
Electronic Theses and Dissertations, 2004-2019

2010

A New Framework For Qos Provisioning In Wireless Lans Using The P-persistent Mac Protocol

Kiran Babu Anna
University of Central Florida

 Part of the [Computer Sciences Commons](#), and the [Engineering Commons](#)
Find similar works at: <https://stars.library.ucf.edu/etd>
University of Central Florida Libraries <http://library.ucf.edu>

This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2004-2019 by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation

Anna, Kiran Babu, "A New Framework For Qos Provisioning In Wireless Lans Using The P-persistent Mac Protocol" (2010). *Electronic Theses and Dissertations, 2004-2019*. 4293.
<https://stars.library.ucf.edu/etd/4293>

A NEW FRAMEWORK FOR QOS PROVISIONING IN WIRELESS
LANS USING THE P-PERSISTENT MAC PROTOCOL

by

KIRAN BABU ANNA

M.S. University of Central Florida, 2003

M.Sc.(Hons.) Birla Institute of Technology and Science, Pilani, India, 1998

B.E.(Hons.) Birla Institute of Technology and Science, Pilani, India, 1998

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the School of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2010

Major Professor: Mostafa A. Bassiouni

© 2010 KIRAN BABU ANNA

ABSTRACT

The support of multimedia traffic over IEEE 802.11 wireless local area networks (WLANs) has recently received considerable attention. This dissertation has proposed a new framework that provides efficient channel access, service differentiation and statistical QoS guarantees in the enhanced distributed channel access (EDCA) protocol of IEEE 802.11e.

In the first part of the dissertation, the new framework to provide QoS support in IEEE 802.11e is presented. The framework uses three independent components, namely, a core MAC layer, a scheduler, and an admission control. The core MAC layer concentrates on the channel access mechanism to improve the overall system efficiency. The scheduler provides service differentiation according to the weights assigned to each Access Category (AC). The admission control provides statistical QoS guarantees. The core MAC layer developed in this dissertation employs a P-Persistent based MAC protocol. A weight-based fair scheduler to obtain throughput service differentiation at each node has been used.

In wireless LANs (WLANs), the MAC protocol is the main element that determines the efficiency of sharing the limited communication bandwidth of the wireless channel. In the second part of the dissertation, analytical Markov chain models for the P-Persistent 802.11 MAC protocol

under unsaturated load conditions with heterogeneous loads are developed. The Markov models provide closed-form formulas for calculating the packet service time, the packet end-to-end delay, and the channel capacity in the unsaturated load conditions. The accuracy of the models has been validated by extensive NS2 simulation tests and the models are shown to give accurate results.

In the final part of the dissertation, the admission control mechanism is developed and evaluated. The analytical model for P-Persistent 802.11 is used to develop a measurement-assisted model-based admission control. The proposed admission control mechanism uses delay as an admission criterion. Both distributed and centralized admission control schemes are developed and the performance results show that both schemes perform very efficiently in providing the QoS guarantees. Since the distributed admission scheme control does not have a complete state information of the WLAN, its performance is generally inferior to the centralized admission control scheme.

The detailed performance results using the NS2 simulator have demonstrated the effectiveness of the proposed framework. Compared to 802.11e EDCA, the scheduler consistently achieved the desired throughput differentiation and easy tuning. The core MAC layer achieved better delays in terms of channel access, average packet service time and end-to-end delay. It also achieved higher system throughput than EDCA for any given service differentiation ratio. The admission control provided the desired statistical QoS guarantees.

To my Mom and Dad.

ACKNOWLEDGMENTS

I just can't describe in words how much I am indebted to my advisor, Dr. Mostafa A. Bassiouni for his role from the beginning to the end of this dissertation work. I wouldn't have completed this work without his guidance, advice, and constant words of encouragement during the difficult times in my research. Above all, he has been a source of inspiration for his work ethic, principles and for being a wonderful role model.

I also like to take this opportunity to thank Dr. Damla Turgut for sharing her research experiences from her PhD program and also, for constantly encouraging me with my research during our conversations in the corridors of the computer science building. I sincerely appreciate my committee members Dr. Ronald D. Dutton and Dr. Ratan K. Guha for their valuable feedback and comments.

A special mention needs to go to my parents. I wouldn't have been the person who I am without their sacrifices, and support. I can't thank them enough for believing in me and having immense patience with me. I am also grateful to my brother and sister for being there for me at all times.

My research experience would not be complete without my lab-mates. They not only made my research memorable by sharing wonderful times but also putting up with my idiosyncrasies. I

thank Ms. Jarawan Mesit for helping me with the diagrams in my dissertation. I must thank Ms. Shafaq Chaudhary for those wonderful conversations about everything in life.

I owe a lot to my friends Honey Dandwani, Kirsten Brinkman, and Sameer Joshi for all the fun and adventures. The occasional get togethers you guys organized kept me sane and provided me with a sense of balance in my life.

And last but not least, my life would be bland and incomplete without all the people from the past and present. You helped me to grow as a person by providing rich experiences in my life and I sincerely thank all of you.

TABLE OF CONTENTS

LIST OF FIGURES	xii
LIST OF TABLES	xv
CHAPTER 1 INTRODUCTION	1
1.1 Wireless LANs	1
1.2 Quality of Service (QoS) in Wireless LANs	2
1.3 Motivation for a New QoS Framework	3
1.4 Analytical Modeling of QoS in WLANs	4
1.5 Organization of the Dissertation	4
CHAPTER 2 802.11 MAC PROTOCOL	6
2.1 The IEEE 802.11 MAC Protocol	6
2.2 The Distributed Coordination Function (DCF)	7
2.3 IEEE 802.11e Standard for QoS	9

2.4	P-Persistent 802.11	12
CHAPTER 3 NEW FRAMEWORK FOR QoS PROVISIONING IN 802.11 WLANs .		15
3.1	Introduction	15
3.2	Related Work	18
3.3	The Proposed QoS Framework	20
3.3.1	The Core MAC Layer	21
3.3.2	The Scheduler	23
3.3.3	Admission Control	24
3.4	Results and Analysis	25
3.4.1	Simulation Setup	26
3.4.2	Throughput Differentiation	28
3.4.3	System Throughput	33
3.4.4	Packet Delay	36
CHAPTER 4 ANALYTICAL MODEL FOR P-PERSISTENT 802.11		41
4.1	Introduction	41
4.2	Related Work	43
4.3	Markov Chain	46

4.3.1	No Retry Limit	47
4.3.2	Finite Retry Limit	50
4.4	Service Time Delay	54
4.4.1	Equal Load in Unsaturated Load Conditions	56
4.4.2	Saturated Load Conditions	57
4.4.3	No Retry Limit	57
4.4.4	Finite Retry Limit	60
4.5	One hop End-to-End Delay	61
4.6	Throughput Analysis	64
4.7	Model Validation	68
4.7.1	Simulation Model	68
4.7.2	Correctness of the P-Persistent 802.11 MAC in NS2	69
4.7.3	Saturated Load	71
4.7.4	Unsaturated System with Equal Load	74
4.7.5	Unsaturated System with Unequal Load	76
 CHAPTER 5 ANALYSIS AND MODELING OF THE SATURATION THROUGH-		
PUT IN 802.11 WLANS		84

5.1	Introduction	84
5.2	Saturation Throughput	85
5.3	Results and Analysis	88
CHAPTER 6 ADMISSION CONTROL FOR P-PERSISTENT 802.11		94
6.1	Introduction	94
6.2	Related Work	97
6.3	Load Estimation	100
6.3.1	Heuristic based Load Estimation	100
6.3.2	Analytical based Load Estimation	103
6.4	Distributed Admission Control	104
6.5	Centralized Admission Control	108
6.6	Performance results	110
CHAPTER 7 CONCLUSIONS AND FUTURE WORK		124
7.1	Conclusions	124
7.2	Future Work	126
LIST OF REFERENCES		128

LIST OF FIGURES

2.1	HCF vs DCF.	10
2.2	Traffic Classes in EDCA.	11
2.3	Service Differentiation in EDCA.	12
2.4	P-Persistent 802.11 vs IEEE 802.11 DCF.	13
3.1	Framework for QoS Support.	20
3.2	AEDCF3A: fraction of the system throughput.	32
3.3	AEDCF3: fraction of the system throughput.	33
3.4	SPDCF3: fraction of the system throughput.	34
3.5	System throughput.	35
3.6	Channel access delay.	37
3.7	Service time delay.	39
3.8	One hop e2e delay.	40
4.1	Markov chain model for the no retry limit case for node j.	48

4.2	Markov chain model for the finite retry limit case for node j.	52
4.3	Correctness of the P-Persistent 802.11 for variable packets under saturated load.	70
4.4	Correctness of the P-Persistent 802.11 for variable nodes under saturated load.	71
4.5	Validation of service time delay for variable nodes under saturated load.	72
4.6	Validation of end-to-end delay for variable nodes under saturated load.	73
4.7	Validation of service time delay for variable packets under saturated load.	74
4.8	Validation of end-to-end delay for variable packets under saturated load.	75
4.9	Validation of service time delay in an unsaturated system with eqload for P= 300.	76
4.10	Validation of end-to-end delay in an unsaturated system with eqload for P= 300.	77
4.11	Validation of service time delay in an unsaturated system with eqload for P= 600.	78
4.12	Validation of end-to-end delay in an unsaturated system with eqload for P= 600.	79
4.13	Validation of service time delay in an unsaturated system with uneqload for P= 300.	80
4.14	Validation of end-to-end delay in an unsaturated system with uneqload for P= 300.	81
4.15	Validation of service time delay in an unsaturated system with uneqload for P = 600.	82
4.16	Validation of end-to-end delay in an unsaturated system with uneqload for P = 600.	83
5.1	Throughput of P-Persistent 802.11.	87
5.2	System throughput for different queue sizes.	88

5.3	Contention Nodes P=300 bytes.	91
6.1	System throughput for different network sizes for P = 300 bytes.	101
6.2	System throughput for different network sizes for P = 600 bytes.	102
6.3	End-to-End Delay (N=10, P=300 bytes) Heuristic based Dist. Adm. Control.	112
6.4	End-to-End Delay (N=10, P=300 bytes) Analytical based Dist. Adm. Control.	113
6.5	End-to-End Delay (N=10, P=300 bytes) Centralized Admission Control.	114
6.6	End-to-End Delay (N=10, P=600 bytes) Heuristic based Dist. Adm. Control.	115
6.7	End-to-End Delay (N=10, P=600 bytes) Analytical based Dist. Adm. Control.	116
6.8	End-to-End Delay (N=10, P=600 bytes) Centralized Admission Control.	117
6.9	End-to-End Delay (N=10, P=300 bytes) Heuristic based Dist. Adm. Control.	118
6.10	End-to-End Delay (N=10, P=300 bytes) Analytical based Dist. Adm. Control.	119
6.11	End-to-End Delay (N=10, P=300 bytes) Centralized Admission Control.	120
6.12	End-to-End Delay (N=10, P=300 bytes) Heuristic based Dist. Adm. Control.	121
6.13	End-to-End Delay (N=10, P=300 bytes) Analytical based Dist. Adm. Control.	122
6.14	End-to-End Delay (N=10, P=300 bytes) Centralized Admission Control.	123

LIST OF TABLES

3.1	802.11 MAC Parameters	28
3.2	AEDCF3A MAC PARAMETERS FOR THE THREE ACs	29
3.3	AEDCF3 MAC PARAMETERS FOR THE THREE ACs (3:2:1)	31
4.1	802.11 MAC Parameters	69
5.1	802.11 MAC Parameters	90
6.1	802.11 MAC Parameters	111

CHAPTER 1 : INTRODUCTION

1.1 Wireless LANs

The development of wired networks brought about a new technology revolution in the form of Internet and along with it, a great number of diverse applications to the user. The user could now pay bills, make reservations online or meet with people located at different parts of the world from the comfort of his/her home or work place. At the same time, advancements in the radio technology and cheaper electronic circuitry have led to cheaper and smaller wireless devices. With the maturity of the wired networks and the availability of cheaper wireless devices, the focus has shifted from wired networks to wireless networks. The goal is to offer flexibility to the user in using the same set of diverse applications without the hassle of wires and fixed locations. This emphasis on wireless networks has led to numerous wireless technologies *viz.*, cellular networks, ad-hoc networks, sensor networks, and mesh networks, each with a specific use. Wireless Local Area Networks (WLANs) is one such technology which can be used in a limited coverage area like airports, bookstores, convention centers, hotels, etc. This is the beginning of the next step in personal computing. The user can carry his/her personal desktop to any location he/she wants. Thus, personal computing evolves into mobile personal computing.

A wireless LAN (WLAN) is similar to the Ethernet based LAN except that it is wireless. It is based on the carrier sense multiple access with collision avoidance (CSMA/CA). Unlike the Carrier Sense Multiple Access with Collision Detection (CSMA/CD) which deals with transmissions after a collision has occurred, CSMA/CA attempts to prevent collisions before they happen. The IEEE has standardized the CSMA/CA access mechanism for WLANs in the form of the IEEE 802.11 standard [IEE99]. The CSMA/CA 802.11 MAC protocol was originally designed to provide best effort service to data traffic like FTP, HTTP and other data base applications. The IEEE 802.11 protocol has one queue for the traffic generated in each wireless device and uses a set of four parameters to access the channel or wireless medium. In Chapter 2, we provide a detailed description of the IEEE 802.11 and its parameters.

