEEE International Conference on Communications*, 2000, vol. 3, pp. 1524 –1529.

Yechiali, U and Eliazar, I (1998). "Randomly Timed Gated Queuing Systems," *SIAM J. Appl. Math*, vol. 59, no.2, 1998, pp 423-441.

Zhang, Z and Acampora, AS (1991). "Performance of a Modified Polling Strategy for Broadband Wireless Lans in a Harsh Fading Environment," *Proc. IEEE GLOBECOM '91*, 1991, pp. 114-146.