1.2 Quality of Service (QoS) in Wireless LANs

With the increasing demand for QoS applications, there has been an increasing need to support QoS traffic in WLANs. The IEEE published another standard called IEEE 802.11e [IEE04] which has four queues, called Access Categories (ACs), to support various classes of traffic. Each AC is provided with a set of four parameters just like in the standard IEEE 802.11. Therefore, each AC competes with the other ACs in a wireless device (station or node) and also competes with the ACs of the other nodes in the wireless network. The other significant thing to note is that QoS support is provided only in terms of service differentiation among the ACs by prioritizing the channel access

parameters. Various research groups came up with different solutions for optimizing the channel access within the framework of the existing functionality [add references]. The main approach of these solutions is to combine QoS (service differentiation) and channel access functionalities together. The solutions were therefore not efficient enough to provide maximum channel utilization and the required QoS at the same time.

1.3 Motivation for a New QoS Framework

A true QoS support that provides delay and/or throughput guarantees cannot be achieved without controlling the admittance of the traffic. Therefore, an admission control is required to provide a true or an improved level of QoS support. In this dissertation, we propose a new framework to support efficient channel utilization, service differentiation and provide delay/throughput guarantees. Our motivation is guided by the rationale that a new design which provides simple and elegant solution is far much better than trying to add some patches or fixes to the existing 802.11e. Specifically, we propose and evaluate the separation of service differentiation and channel access components by means of a scheduler based architecture so that both the QoS differentiation and high throughput are achieved. In our framework, an admission control is added to provide QoS guarantees.

1.4 Analytical Modeling of QoS in WLANs

The literature on the analytical modeling and performance evaluation of 802.11 WLANs is rich and growing. A major focus was to model the capacity of 802.11 MAC and find the optimum parameter values for better performance. However, 802.11 MAC has too many parameters at the channel access level and analytical modeling of 802.11 becomes more complex with the addition of QoS features such as service differentiation.

In this dissertation, we use P-Persistent 802.11 as our access mechanism irrespective of the number of ACs in the node. The P-Persistent protocol employs only one parameter and is different from that of 802.11 MAC only in the back-off algorithm. We develop and validate analytical models to estimate the average service time delay, end-to-end delay, and throughput for the P-Persistent 802.11 MAC protocol. We use the analytical models in designing an efficient admission control for our QoS framework.

1.5 Organization of the Dissertation

Our dissertation is organized as follows. In Chapter 2, we discuss the functionalities of IEEE 802.11, IEEE 802.11e and P-Persistent 802.11 in detail. We present a scheduler-based architecture for QoS provisioning in 802.11 and evaluate its efficiency in Chapter 3. We discuss the advantages of separating the QoS and channel access components from each other and support our claims

with simulation results. In Chapter 4, we develop an analytical model for the P-Persistent 802.11 MAC protocol and derive expressions to estimate average service time delay, end-to-end delay and throughput. During our work, we came across an interesting observation that the assumptions (requirements) used in the literature to evaluate the saturation throughput in saturated load conditions are not completely true. We discuss this observation in Chapter 5. We present an admission control that works in both distributed and centralized scenarios in Chapter 6. We conclude the dissertation and discuss our future work in Chapter 7.

CHAPTER 2 : 802.11 MAC PROTOCOL

2.1 The IEEE 802.11 MAC Protocol

In WLANs, IEEE 802.11 has become the standard medium access control (MAC) and physical layer (PHY) specification [IEE99]. It provides two access methods: (i) the Distributed Coordination Function (DCF), and (ii) the Point Coordination Function (PCF). The DCF, also known as the basic access method, is a carrier sense multiple access with collision avoidance (CSMA/CA) protocol, whereas PCF is similar to a polling system and uses a centralized point coordinator that determines the next station that can transmit. The data traffic in DCF is transmitted on a first-come first-serve, best effort basis. All the data traffic has the same priority and the nodes in a basic service set (BSS) contend with the same priority.

PCF on the other hand is designed to support time bound traffic. It divides the time between the transmission of two consecutive Delivery Traffic Indication Message (DTIM) beacon frames into two periods viz., the contention period (CP) and the contention free period (CFP). The beacon frames carrying synchronization and the BSS information are sent periodically by the Access Point (AP). The beacon frames indicate the start of the CFP and its duration. The nodes access the channel using the DCF mechanism during the contention period and using the PCF mechanism

during the contention free period. Even though PCF was designed to support time bound traffic, unpredictable beacon delays leading to reduced CFP and unknown transmission durations of the polled stations made it difficult for the PCF to maintain the polling schedule of the nodes during the CFP. As a result, PCF has not been widely used in service differentiation [GHC07]. Since we are more interested in QoS in a distributed scenario, we only present the main features of DCF [IEE99].

2.2 The Distributed Coordination Function (DCF)

In DCF, a node with a frame to transmit senses the channel to determine whether another node is transmitting or not. If the channel is sensed idle for a specified time period equal to the Distributed Inter-Frame Space (DIFS), the node is allowed to transmit. If the channel is sensed busy either immediately or during the DIFS, the transmission is deferred until the node finds the channel to be idle for a DIFS. When the channel is observed to be idle for an amount of time equal to DIFS, the node generates a random back-off interval to minimize the probability of collision with frames from other nodes. This is because every node in the system observes the same channel behavior and if they transmit immediately after DIFS, there will be collisions all the time whenever there are more than two nodes contending for the channel. Therefore, the random back-off delay helps to reduce the number of collisions.

DCF adopts an exponential back-off scheme. The back-off time is uniformly chosen in the range $[0, CW-1]$, where CW is called the *contention window*. The CW depends on the number of failed transmission attempts for the frame. The CW is initialized to CW_{min} (Contention Window Minimum) at the first transmission attempt of a frame and is doubled up to a maximum value of CW_{max} (Contention Window Maximum) for each subsequent unsuccessful transmission attempt.

The back-off timer is decreased as long as the channel is sensed idle, stopped when a transmission is in progress, and reactivated when the channel is sensed idle again for more than DIFS. As soon as the back-off timer expires, the node starts to transmit at the beginning of the next slot time.

Since the CSMA/CA protocol does not rely on the capability of the nodes to detect a collision, a positive acknowledgement (ACK) is transmitted by the receiver node to signal the successful reception of the frame. The receiver sends an ACK frame after a time period equal to the Short Inter-Frame Space (SIFS), which is less than DIFS. If the transmitting station does not receive an ACK within a specified ACK Timeout, the data frame is presumed to be lost, and after extended inter-frame space (EIFS), a retransmission is scheduled. According to the standard, a maximum number of retransmissions (default=6) are allowed before the frame is dropped.

The two-way handshaking technique outlined above for frame transmission is called the basic access mechanism. In addition, DCF defines an optional four-way handshaking technique called request to send/clear to send (RTS/CTS) mechanism to solve the hidden terminal problem. In this mechanism, a node that wants to transmit a frame, waits until the channel is sensed idle for DIFS,

follows the back-off rules explained above, and then transmits a special short frame called Request To Send (RTS) instead of a frame. When the receiving node detects an RTS frame, it responds with a Clear To Send (CTS) frame after a SIFS. The transmitting station is allowed to transmit its frame only if the CTS frame is correctly received. The frames RTS and CTS carry the information of the length of the frame to be transmitted. This information can be read by any listening station, which is then able to perform virtual sensing through its Network Allocation vector (NAV) that contains information about the amount of time the channel will remain busy. The RTS/CTS mechanism is very effective in terms of system performance, especially when large frames are considered.

2.3 IEEE 802.11e Standard for QoS

In order to support QoS to different classes of traffic, the IEEE 802.11 standard has created working group E to come up with the enhancements to the IEEE 802.11 MAC Protocol. The draft standard for IEEE 802.11e defines a new coordination function called hybrid coordination function (HCF). The HCF consists of two channel access methods called "Enhanced Distributed Channel Access" (EDCA) mechanism and "HCF Controlled Channel Access" (HCCA) mechanism. The EDCA, an extension of DCF, is a contention based access mechanism. The HCCA is a controlled access mechanism which works in both the CP and CFP. From now onwards, we will discuss EDCA just like DCF in 802.11 as it works in the distributed or contention based scenario. Fig. 2.1 shows the difference between HCF and DCF. One of the significant extensions to IEEE 802.11 in 802.11e is

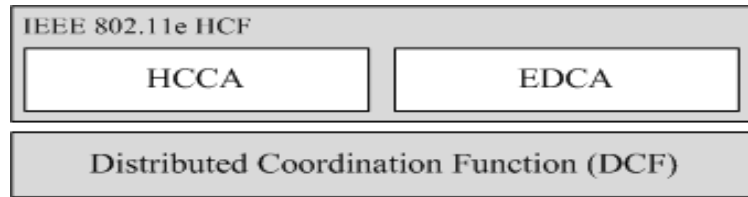


Figure 2.1: HCF vs DCF.

the support for different types of traffic identified by different priorities. A total of eight priority levels are available and are mapped to four access categories (ACs). Each AC, represented by a queue in the 802.11e MAC, contends for the channel with the help of several parameters *viz.*, contention window minimum (CW_{min}), contention window maximum (CW_{max}), inter-frame space and transmission opportunity. A single AC can be thought of as a single DCF and therefore contends internally with other ACs in a node and externally with all the ACs in other nodes (see Fig. 2.2).

The access to the channel is prioritized among the ACs by using different values for the same parameter in each AC. For example, the inter-frame space called arbitration inter-frame space (AIFS) functions the same way as DIFS except that each AC has different AIFS value and therefore each AC has to wait for a different AIFS duration of idle time. Thus the EDCA supports service differentiation among different ACs by using different parameter values (see Fig. 2.3).

When two ACs in a node decide to send packets at the same time, after following the channel access mechanism as described in DCF, an internal collision occurs. The internal collision is resolved by transmitting the packet from the higher priority AC and executing a queue-specific back-off procedure on the packet from the lower priority AC as if it had incurred a collision. In

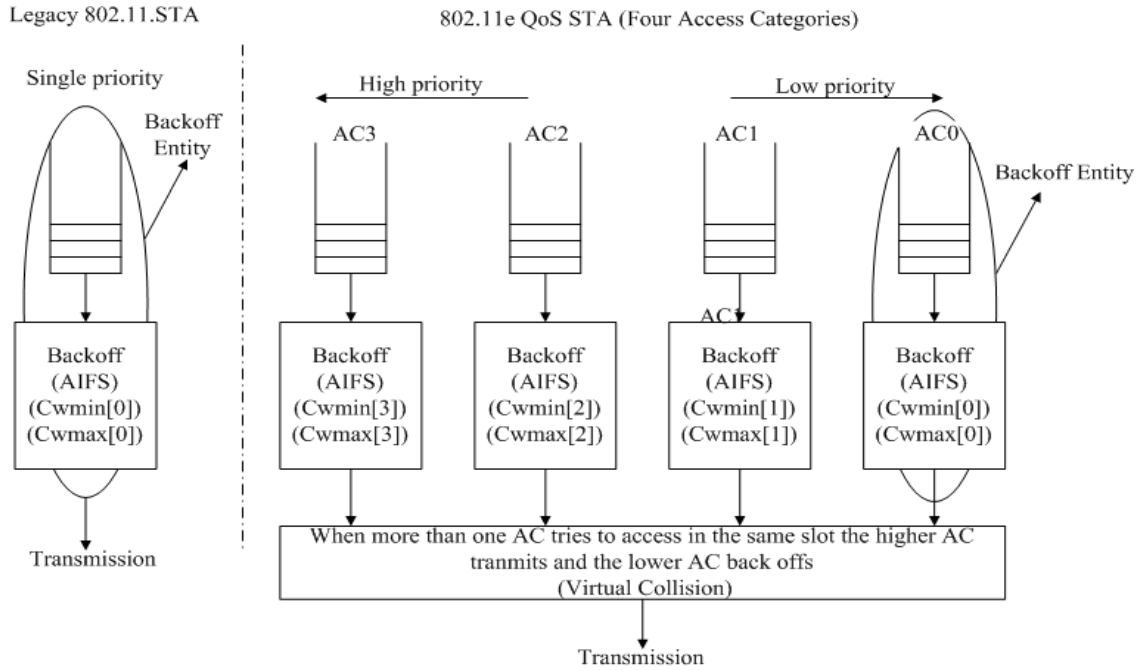


Figure 2.2: Traffic Classes in EDCA.

this case, the low priority AC doubles its contention window or uses its maximum contention window as in DCF.

A station is granted a Transmission Opportunity (TXOP), a period of time to use the medium, when it wins the contention. The duration of the TXOP is specified per AC and is used to send multiple frames from the same AC as long as they fit within the TXOP duration. These TXOPs are adjusted dynamically according to the values sent by the AP in a beacon frame thus adapting to the ever changing conditions of the network. In EDCA, each AC uses either the two way-handshake or the RTS/CTS based four-way handshake just as in DCF.

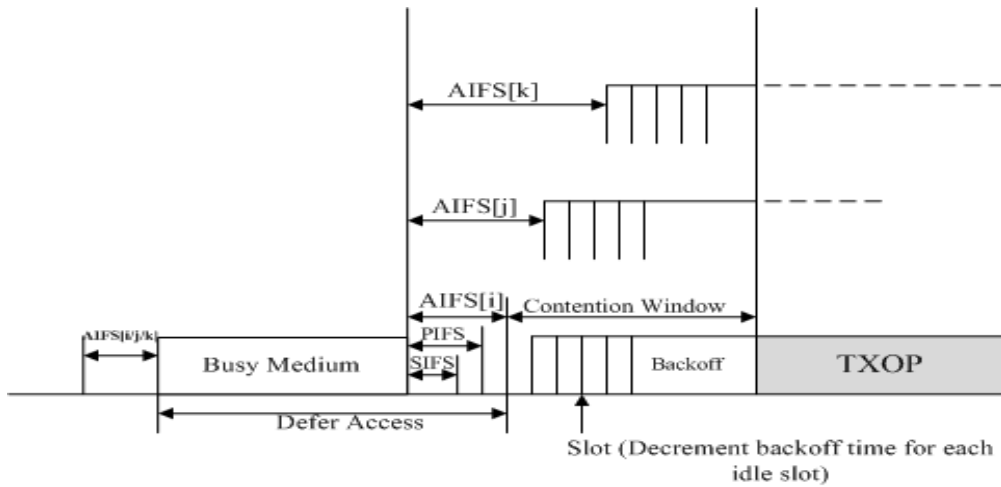


Figure 2.3: Service Differentiation in EDCA.

Two of the significant observations to note in EDCA are: Each AC uses separate set of channel access parameters; and the same set of channel access parameters in each AC are used for both the contention and service differentiation. The QoS support in EDCA is provided only in terms of service differentiation.

2.4 P-Persistent 802.11

In the P-Persistent 802.11 protocol, a station, at the beginning of each empty slot, contends for the channel (transmits a frame) with a probability p or defers with a probability $1-p$. It waits for a DIFS period of idle time before it attempts to access the channel. Therefore, it is very similar

to the standard IEEE 802.11 protocol. The only difference between them is the selection of the back-off interval and the parameters they use to do so (see Fig. 2.4).

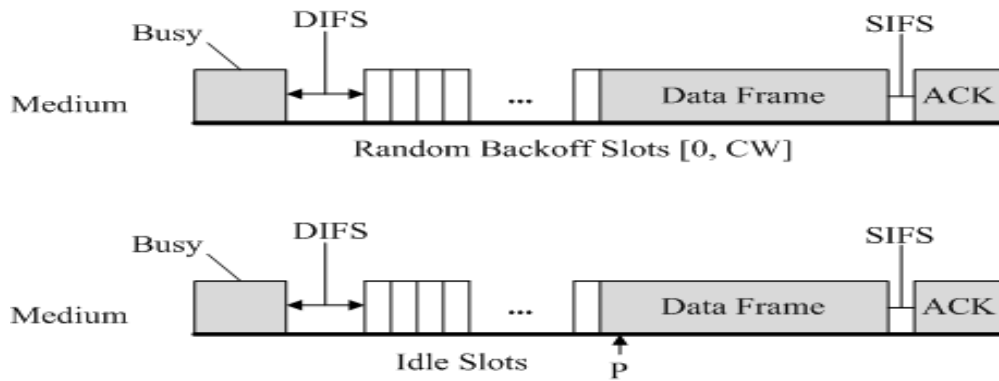


Figure 2.4: P-Persistent 802.11 vs IEEE 802.11 DCF.

The P-Persistent 802.11 protocol uses one parameter p in the back-off interval generation whereas the standard 802.11e uses four parameters CW_{min} , CW_{max} , AIFS, and the persistence factor PF. In P-Persistent 802.11, the back-off interval is sampled from a geometric distribution with parameter p (as opposed to the binary exponential back-off used in the standard DCF version). It is also verified that it can approximate the standard 802.11 at least from the capacity point of view when the average back-off interval is the same as in the standard 802.11 [CCG00a, CCG00b]. The relation between p and CW is given in [Bia00] as

$$P = \frac{2}{CW + 1} \quad (2.1)$$

Where, CW is the average contention window corresponding to that P-Persistent 802.11. Hereafter, we refer to the P-Persistent 802.11 protocol as PDCF.

CHAPTER 3 : NEW FRAMEWORK FOR QoS PROVISIONING IN 802.11 WLANs

3.1 Introduction

In the recent past, the deployment of WLANs has grown many folds for its flexibility and non-wired capability especially, in hotspot locations like bookstores, airports, shopping malls, etc. With the advent of multi-media capable mobile devices like iPhone, android based smart phones, iPad, and cheaper laptops, the use of multi-media applications will increase tremendously in WLANs and so is the requirement to provide QoS delay and/or throughput guarantees at the same level as the wired counterpart.

IEEE 802.11 employs a carrier sense multiple accesses with collision avoidance (CSMA/CA) mechanism called Distributed Coordinated Function (DCF) as a basic access method and is designed for best effort services only. Due to collisions and subsequent retransmissions, the IEEE 802.11 cannot achieve theoretical capacity bound [Bia00] even though it is compatible with the current best effort service model in wired networks. IEEE 802.11 also suffers from variant transmission delay [KH04].

The lack of built-in mechanism to support real-time services makes it very difficult to provide quality of service (QoS) guarantees for the throughput-sensitive and delay sensitive multimedia applications. Therefore, modification of the existing 802.11 standards is necessary.

The new 802.11e standard [IEE04] provides an enhanced DCF (EDCF) protocol, also referred to as enhanced distributed channel access (EDCA). In IEEE 802.11e, the Hybrid Coordination Function (HCF) provides QoS guarantees through a centralized controlled channel access (HCCA) using polling and scheduling of the traffic. But, HCF is far from deployment in the real world. Although, IEEE 802.11e is being proposed as the upcoming standard for the enhancement of service differentiation in the EDCA, QoS guarantee is still a challenging problem in EDCA.

In the EDCA, different classes of traffic are mapped to four access categories (ACs). Each AC uses a set of parameters for channel access *viz.*, CW_{min} (minimum contention window), CW_{max} (maximum contention window), AIFS (arbitration inter-frame space), and a period of transmission time called Transmission Opportunity (TXOP). The persistence factor PF is the parameter used in EDCA to change the size of the contention window after collisions. The ACs in a node not only compete for the channel among themselves but also compete with the ACs of the other nodes in the WLAN. The internal competition leads to virtual collisions. If the value of PF=2, the virtually collided ACs will double their CW just as in the regular collisions between nodes and thereby leading to reduced channel utilization. This drawback is overcome by a virtual collision handler which gives channel access to a higher priority AC, causing the lower priority AC to multiply its CW by the value of PF. There are two drawbacks with this solution. When there is no real collision

for the higher priority AC then the lower priority AC is unnecessarily penalized. Also, it is robbed of fairness in channel access when compared to the same priority ACs from other nodes. The external competition leads to real collisions which also contribute to the reduction in the system utilization.

The QoS support offered in EDCA is only through service differentiation of the traffic by prioritizing the values of the channel access parameters in ACs. We observe that the service differentiation and channel access are two different components supported by the same set of parameters, thereby increasing the complexity of tuning the 802.11e MAC. A desired performance on one component may lead to a negative impact on the other component.

EDCA does not provide delay and/or throughput guarantees as there is no mechanism to prevent the excess of traffic that affects the QoS of already admitted traffic flows. Even if HCCA is used, new traffic flows might use contention based EDCA. This will result in excess of traffic in each node and leads to packet drops of the flows admitted by HCCA. Therefore, QoS guarantee is still a challenging problem in IEEE 802.11e and needs further study [ZLC04]. In order to support QoS guarantees, an admission control is essential in both EDCA and HCCA.

Several QoS-enabling mechanisms have been proposed as extensions to DCF over the last few years. These proposals provide service differentiation among different traffic classes, but they also suffer from high throughput variability, inability to achieve maximum system capacity and lack of fairness, especially to traffic from a low priority class.

As wireless LANs may be considered as just another technology in the communications path, it is desirable that the architecture for QoS follow the same principles in the wireless network as in the wire-line Internet, assuring compatibility among the wireless and the wire-line parts [BP02]. Most of the solutions in the literature add some extensions or modify the components to address the problems discussed above. Our approach to solve this problem is to separate the different functionalities into distinct components with each component using its own set of parameters. In this chapter, we propose a new framework that provides service differentiation and channel access, and provides statistical QoS guarantees in EDCA.

The rest of this chapter is organized as follows. We review the literature and present related work in section 3.2. In section 3.3, we present our QoS framework. The simulation results are presented in section 3.4.

3.2 Related Work

IEEE 802.11 WLANs have received considerable attention in recent years. Various Distributed Fair Scheduling algorithms are proposed in the literature to provide fairness of the bandwidth and service differentiation by assigning different weights to flows based on their priority [VDG05, BP02, PBK02].

Yang [YK04] proposed QPART (QoS Protocol for ad Hoc Real-time Traffic) which provides QoS guarantees by adapting the contention window sizes at the MAC layer. The architecture consists of a QoS Aware Scheduler and QoS Manager above a core MAC layer. The scheduler realizes the functionality of QoS aware scheduling while the QoS manager implements admission control and conflict resolution. The Core MAC layer uses 802.11 EDCF and there would still be internal collision and therefore reduced system utilization. The service differentiation and channel access are still not separated. Its architecture and implementation are completely different from the framework proposed in this dissertation, which is based on the P-Persistent 802.11 MAC protocol. Liu [LLC03] proposed a MAC architecture where different adaptors can be added above the core MAC layer to provide service differentiation. Each adaptor works on adjusting only one of the parameters *viz.*, CW , CW_{min} , CW_{max} , etc. But, the adaptors are very basic and do not tune based on the network conditions. The ACs still compete among themselves and the parameters are used for both service differentiation and channel access.

Ge [Ge04] extended Cali's work to multiple queues to provide throughput service differentiation while achieving the maximum system capacity. However, the scheme proposed in [Ge04] uses separate P-Persistent probabilities for each access category and the AC still compete among themselves. And in order to achieve a given throughput differentiation, we need to calculate the P-Persistent probabilities for each AC.

3.3 The Proposed QoS Framework

Our framework consists of three components *viz.*, a core MAC layer, a Scheduler and an admission control. The core MAC layer, implemented using P-Persistent 802.11, concentrates on just the channel access to improve channel utilization. The scheduler provides service differentiation according to the weights assigned to each AC. The admission control provides statistical QoS guarantees in collaboration with the Core MAC layer and the scheduler. Fig. 3.1 shows the block diagram of our framework. One additional feature with this framework is that each component can be tuned and improved independently of the other.

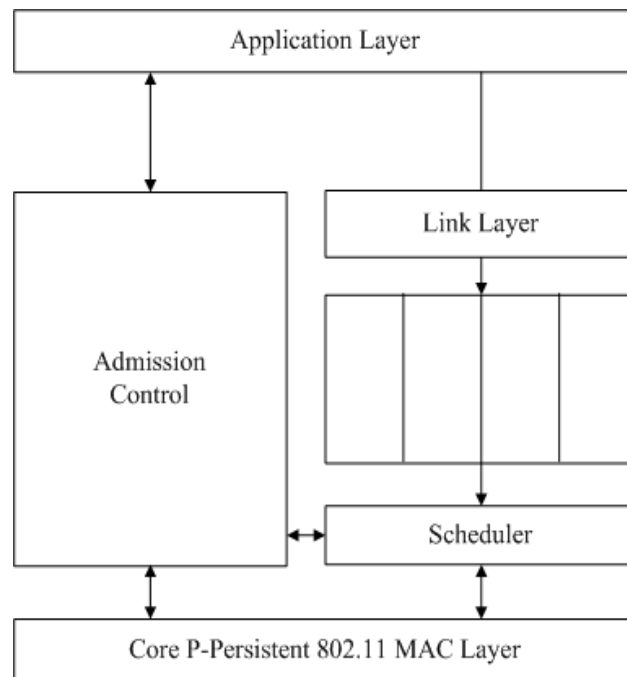


Figure 3.1: Framework for QoS Support.

3.3.1 The Core MAC Layer

In the core MAC layer, there are no separate channel access parameters for different traffic classes as in the standard EDCA. It has only one parameter p , called the transmission probability, which the P-Persistent 802.11 in the core MAC layer uses for the channel access. The core MAC layer receives only one packet from the scheduler from any of the four ACs for the channel access. Thus, it can just concentrate on optimizing the channel access irrespective of the traffic class and thereby improves the overall channel utilization. Recall that in the EDCA protocol, each traffic class has different channel access time leading to reduced channel utilization.

The core MAC layer plays an important role in the implementation of the admission control by providing information about channel access delays, network and load conditions. Since it is difficult to get this information in the distributed scenario, a complete understanding of the behavior of the P-Persistent 802.11 used in the core MAC layer helps the implementation of an admission control. Below, we define the calculation of the optimum p to achieve efficient channel access.

3.3.1.1 Optimum p for Efficient Channel Access

In order to achieve a maximum possible system throughput, the core MAC layer needs to use an optimum value for the channel access parameter p for a given network and load conditions. In

[Bia00], Bianchi derived a closed form for this parameter, denoted P_{opt} , under saturated conditions as given below:

$$P_{opt} = \frac{1}{N \cdot \sqrt{\frac{T_C^*}{2}}} \quad (3.1)$$

Where N = number of nodes and $T_C^* = \frac{T_C}{\sigma}$. σ is the slot time and T_C is the average time the channel is sensed busy due to collision. For the basic and RTS/CTS access methods, T_C is given by

$$\begin{aligned} T_C^{basic} &= H + L + EIFS \\ T_C^{RTS/CTS} &= RTS + EIFS \end{aligned} \quad (3.2)$$

Where $H = PHY_{hdr} + MAC_{hdr}$ is the time to transmit a packet header, RTS is the time to transmit the RTS frame and L is the packet transmission time. Our core MAC layer uses p calculated from eq. 3.1 dynamically based on the network and load conditions at regular intervals.

3.3.2 The Scheduler

The scheduler is built above the core MAC layer and its basic functionality is to provide service differentiation to the different classes of traffic. The scheduler selects the next packet from one of the ACs, after the core MAC layer sends a packet successfully and requests for another packet, or at the arrival of a new packet when all the ACs are empty.

In our framework, the scheduler is provided with weights for each AC. It sends the next packet from an AC based on the weights assigned to it. Thus, it completely eliminates the virtual collisions. It also provides a much easier and more accurate control in providing the QoS service differentiation for all classes within a node. In addition, the weights in the scheduler can also be changed by an access point in an infrastructure based hotspot to optimize the revenue for the service providers. Since the same set of parameters is used for both the channel access and the service differentiation in EDCF, there is no easy way to enforce a certain differentiation ratio among different traffic classes in a node.

In our QoS framework, we used a packet-based weighted fair queuing algorithm in the scheduler to provide throughput service differentiation at the node level. When one of the queues is empty, it sends the packets available in the remaining queues while keeping track of the deficit in the empty queue. It tries to compensate the deficit once the empty queue is filled with packets. We also like to point out that throughput service differentiation means bandwidth reservation per class within a node. The scheduler divides the bandwidth among the classes based on the differentiation

parameters. For example, a service differentiation of 3:2:1 provides a bandwidth reservation of $1/2$, $1/3$ and $1/6$ of the total bandwidth allocated to a node to the three classes within a node.

3.3.3 Admission Control

The IEEE 802.11 MAC protocol is based on the CSMA/CA access mechanism. Due to its inherent distributed nature, it is very difficult to provide QoS guarantees in an ad hoc scenario. In order to provide QoS guarantees, there needs to be some kind of check on the number of flows or load in the system; admission control is the best approach to provide this check. Even with the presence of admission control in 802.11, we can only provide statistical QoS guarantees.

The admission control in our framework can be implemented in both distributed and centralized scenarios. We developed a measurement-aided model-based admission control in both the distributed and centralized scenarios. In our framework, the application layer interacts with the admission control to find out whether it can allow a new flow of traffic. The core MAC layer can measure various parameters like service time delay, channel capacity or utilization, collisions, collision length and nodes. The admission control unit interacts with the core MAC layer to get the measured information. It uses the measured information along with the analytical model to estimate the new delays on account of the new flow. It admits the new flow if the estimated new

delay does not affect the previously admitted flows and also meets the delay requirement of the new flow.

3.4 Results and Analysis

In this section, we evaluate the effectiveness of the separation of channel access and service differentiation. We present the simulation results and analysis for the scheduler and the core MAC layer proposed in our framework. The admission control is evaluated in a later chapter. The simulation results show that our framework provides the desired throughput service differentiation according to the values set in the QoS differentiation parameters while providing the maximum possible system throughput.

We compared the performance of our scheme with the adaptive EDCF (AEDCF) scheme [RNT03]. AEDCF is a dynamic scheme that adapts to the varying network and load conditions to provide a best possible system performance for the given parameters. We use AEDCF scheme for comparison for the following reasons. The main emphasis in this section is to show the advantages of separating the channel access and scheduling components in the IEEE 802.11e MAC protocol and so we did not choose a scheme which implements an admission control. The AEDCF scheme uses the same set of parameters for both throughput QoS differentiation and channel access. Thus, it allows us to compare the efficiency of separating channel access and service differentiation in our

framework with a scheme which uses the same set of parameters for both the functionalities. The NS2 implementation of [Ni02] was obtained with permission from the author Qiang Ni [RNT03].

Our tests have shown that it is very difficult to set the AEDCF parameters to accurately achieve a desired throughput QoS differentiation among the ACs of a node. For example, there is no clear way of setting the parameters CW_{min} , CW_{max} , $AIFS$, and PF in order to achieve a throughput ratio of 2:1 (3:2:1) between two (three) ACs of traffic. In contrast, achieving an exact throughput differentiation in our framework is straightforward. Since, we separated the channel access component from the service differentiation component the core MAC layer can concentrate on achieving efficient channel access and is therefore capable of achieving maximum possible system throughput than the AEDCF.

Our performance tests also showed that for a given throughput differentiation, our scheme provides better channel access delay, packet service time delay and one hop delay than the AEDCF scheme.

3.4.1 Simulation Setup

The simulation results for all the tests reported in this section are generated using the NS2 simulator [Pro09]. The scheduler and the core MAC layer are incorporated into the NS2 simulator by modifying the protocol stack of the wireless node and its IEEE 802.11 EDCA implementation.

The original 802.11e MAC can still be run without any change in its functional behavior. The scheduler decides the next frame to be sent from the queues associated with the different ACs of traffic and informs the Core MAC layer about the selected frame. Upon transmitting the frame, the MAC layer calls the scheduler to provide the next frame. We also added a static address resolution protocol (ARP) and a dumb routing protocol to the NS2 code to avoid any overhead due to the ARP and the routing protocol. The static ARP doesn't send any ARP packets and the dumb routing protocol does not use any routing mechanism. It just sends the data packets down the stack to the link layer.

In our study, we simulate a WLAN made up of wireless stations with no hidden nodes. We studied the performance of our framework for different sets of wireless stations varying from 5 to 100. However, we present results only for 10 nodes unless specified exclusively in the graphs. The packet sizes were also varied from 10 to 100 data transmission slots.

The WLAN operates in IEEE 802.11a mode with each station operating at 54Mbps data transmission rate. Our simulation scenario consists of a circular topology, where node 'i' sends traffic from all of its ACs to node 'i+1' when i is less than 10, and the last node '10' sends its traffic to the node '1'. The MAC layer system parameters are set as shown in Table 3.1.

Table 3.1: 802.11 MAC Parameters

tslot	9 μ s
SIFS	16 μ s
DIFS	34 μ s
Data Rate	54 Mbps
PLCP Data Rate	6 Mbps
Preamble Length Bits	144 bits
PLCP Header Length	48 bits
ACK Length Bytes	14 bytes
MAC Header Length Bytes	28 bytes

3.4.2 Throughput Differentiation

In the following experiment, we evaluated the performance of the scheduler from our framework and AEDCF in terms of throughput service differentiation among the traffic ACs. We ran each simulation run five times for a duration of 50 seconds with traffic sources activated at $t = 0.000001$ seconds and deactivated at $t = 50$ seconds. We generated the CBR traffic at the same rate in each AC in each station. We conducted the tests for two (and three) ACs and the traffic rates are varied from 1 Mbps to 3.5 Mbps to generate a system load of 20 Mbps (30Mbps) to 70 Mbps (105 Mbps). The data packet sizes are also varied from 10 to 100 transmission slots (a size of 100 slots is larger than the maximum 802.11 frame size but is used for evaluation purposes only).

In our scheme, the desired throughput service differentiation is easily and accurately achieved by simply setting the weights of the different traffic ACs in the desired proportions in the scheduler component. The AEDCF scheme, however, uses a set of parameters namely CW_{min} , CW_{max} ,

AIFS, and *PF* for each AC to provide throughput differentiation as well as channel access. It becomes a very complex operation to set these parameters correctly in order to achieve a desired throughput differentiation among the ACs.

To further illustrate this important issue, let us assume we want to achieve a throughput service differentiation ratio of 3:2:1 between AC3, AC2 and AC1 traffics. To do so, we simply set the weights of AC3, AC2 and AC1 to 1, 2/3, and 1/3 respectively in our scheduler. In the AEDCF scheme, however, a tedious search process is needed to find the parameter values to achieve the required differentiation ratio. For example, setting each of the parameters CW_{min} , CW_{max} , PF for AC2 and AC1 to two and three times the values of AC3 parameters does not result in a throughput differentiation ratio of 3:2:1.

We evaluated the behavior of AEDCF when we double and triple the parameter values of AC2 and AC1 with respect to AC3. For example, we used the MAC parameters given in 3.2 for the AEDCF scheme with three ACs. In our results, we present the throughput ratio for ACs

Table 3.2: AEDCF3A MAC PARAMETERS FOR THE THREE ACs

Paramters	AC3	AC2	AC1
CWmin	5	10	15
CWmax	200	400	600
AIFS (/mus)	34	34	34
PF	2	4	6

as a fraction of system throughput. So, a throughput ratio of 3:2:1 is equivalent to 0.5, 0.33 and 0.17 in terms of fraction of the system throughput for AC3, AC2 and AC1 respectively. Our

framework with an underlying P-Persistent Core MAC is referred to as SPDCF from now onwards. We use SPDCF3 to mean our scheme with a throughput differentiation of 3:2:1 for three ACs of traffic respectively. We use the notation AEDCF3A to denote the AEDCF scheme with three ACs and with an ad-hoc approach of setting the parameter values in proportion to the required throughput differentiation. Thus we use AEDCF3A to denote the AEDCF scheme with parameters set according to 3.2 for three ACs of traffic.

In order to get a throughput ratio of 3:2:1 for the AEDCF scheme, we found that the parameter values depend on the traffic load and we had to run several simulation tests with different parameter combinations at each load rate. Table 3.3 gives the values of the parameters that reasonably produced the desired throughput ratio 3:2:1 at the various traffic loads for 10 nodes. We refer to the AEDCF scheme that uses the parameters values given in Table 3.3 as AEDCF3. The parameter values for the ACs given in Tables 3.3 show that AC2 and AC1 were given higher priority than AC3 in terms of AIFS. It should be noted that the virtual collision handler and the AEDCF behavior give skewed priority for AC3 over AC2 and AC1. In order to achieve 3:2:1 differentiation, the parameter values were chosen so that they compensate the emphasis given by the virtual collision handler to AC3. This explains why AC2 was sometimes given more favorable parameter values compared to AC3 but AC3 still ended up getting 1/2 the throughput. Also, the throughput differentiation in AEDCF is not dependent on just one parameter but a combination of all the four parameters from each AC.

Table 3.3: AEDCF3 MAC PARAMETERS FOR THE THREE ACs (3:2:1)

load/class/node (Mbps)	AC	CW _{min}	CW _{max}	AIFS	PF
1, 1.25, 1.5, 1.75, 2.0 Mbps	3	7	200	43	2
	2	5	275	34	2
	1	9	350	14	2
2.25 Mbps	3	7	225	43	2
	2	5	250	34	2
	1	9	325	34	2
2.5, 2.75 Mbps	3	7	250	43	2
	2	5	200	34	2
	1	9	300	34	2
3.0 Mbps	3	7	300	43	2
	2	5	200	34	2
	1	9	300	34	2
3.25, 3.5 Mbps	3	8	350	43	2
	2	5	250	34	2
	1	9	325	34	2

In summary, the ad-hoc scheme AEDCF3A is based on Table 3.2 that increases the value of the parameters CW_{min} , CW_{max} , and PF for a traffic AC in proportion to the desired increase in the throughput of this AC. The guided scheme AEDCF3 is based on Table 3.3 that sets the values of the channel access parameters guided by an exhaustive search to achieve the desired throughput differentiation.

We next examine the performance of the two schemes under different CBR traffic loads. The load rate for each AC per node is varied from 1 Mbps to 3.5 Mbps giving a total system load of 30 Mbps to 105 Mbps for the traffic in the three ACs.

We observe from Figs. 3.2, 3.3 and 3.4 that there is no throughput differentiation for loads below 1.25 Mbps in all the schemes AEDCF3A, AEDCF3 and SPDCF3. This is because the

traffic load is not sufficient enough to differentiate between the three ACs. As the load is increased up to 2 Mbps, the AC3, AC2 and AC1 are differentiated but not to the desired ratio as the total load is still not sufficiently large enough. As the load is increased further, the AEDCF3 and SPDCF3 maintained the desired throughput differentiation of 3:2:1 (0.5, 0.33 and 0.17 fractions of system throughput). However, the ad-hoc AEDCF3A scheme gave less preference to AC2 traffic and least preference to AC1 traffic in accordance to the parameter values given to it. Thus, we show

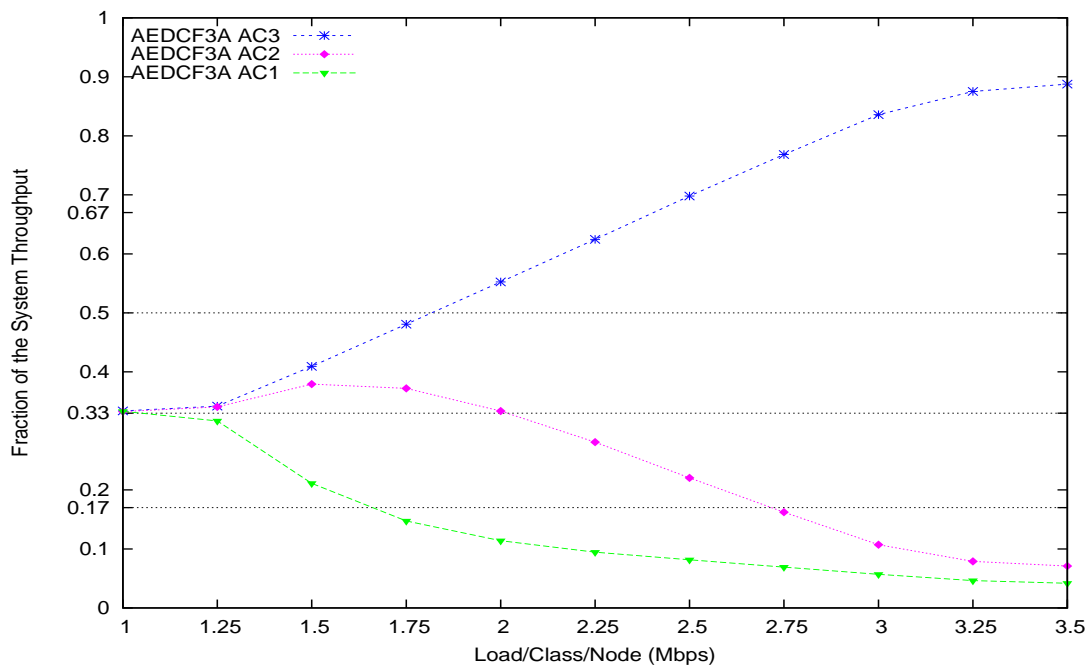


Figure 3.2: AEDCF3A: fraction of the system throughput.

that achieving a desired throughput differentiation is not a straightforward process in the IEEE 802.11e. We achieve this very easily in our framework by separating the channel access and

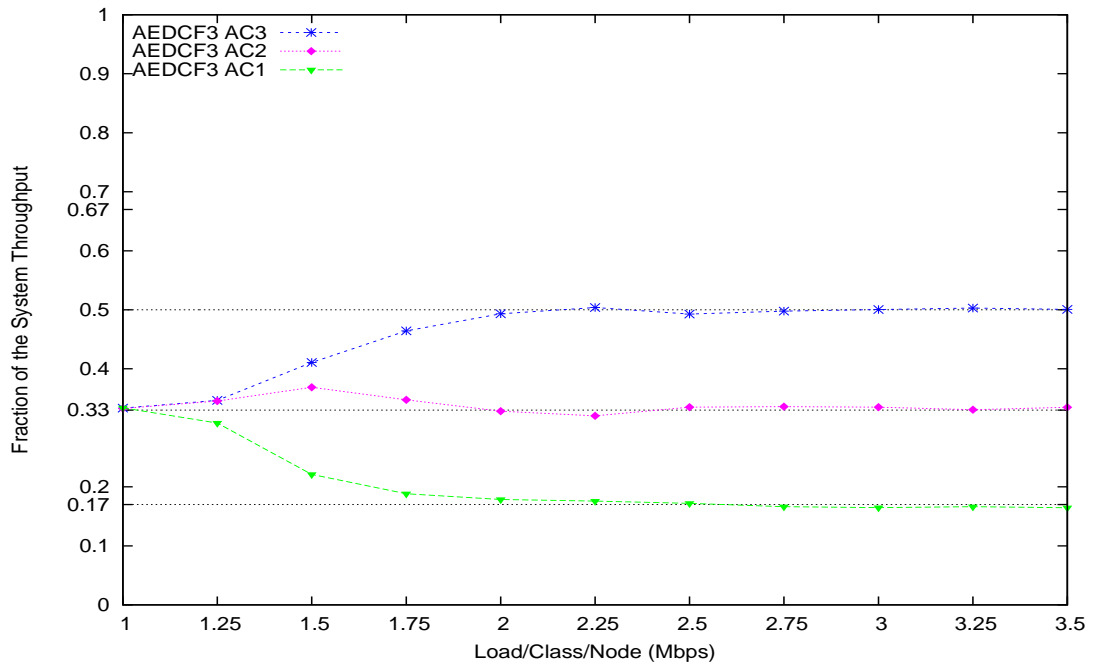


Figure 3.3: AEDCF3: fraction of the system throughput.

scheduling components. The desired throughput differentiation is achieved very easily by setting the differentiation parameters in the scheduler.

3.4.3 System Throughput

In the previous section, we saw the advantage of the scheduler in providing throughput differentiation. In this section, we look into the performance of the core MAC layer in terms of system throughput. A higher system throughput means better channel access.

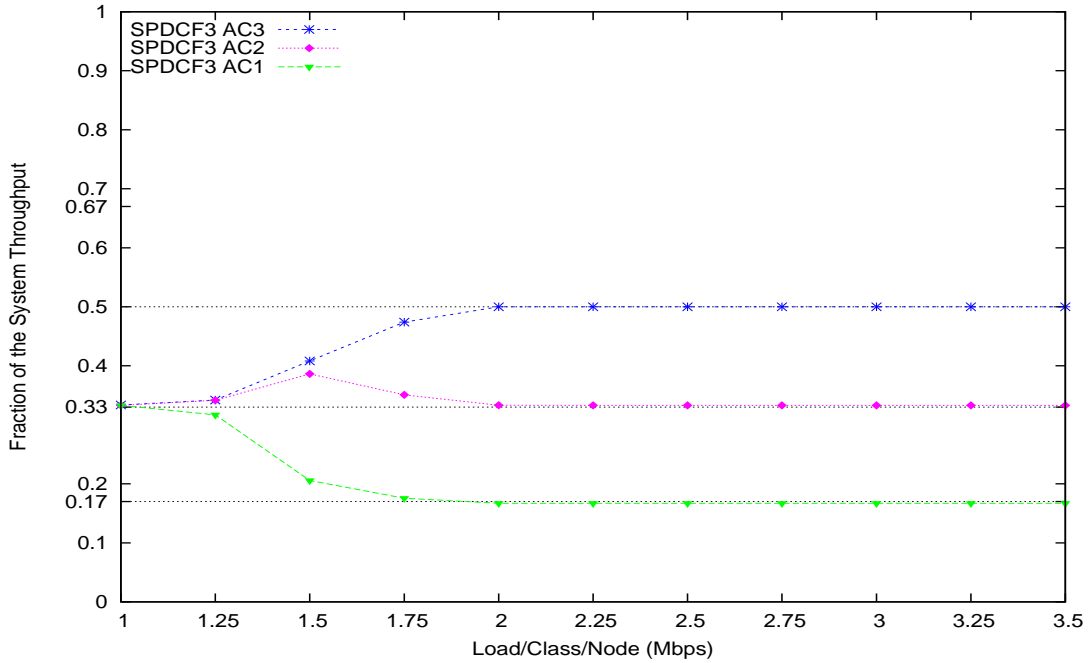


Figure 3.4: SPDCF3: fraction of the system throughput.

Fig. 3.5 shows that the SPDCF3 achieved better system throughput than both the AEDCF3A and AEDCF3 schemes for a given throughput differentiation of 3:2:1. In SPDCF3, the core MAC layer's function is to just provide the channel access to the packet selected by the scheduler. So, it sets its channel access parameter p to an optimum value, based on the network and load conditions without any concern to the differentiation requirement. However, in the AEDCF, the differentiation component and channel access component are catered to by the same parameters. Even though the AEDCF scheme dynamically adapts its channel access parameters, it is still bound by the initial input access parameters. The virtual collisions within a node also contribute to lower system throughput.

The AEDCF3 performed lower than the ad-hoc AEDCF3A because the MAC parameters for the AEDCF3 were set, via exhaustive search, to provide a specific throughput differentiation of 3:2:1. In the process, this setting affected the overall system throughput as the parameters are also used for channel access and may not be the optimum values to achieve optimum system throughput. This shows that using the same set of parameters for both channel access and service differentiation will not result in optimum performance in both components at the same time.

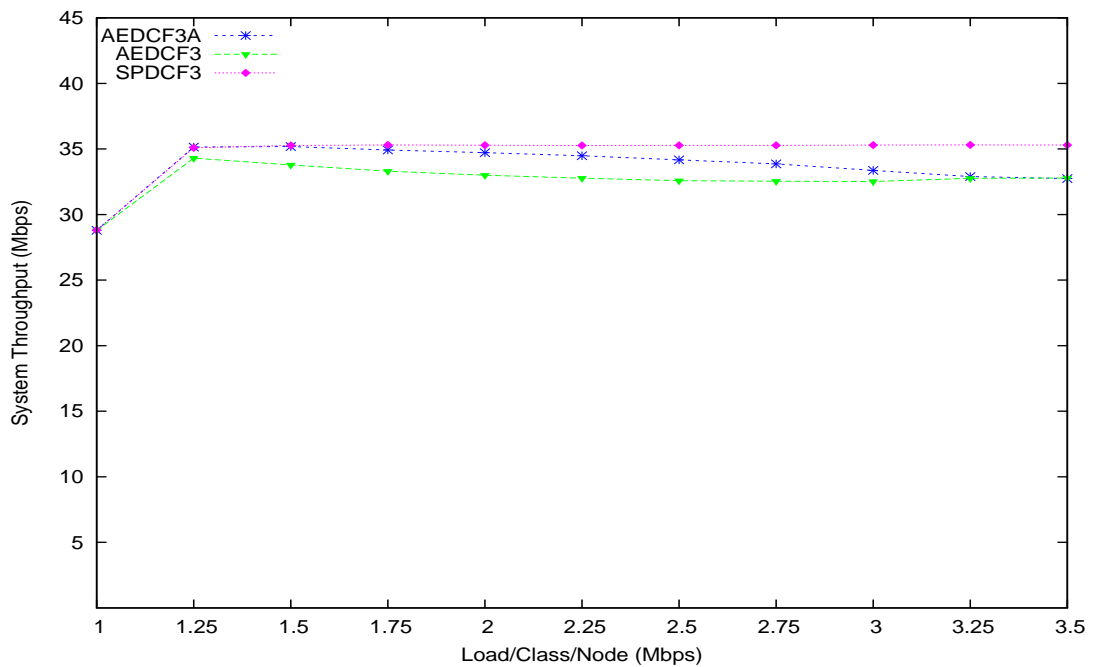


Figure 3.5: System throughput.

3.4.4 Packet Delay

In this section, we evaluate the performance of our scheme and AEDCF in terms of three types of packet delays *viz.* channel access delay, packet service time delay and one hop delay for a given throughput differentiation of 3:2:1.

We define the three delays as follows: The channel access delay is the amount of time a packet takes from the time it is at the head of the MAC queue ready for transmission till the time it starts accessing the channel (i.e., till the time the first transmission attempt of this packet starts). Notice that the channel access delay does not depend on the length of the packet. The packet service time delay is the amount of time it takes from the time the packet is at the head of the MAC queue till it receives an acknowledgement or is dropped after its retransmission limit is exhausted. The one hop delay is the amount of time a packet takes from the time it arrives at the queue till it is successfully transmitted. The one hop delay is the sum of queue delay and service time delay for a packet.

3.4.4.1 Channel Access Delay

We study the channel access delay for both the AEDCF and SPDCF schemes for a throughput differentiation of 3:2:1. The AEDCF uses the same parameters given in Table 3.3. Since, we now have the same throughput ratios in both schemes; we compare their channel access delay for the three traffic ACs.

Fig. 3.6 shows the channel access delays for each of the three ACs of traffic for the AEDCF3 and SPDCF3 schemes. The SPDCF3 scheme performed better than the AEDCF3. In our scheme, the core MAC layer tries to improve the system capacity by using an optimal p , obtained from Eq. 3.1. A greater throughput implies lesser channel access latency. However, in the AEDCF3 scheme, the parameters used provide prioritized channel access without emphasizing on optimal throughput. When we set the parameters to achieve a throughput differentiation of 3:2:1, the parameters are not optimal enough to provide better channel access delay. As shown in Fig. 3.6,

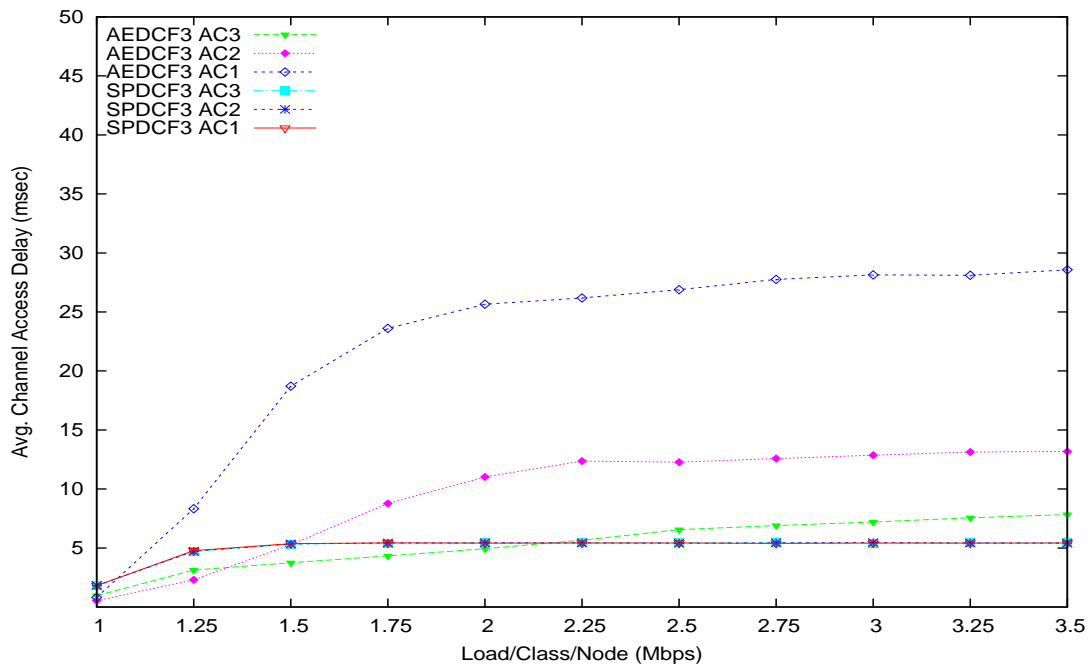


Figure 3.6: Channel access delay.

each AC in the AEDCF3 contends for the channel independently and thereby, the channel access delay will be different for different ACs whereas in SPDCF3, there is only one channel access for

all the ACs and so all the ACs will have the same channel access delay. The SPDCF3 provides better channel access delay than that provided by AEDCF3.

3.4.4.2 Packet Service Time Delay

The service time of a packet depends on the efficiency of its channel access. Therefore, the results from channel access in the previous section would give us some idea about the packet service time. Since AEDCF3 is not optimized for channel access, it might take more time to access the channel and might involve more collisions. Fig. 3.7 shows that the packet service time for AEDCF3 is more than that of the SPDCF3 for all the three traffic ACs. The packet service time for AC1 traffic in the AEDCF3 scheme is higher than that of AC3 and AC2 as it is given least priority channel access than AC3 and AC2. Similarly, the packet service time for AC2 traffic in AEDCF3 scheme is higher than that of AC3 as it is given lesser priority channel access than AC3 whereas the packet service time in the SPDCF3 is nearly same for all the three ACs as they have one channel access and the scheduler decides the next traffic AC which will compete for the channel access.

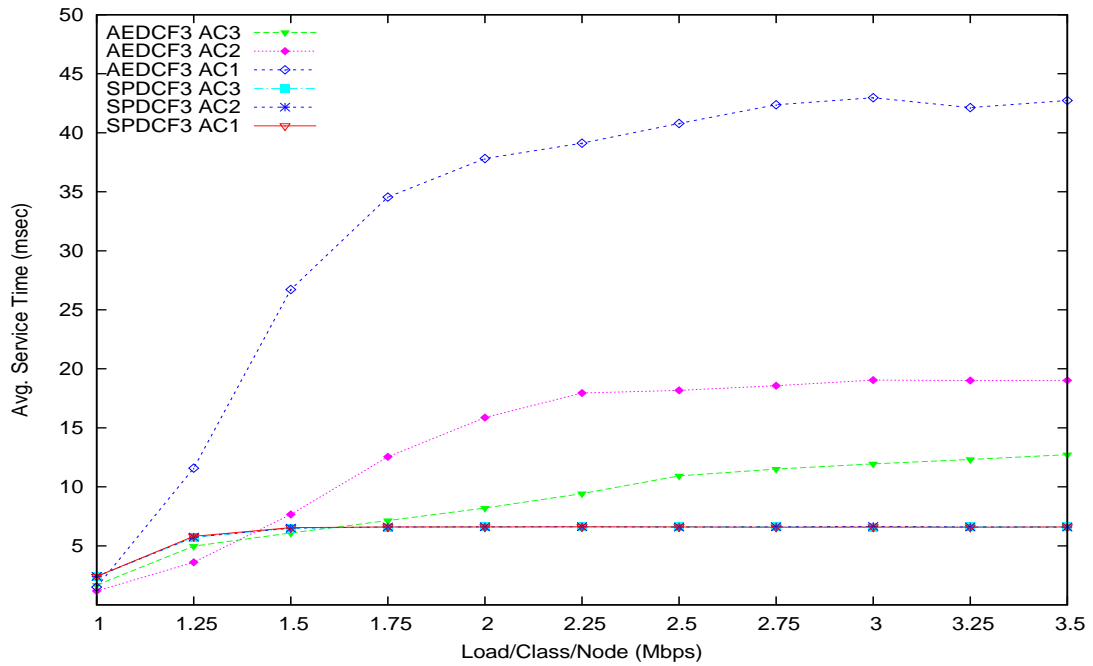


Figure 3.7: Service time delay.

3.4.4.3 One Hop Delay

The one hop delay is an important QoS metric which determines whether we can support real-time applications that require stringent delay bounds. The one hop delay in our scheme depends on two components. The core MAC layer provides the best one hop delay possible in a given network and load conditions. The scheduler provides the differentiation of the one hop delay based on the priorities of the traffic. Fig. 3.8 shows that the SPDCF3 scheme provides better one hop delay than the AEDCF3 scheme. At a load 3.5 Mbps, the one hop delay of AC1 and AC2 in SPDCF3 is about 2.91 and 1.6 times the one hop delay of AC3, whereas the one hop delay of AC1 and AC2

in AEDCF3 is about 4.16 and 2.36 times the one hop delay of AC3. In general when achieving a throughput differentiation of 3:2:1, the ratios of the one hop delay between the three classes in SPDCF3 is slightly lesser than 1:2:3 but is considerably closer to 1:2:3 than the ratio obtained by the AEDCF3. Therefore, we can use either the one hop delay metric or the throughput metric at the scheduler to represent the service differentiation. We observed similar results for the two ACs of traffic [AB06a]. Due to lack of space, we don't show the results for the case of two ACs.

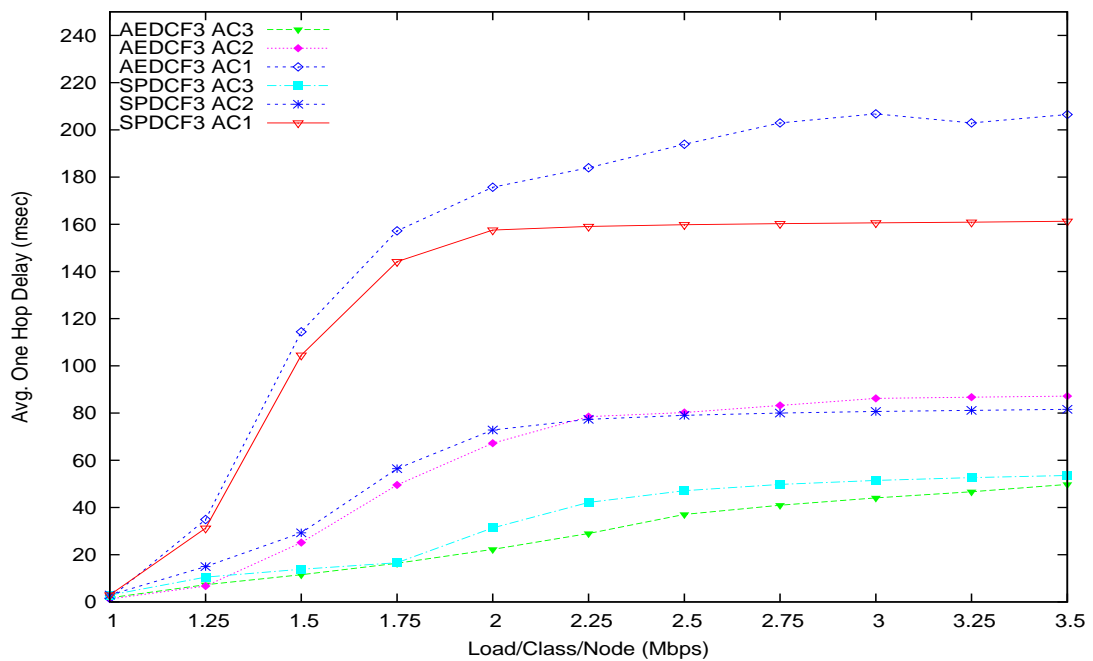


Figure 3.8: One hop e2e delay.

CHAPTER 4 : ANALYTICAL MODEL FOR P-PERSISTENT 802.11

4.1 Introduction

In WLANs, the physical medium, shared by all stations, has limited connection range and has significant differences when compared to the wired medium. Therefore, the design of wireless LANs (WLANs) needs to concentrate more on bandwidth consumption than wired networks. The design of a WLAN MAC protocol is further complicated by the presence of hidden terminal and capture effects [WPK02]. Since a WLAN relies on a common transmission medium, the transmissions of the network stations must be coordinated by the MAC protocol. Thus, the MAC protocol is the main element that determines the efficiency of sharing the limited communication bandwidth of the wireless channel. The fraction of channel bandwidth used by successfully transmitted messages gives a good indication of the overhead required by the MAC protocol to perform its coordination task among stations. This fraction is known as the utilization of the channel, and the maximum value it can attain is known as the capacity of the MAC protocol [CCG00b, CCG00a].

Mathematical models are generally used to study or describe the behavior of a system. It helps us to quantify certain performance metrics in terms of parameters of the system. Therefore, by deriving an analytical model, we can obtain better insight and therefore can use the system more

efficiently. The capacity limit of a MAC protocol also provides us with the minimum possible packet delay and end-to-end latency bounds.

In this chapter, we present an analytical model for the P-Persistent 802.11 for un-saturated conditions with unequal load distribution under the ideal channel conditions and finite number of stations. The analysis doesn't consider hidden terminals. Bianchi presented a similar evaluation on legacy 802.11 MAC using a Markov chain based model [Bia00]. He also presented a simplified case of constant window which is an approximation of the P-Persistent 802.11. However, the analysis didn't consider finite retry limits and so, may not provide an accurate model.

We believe that the Markov chain approach is fit for examining various performance metrics such as throughput, average packet service time delay and average packet one hop end-to-end delay for P-Persistent 802.11 as it is very similar to the legacy 802.11 except for the back-off window mechanism. We define our model based on the notation given in [Bia00, WPK02] and use the same assumptions. In this chapter, we present a Markov chain model for both infinite and finite retry limits of a station in a P-Persistent 802.11 WLAN.

The rest of this chapter is organized as follows. In section 4.2, we present a review of previous work on the analytical modeling of 802.11 WLANs. In section 4.3, we present a Markov chain based model for the P-Persistent 802.11 Protocol. We extend the model for the P-Persistent 802.11 to obtain the service time required by a packet present at the head of the queue in section 4.4. In section 4.5, we present the mathematical model for the end-to-end latency for a packet in a

P-Persistent 802.11 using the service time from the previous section. In section 4.6, we present an analytical model for the throughput of P-Persistent 802.11. Model validation is presented in section 4.7.

4.2 Related Work

The modeling of 802.11 has been a research focus since the standardization of IEEE 802.11 MAC. There are several research efforts in the literature that model the performance of legacy 802.11 DCF [CCG00b, CCG00a, Bia00, BT03, FZ02, WPK02, Ge04, GK04, CBV03, CBV05, ZA02, CG03].

Cali et al. [CCG00b, CCG00a] analyze the performance of the legacy IEEE 802.11 DCF protocol through the P-Persistent version of IEEE 802.11 protocol. Based on the analytical model, they observe that the system throughput relies on the value of the transmission probability p and the number of active stations [CCG00a]. Also, they suggest that the IEEE 802.11 protocol with its current settings can hardly achieve the theoretical capacity bound. They extended their work by dynamically tuning the value of p so as to achieve the capacity bound under varying network conditions [CCG00a]. Cali et al. in [CCG00b] propose a method of estimating the number of active stations via the number of empty slots and exploit the estimated value to tune the CW parameter based on a P-Persistent version of the IEEE 802.11 protocol. However, it is very difficult to model the average packet service time delay and average packet one hop end-to-end delay in

P-Persistent 802.11 using Cali's model as it is based on the number of idle slots, and the number of collisions between two successful transmissions at a system level. As a result, Cali's model doesn't provide any details at a node level.

Several other papers in the literature have attempted to improve IEEE 802.11 performance by either modifying the back-off mechanism or by fine tuning certain protocol parameters. Carvalho and Garcia Luna Aceves in [CG03] considered the impact of the minimum contention window (CW) size and the corresponding capacity improvement that is achieved when CW increases but not combined with packet retry limits and other protocol parameters. They also presented a model for average service time, i.e., the amount of time it takes for a packet at the head of the queue till it gets acknowledgement. Most of the above models describe the legacy 802.11 DCF with binary exponential back-off.

Bianchi in [Bia00] models the behavior of a station with a Markov chain model under saturated load and analyzes the throughput of the 802.11 MAC protocol in ideal channel conditions and a finite number of stations. In particular, Bianchi assumes that the packet retransmissions are unlimited and a packet is transmitted until its successful reception. Also, he assumes that the collision probability of a packet transmitted by a station is constant and independent of the number of retransmissions of the packet. Wu et al. extend the Bianchi's model to include the finite packet retry limits as defined in the IEEE 802.11 standards [IEE99] and therefore, present a more accurate model in [WPK02].

There have also been several efforts to model the average service time for the legacy 802.11 in both the saturated [GK04, CBV03, CBV05, CG03, ZA02] and unsaturated load conditions [TS04, DML05, BBA05, GC05] respectively. Most of the models are based on Bianchi's work on 802.11 in [Bia00].

Chatzimisios et al. [CBV03, CBV05], based on the Markov chain model of Bianchi [Bia00], compute various performance metrics such as the average packet delay, the packet drop probability, the average time to drop a packet, the packet inter arrival time, and the throughput efficiency. However, they did not consider the delay due to the dropped packet after the retry limit stage in the calculation of the service time. Similarly, Carvalho et al. presented analytical models for service time in legacy 802.11 MAC in [CG03]. But, the analysis didn't consider finite retry limits.

Gupta et al. [GK04] extend Wu's model [WPK02] to include channel errors for a collocated one hop 802.11 WLANs. They also calculate the average packet delay for the single hop case and the end-to-end delay for non-collocated multi-hop ring topologies.

Ziouva in [ZA02] develops a Markov chain model that introduces an additional transition state to the models of [Bia00, WPK02, CBV03] and allows the stations to transmit consecutive packets without activating the back-off procedure. This feature, which is not specified in any IEEE 802.11 standard, causes an unfair use of the medium since stations are not treated in the same way after a successful transmission. The proposed model in [ZA02] lacks validation and the calculation of the average packet delay utilizes a complicated approach that calculates the average number of the

collisions of a packet before its successful reception and the average time a station's back-off timer remains stopped.

In [TS04], Tickoo modeled each station as a $G/G/1$ queue and derived a service time distribution. Barowski in [BBA05] extended Bianchi's model [Bia00] for saturated conditions to unsaturated conditions using $M/M/1/K$ analysis. However, the model couldn't describe the behavior accurately when the system is slightly below saturation.

Most of these models use equal load among the nodes in both saturated load and unsaturated load conditions. However, in real world scenarios, this assumption is not accurate. In [TDE09], Taher developed a model for finite load for 802.11e but all the classes in every node had equal load. The first work, to our knowledge, for unequal load in 802.11 was developed by Alizadeh-Shabdiz in [AS03]. In the following sections, we present an analytical model for the P-Persistent 802.11 MAC protocol to unequal load conditions to calculate average packet service time delay and average packet one hop end-to-end delay.

4.3 Markov Chain

Consider a fixed number of contending stations, say N . A station always resides in an empty state (E) when there are no packets to transmit. In the presence of at least one packet, it resides in one of following three states: at a non-empty state (NE), where a station contends for the channel; it

moves to a transmission state (X) or an idle state (I) based on whether a packet is transmitted or not in a given time slot. After a successful transmission a station moves back to either E or NE depending on the presence of a packet. The NE state is a transient state; visiting the NE state means processing a new packet.

Let q_{0j} be the probability that a station j ($1 \leq j \leq N$) has an empty queue. Let $s_j(t)$ be the stochastic process that represents a station j 's state at a time slot t . When a station (node) is competing for the channel, the time interval between its two consecutive states is referred to as the slot transition time and is variable as it observes one of the three mutually exclusive events namely, a successful transmission, a collision or an idle slot (no transmission) from the rest of nodes in the network. The idle slot is equal to the constant system parameter σ based on the PHY layer used. We develop our model based on the notation and assumptions in [Bia00]. Let K denote the maximum queue size in terms of the number of packets submitted to the MAC layer for transmission and λ_j be the data load rate at node j . We use M/M/1/K queue for the analysis.

4.3.1 No Retry Limit

We first present a very simple case of unbounded retransmissions. When there is no limit on the number of retransmissions, a packet at the head of the queue remains there until it is successfully transmitted. Due to this, the average service time delay and average one hop end-to-end delay of

a packet could be negatively impacted. The state model in Fig. 4.1 shows the behavior of a node j for the case of no retry limit. As described above, the node remains in one of the four states. A station j transmits in a physical slot time with probability p and remains idle with probability q . It will succeed in its transmission with probability V_{Sj} and fail with probability V_{Cj} . The relationship between them is given below:

$$p + q = 1 \tag{4.1}$$

$$V_{Sj} + V_{Cj} = 1 \tag{4.2}$$

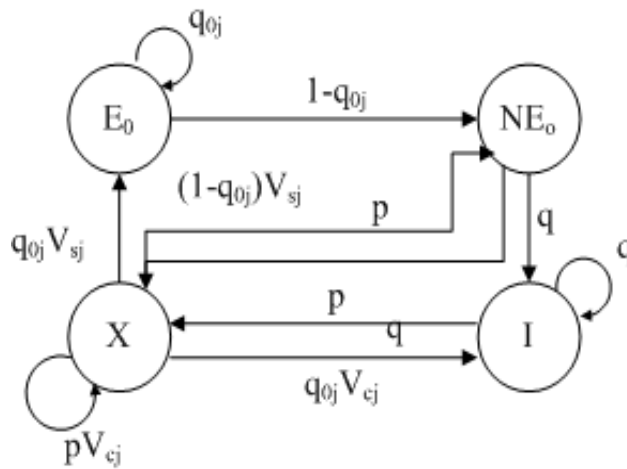


Figure 4.1: Markov chain model for the no retry limit case for node j .

The non-null one step transition probabilities of station j from the above model are the following:

- Station j remains in an empty state E if the empty queue has no packet arrival in any slot time or the queue becomes empty after the successful transmission of the last packet in the queue.

$$P\{E|E\} = q_{0j}$$

$$P\{E|X\} = q_{0j} \cdot V_{Sj}$$

- Station j moves to a non-empty state NE from an empty state E when there is a new arrival or the queue is non-empty after a successful transmission.

$$P\{NE|E\} = 1 - q_{0j}$$

$$P\{NE|X\} = (1 - q_{0j}) \cdot V_{Sj}$$

- Station j is in idle state I when it doesn't transmit a packet with probability q . It can be in idle state either from NE , I or X ; the transition from X to idle occurs when the previous transmission ended in collision and station j does not send a packet in the next slot.

$$P\{I|NE\} = q$$

$$P\{I|I\} = q$$

$$P\{I|X\} = q \cdot V_{Cj}$$

- Station j is in transmission state X when it transmits a packet. It can transmit a packet either from NE , I or X . The transmission from state X to X happens when the previous transmission ended in a collision.

$$P\{X|NE\} = p$$

$$P\{X|I\} = p$$

$$P\{X|X\} = p \cdot V_{Cj}$$

Where, $P\{X|I\} = P\{s(t+1) = X, |s(t) = I\}$. Notice that q is the probability that the station has a packet but does not transmit it in a given time slot whereas q_{0j} is the probability that station j does not have any packet in its queue.

4.3.2 Finite Retry Limit

In this section, we consider finite retransmissions. Let $m-1$ be the maximum allowed number of retransmission attempts and $r_j(t)$ be the stochastic process representing the transmission stage or re-transmission $(1, \dots, m)$ of a station j at time t , where stage 1 is the initial transmission and stages

2 through m are retransmissions. We use the same value p in all the stages. Therefore, we have

$$p_i = p \quad 1 \leq i \leq m \quad (4.3)$$

The empty and non-empty states are the same as in the no retry limit model. As in [Bia00], the key assumption in our model is that in each transmission attempt, a packet in station j collides with constant and independent probability V_{Cj} irrespective of the number of the retransmission attempts. The probability V_{Cj} is a conditional probability because it is the probability of collision seen by a station j transmitting a packet, i.e., it is conditioned on the existence of the other station(s) transmitting a packet in a given time slot. Thus, the two-dimensional process $\{r_j(t), s_j(t)\}$ for station j is a discrete-time Markov chain, which is shown in Fig. 4.2. On careful observation, the finite retry limit model is the same as the no retry limit model except that the infinite stages are replaced with finite stages. The transition probabilities of station j for the finite retry limit case are the following:

- Station j remains in an empty state E_0 if the empty queue has no packet arrival in any slot time or the queue becomes empty after the successful transmission of the last packet in the queue at any stage. The queue becomes empty in stage m when the last packet is removed

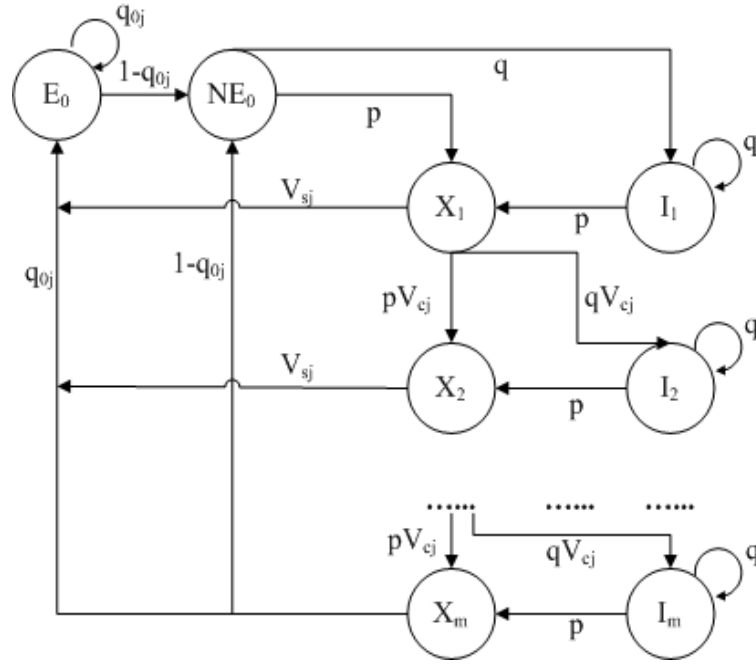


Figure 4.2: Markov chain model for the finite retry limit case for node j.

(dropped) from the queue irrespective of success or collision.

$$P\{E_0|E_0\} = q_{0j}$$

$$P\{E_0|X_i\} = q_{0j} \cdot V_{sj} \quad 1 \leq i \leq m - 1$$

$$P\{E_0|X_m\} = q_{0j}$$

- Station j is in non empty state NE_0 when there is a new arrival in an empty queue or the queue remains non empty after a successful transmission of a packet in any stage. As mentioned above, in stage m, the packet is removed from the queue irrespective of its transmission

status. So, if the queue remains non empty after this, then it moves to the state NE_0 .

$$P\{NE_0|E_0\} = 1 - q_{0j}$$

$$P\{NE_0|X_i\} = (1 - q_{0j}) \cdot V_{Sj} \quad 1 \leq i \leq m - 1$$

$$P\{NE_0|X_m\} = 1 - q_{0j}$$

- Station j is in idle state I_i when it doesn't transmit a packet from the states NE_0 , X_{i-1} or I_i .

$$P\{I_1|NE_0\} = q$$

$$P\{I_i|I_i\} = q \quad 1 \leq i \leq m$$

$$P\{I_i|X_{i-1}\} = q \cdot V_{Cj} \quad 2 \leq i \leq m$$

- Station j is in transmission state X_i when a packet is transmitted in stage 'i'. The transition from state X_{i-1} to X_i occurs when the transmission from X_{i-1} ends in a collision. The transition from NE_0 to X_1 occurs when the station j transmits a packet.

$$P\{X_1|NE_0\} = p$$

$$P\{X_i|I_i\} = p \quad 1 \leq i \leq m$$

$$P\{X_i|X_{i-1}\} = p \cdot V_{Cj} \quad 1 \leq i \leq m - 1$$

Where, $P\{Y_k|Z_l\} = P\{r(t+1) = k, s(t+1) = Y | r(t) = l, s(t) = Z\}$, and $k, l \in [1, m]$ and $Y, Z \in \{X, I, E_0, NE_0\}$.

4.4 Service Time Delay

We define the service time delay of a packet in a station j as the time interval between the time the packet is at the head of its MAC queue ready for transmission and the time when an acknowledgement is received or the packet is discarded after its retry limit for collisions is exhausted. In the no retry limit case, the packet is never dropped irrespective of the number of collisions. The assumptions for channel conditions, number of terminals and hidden nodes remain the same as in the previous section.

Chatzimisios and Carvalho presented analytical models for service time in legacy 802.11 MAC based on Bianchi's model in [CBV05] and [CG03] respectively. Chatzimisios did not consider ACK timeout in the service time for the dropped packet in the retry limit stage. Carvalho presented a simplified case of constant window in each backoff stage, but the analysis didn't consider finite retry limits. In this section, we present an analytical model for service time delay of a packet at station j for P-Persistent 802.11 for both the no retry limit and the finite retry limit cases.

We use "back-off step" and "state transition step" to mean the same thing. The time interval between two consecutive state transitions is called slot time and it is variable due to the occurrence

of three mutually exclusive events in the channel: a successful transmission, a collision or an idle slot. Therefore, the time intervals between two consecutive state transitions at a station j will contain one of the three mutually exclusive events *viz.*, 1) a successful transmission (E_{Sj}), 2) a collision (E_{Cj}), and 3) an idle channel (E_{Ij}).

We also assume that events in successive state transition steps are independent. In the DCF mode of operation, a transmitting node detects a collision if it does not receive an acknowledgment after a certain timeout (the ACK Timeout in the basic access mechanism and the CTS Timeout in the RTS/CTS mechanism). So, when a collision occurs in a state transition step, the colliding nodes resolve the collision in the same transition step and are ready for transmission in the next transition step thus, avoiding any dependencies on the number of colliding nodes in the previous transition steps. We assume that the transmission probability p is constant in all retransmission attempts and use the Markov chains presented in the previous section to derive the service time delay.

Since a node j is empty with probability q_{0j} ($j = 1$ to N), there are only a fraction of nodes which contend for the channel. Therefore for given N nodes, the number of active nodes (N_{eqv}) is given by:

$$N_{eqv} = \sum_{j=1}^N (1 - q_{0j}) \quad (4.4)$$

When a node j is observing the system, it observes three events due to the active nodes among the remaining $N - 1$ nodes *viz.*, 1) a successful transmission from one of the active nodes among the $N - 1$ nodes; 2) an idle state when none of the active nodes transmits a packet; 3) a collision when two or more of the active nodes from the remaining $N - 1$ nodes transmit at the same time.

We define $N_{(eqv-1)j}$ as the number of active nodes among the remaining $N - 1$ nodes from the node j 's perspective. Since only one node transmits in a successful transmission, we define $N_{(eqv-2)j}$ as the number of active nodes which do not transmit from the remaining $N - 2$ nodes from the node j 's perspective.

$$N_{(eqv-1)j} = \sum_{k=1; k \neq j}^N (1 - q_{0k}) \quad (4.5)$$

$$N_{(eqv-2)j} = \frac{\sum_{l=1; l \neq j}^N \sum_{k=1; k \neq j; k \neq l}^N (1 - q_{0k})}{N - 1} \quad (4.6)$$

4.4.1 Equal Load in Unsaturated Load Conditions

In the equal load case, all the nodes have the same load and so, $q_{0j} = q_0$ for all $j = 1$ to N . Since each node is empty with probability q_0 , there are only a fraction of nodes which contend for the

channel.

$$N_{eqv} = N \cdot (1 - q_0) \quad (4.7)$$

$$N_{(eqv-1)j} = (N - 1) \cdot (1 - q_0) \quad (4.8)$$

$$N_{(eqv-2)j} = (N - 2) \cdot (1 - q_0) \quad (4.9)$$

4.4.2 Saturated Load Conditions

In saturated load conditions, the queue is never empty for each node and so $q_{0j} = 0$ for all $j = 1$ to N [AB06b].

$$N_{eqv} = N \quad (4.10)$$

$$N_{(eqv-1)j} = (N - 1) \quad (4.11)$$

$$N_{(eqv-2)j} = (N - 2) \quad (4.12)$$

4.4.3 No Retry Limit

Let r_j be the average number of state transition steps a node j observes before it transmits a packet (i.e., not including the step in which the node transmits the packet). Since node j transmits a packet

with probability p , the average number of state transition (back-off) steps r_j before node j transmits a packet is given by

$$r_j = \frac{1-p}{p} \quad (4.13)$$

Since at each transition step, node j observes one of the three events on the channel *viz.*, a successful transmission E_{Sj} , a collision E_{Cj} , or an idle slot E_{Ij} . Let $g_{Sj} = P\{E_{Sj}\}$, $g_{Cj} = P\{E_{Cj}\}$ and $g_{Ij} = P\{E_{Ij}\}$ be the probabilities for the above three events observed by node j . Since these three events are independent and mutually exclusive at each state transition step, the probability that there are r_{Ij} "idle slots", r_{Cj} "collision slots", and r_{Sj} "successful slots" in r_j slots is given by,

$$\begin{aligned} g_{Ij} &= (1-p)^{N_{(eqv-1)j}} \\ g_{Sj} &= N_{(eqv-1)j} \cdot p \cdot (1-p)^{N_{(eqv-2)j}} \\ g_{Cj} &= 1 - g_{Ij} - g_{Sj} \end{aligned} \quad (4.14)$$

Then, $r_{Ij} = r_j \cdot g_{Ij}$, $r_{Sj} = r_j \cdot g_{Sj}$ and $r_{Cj} = r_j \cdot g_{Cj}$. Therefore, the total average back-off time or state transition time (T_{Bj}) spent by the node j before transmitting a packet is,

$$\begin{aligned} T_{Bj} &= \sigma \cdot r_{Ij} + T_C \cdot r_{Cj} + T_S \cdot r_{Sj} \\ &= r_j \cdot [\sigma \cdot g_{Ij} + T_C \cdot g_{Cj} + T_S \cdot g_{Sj}] \end{aligned} \quad (4.15)$$

T_C and T_S are the average time for a collision and successful transmission respectively. The symbol σ denotes the PHY layer slot time parameter. After T_{Bj} , node j transmits a packet and succeeds with probability V_{Sj} or fails with probability $V_{Cj} = 1 - V_{Sj}$. The packet transmission will be successful only when none of the other nodes with non empty queues transmit packets at the same time. Thus,

$$V_{Sj} = (1 - p)^{N_{(eqv-1)j}} \quad (4.16)$$

$$V_{Cj} = 1 - V_{Sj} \quad (4.17)$$

Since p is constant in all transmission stages, V_{Sj} is the same in all the transmission attempts and therefore the average number of transmission attempts ending with a collision S_{Cj} before a packet is successfully sent by node j is given by

$$S_{Cj} = \frac{1 - V_{Sj}}{V_{Sj}} \quad (4.18)$$

Therefore the average service time delay for node j is equal to

$$\bar{T}_j = S_{Cj} \cdot (T_{Bj} + T_C) + (T_{Bj} + T_S) \quad (4.19)$$

4.4.4 Finite Retry Limit

In this section, we extend our analysis on the service time from the previous section to the case of finite retry limit. Let $m - 1$ be the upper bound on the number of packet retransmissions. We have m transmission stages with the node behavior in each stage same as that of the single stage in the "no retry limit" case. Also, the events that occur in a node j at the k^{th} transmission stage are independent of the events that occur in the $(k - 1)^{th}$ stage. Since p remains constant in all stages, the average number of transition slots r_j before a node j transmits a packet is the same as that in the "no retry limit" case in each of the stages 1 to m . Therefore, the average total back-off time or total transition time in each of the m stages is the same as that in the "no retry limit" case.

The probability of success that a packet in node j experiences when it is transmitted at the end of the k^{th} back-off stage is $V_{Sj} = 1 - V_{Cj}$. Let XS_k be the event that the packet is successfully transmitted at the end of k^{th} stage and XD be the event that the packet is discarded at the end of the m^{th} stage.

$$P\{XS_k\}_j = V_{Cj}^{k-1} \cdot V_{Sj} \quad 1 \leq k \leq m \quad (4.20)$$

$$P\{XD\}_j = V_{Cj}^m \quad (4.21)$$

Let $T_{Bj}(k)$ be the total time spent by node j before transmitting a packet successfully in stage k .

Therefore $T_{Bj}(k)$ is given by

$$T_{Bj}(k) = k \cdot T_{Bj} + (k - 1) \cdot T_C \quad 1 \leq k \leq m \quad (4.22)$$

Let T_{kj} be the average service time corresponding to the event XS_k and T_{wj} be the average wasted time corresponding to the event XD for node j . Then we have

$$T_{kj} = T_{Bj}(k) + T_S \quad 1 \leq k \leq m \quad (4.23)$$

$$T_{wj} = T_{Bj}(m) + T_C \quad (4.24)$$

Therefore, the average service time for node j is given by,

$$\begin{aligned} \bar{T}_j &= \sum_{k=1}^m [P\{XS_k\}_j \cdot T_{kj}] + [P\{XD\}_j \cdot T_{wj}] \\ &= (1 - V_{Cj}^m) \cdot \{S_{Cj} \cdot (T_{Bj} + T_C) + (T_{Bj} + T_S)\} \end{aligned} \quad (4.25)$$

4.5 One hop End-to-End Delay

There is a growing demand for QoS applications in WLANS. Most of the QoS based applications require delay guarantees and/or throughput guarantees. The end-to-end delay provides a measure

whether delay guarantees can be provided. A one hop delay in a single hop WLANs is defined as the time interval between the time at which a packet is placed in the queue and the time at which a corresponding ACK is received after transmitting this packet. With increasing emphasis on QoS in WLANs, a thorough study of one hop delay in WLANs helps us identify its limitations and helps in designing an efficient admission control.

The service rate μ_j is the inverse of the average service time of node j . In [Kle75], the probability of finding k packets in the queue for node j with a maximum size of K in M/M/1/K queue is given by

$$q_{kj} = \begin{cases} \left(\frac{(1 - \rho_j)}{(1 - \rho_j^{K+1})} \right) \cdot \rho_j^k & 0 \leq k \leq K \\ = 0 & \text{Otherwise} \end{cases} \quad (4.26)$$

Where, $\rho_j = \frac{\lambda_j}{\mu_j}$. Once the queue is full, the packets arriving at the queue are dropped and as a result the input load rate is not the effective load anymore. Therefore, the average load λ_{avgj} for node j is given by

$$\lambda_{avgj} = (1 - q_{Kj}) \cdot \lambda_j \quad (4.27)$$

The average queue size for node j is given by,

$$Q_{avgj} = \left[\frac{\rho_j}{(1-\rho_j)} \right] - \left\{ \frac{(K+1) \cdot \rho_j^{K+1}}{(1-\rho_j^{K+1})} \right\} \quad (4.28)$$

Thus, the average one hop delay (\bar{D}_j) for node j in unsaturated conditions is given by

$$\bar{D}_j = \frac{Q_{avgj}}{\lambda_{avgj}} \quad (4.29)$$

Under saturated conditions, the queue is always full with a number of packets equal to the size K .

Therefore the one hop delay (\bar{D}_j) is given by

$$\bar{D}_j = K \cdot \bar{T}_j \quad (4.30)$$

However, from the previous analysis, we know that \bar{T}_j is dependent on p , active nodes in N , and q_{0j} for $j = 1$ to N . Also, for a given λ_j , and K , the value of q_{0j} is given by,

$$q_{0j} = \frac{(1-\rho_j)}{(1-\rho_j^K + 1)} \quad (4.31)$$

Where $\rho_j = \frac{\lambda_j}{\mu_j}$, and K is the maximum queue size. Thus, for a given λ_j , we observe that q_{0j} and \bar{T}_j are dependent on each other. To overcome the cyclic dependency, we use an iterative method

to find \bar{T}_j and thereby the one hop delay (\bar{D}_j) . We have observed in our extensive tests that our iterative method converges in three to four iterations.

Algorithm 4.1 end-to-end delay

Input: $q_{0j}, \lambda_j, \forall j 1 \text{ to } N$
Initialize \bar{T}_{init} to 0
repeat
 for $j = 1 \text{ to } N$ **do**
 Calculate \bar{T}_j using q_{0j} ;
 $\mu_j = \frac{1}{\bar{T}_j}$;
 $\rho_j = \frac{\lambda_j}{\mu_j}$;
 $q_{0j} = \frac{(1-\rho_j)}{(1-\rho_j^{K+1})}$;
 end for
 Set \bar{T} to Average of $\bar{T}_j \forall j$;
 $\bar{T}_{diff} = |\bar{T} - \bar{T}_{init}|$;
 $\bar{T}_{init} = \bar{T}$;
until ($\bar{T}_{diff} < 0.00001$)
for $j = 1 \text{ to } N$ **do**
 $Q_{avgj} = \left\{ \frac{\rho_j}{(1-\rho_j)} \right\} - \left\{ \frac{(K+1) \cdot \rho_j^{K+1}}{(1-\rho_j^{K+1})} \right\}$;
 $\lambda_{avgj} = (1 - q_{Kj}) \cdot \lambda_j$;
 $\bar{D}_j = \frac{Q_{avgj}}{\lambda_{avgj}}$;
end for

4.6 Throughput Analysis

In this section, we present an analytical model for the throughput of P-Persistent 802.11 under ideal channel conditions and finite number of terminals. The analysis doesn't consider hidden terminals.

Bianchi presented a similar evaluation on legacy 802.11 MAC using a Markov chain based model

[Bia00]. Bianchi's analysis didn't consider finite retry limits and so, may not provide an accurate model for the legacy 802.11 MAC protocol.

The fraction of time a channel is used to successfully transmit payload bits, called normalized system throughput, gives us a better view about the performance of the P-Persistent 802.11 MAC protocol. Let S be the normalized system throughput, P_{tr} and P_I be the probabilities that there is a transmission and the channel is idle in the considered slot time respectively. Since each of the active nodes from the N stations contend for the channel with transmission probability p in a slot time, we have

$$P_I = (1 - p)^{N_{eqv}} \quad (4.32)$$

$$P_{tr} = 1 - P_I \quad (4.33)$$

Let P_S be the probability for a successful transmission in a random slot given there is a transmission P_{tr} in the random slot, i.e.,

$$P_S = \frac{N_{eqv} \cdot p \cdot (1 - p)^{N_{eqv}-1}}{P_{tr}} \quad (4.34)$$

Similarly, let P_C be the probability that there is a collision in a random slot when there is a transmission P_{tr}

$$P_C = 1 - P_S \quad (4.35)$$

Where, N_{eqv-1} is given by,

$$N_{eqv-1} = \frac{\sum_{j=1}^N N_{(eqv-1)j}}{N} \quad (4.36)$$

Therefore, the normalized system throughput is given by the ratio

$$\begin{aligned} S &= \frac{E[\text{payload transmitted in a slot time}]}{E[\text{length of the slot time}]} \\ &= \frac{P_S \cdot P_{tr} \cdot E[P]}{P_I \cdot \sigma + P_S \cdot P_{tr} \cdot T_S + P_C \cdot P_{tr} \cdot T_C} \end{aligned} \quad (4.37)$$

Where, $E[P]$ is the average packet size, σ is the empty slot time, and T_S and T_C are average times the channel is sensed busy due to a successful transmission and collision, respectively. Eq. 4.37 holds true irrespective of the access mechanism used say, a basic access or RTS/CTS access.

Let the packet header be $H = PHY_{hdr} + MAC_{hdr}$ and the propagation delay be δ . Then the expressions for T_S and T_C for both the access mechanisms is given by

$$T_S^{Basic} = DIFS + H + E[P] + \delta + SIFS + ACK + \delta$$

$$T_C^{Basic} = H + E[P^*] + \delta + EIFS \quad (4.38)$$

$$T_S^{RTS} = DIFS + RTS + \delta + SIFS + CTS + \delta + H + E[P] + \delta + SIFS + ACK + \delta$$

$$T_C^{RTS} = RTS + \delta + EIFS \quad (4.39)$$

Where $E[P^*]$ is the average length of the longest packet payload involved in a collision. In all our cases, all the packets have the same fixed size, therefore, we have $E[P] = E[P^*] = P$. We also assume that in the RTS access mechanism, all collisions occur only between RTS frames. The maximum value for S is obtained by optimizing the parameter p . The analysis for optimum p is the same as given in [Bia00]. The approximate value of p is given by

$$P_{opt} \approx \frac{1}{N \cdot \sqrt{\frac{T_C^*}{2}}} \quad (4.40)$$

4.7 Model Validation

In this section, we carry out the validation of the analytical models for average service time delay, average one hop end-to-end delay, and the throughput of the P-Persistent 802.11 MAC protocol. We show that our models provide a good approximation for all the three metrics.

4.7.1 Simulation Model

To validate our analytical model, we conducted extensive simulation tests using the NS2 simulator [Pro09]. The P-Persistent 802.11 MAC protocol is added to the existing IEEE 802.11 DCF implementation of NS2. We also added a static ARP and dumb routing protocol to the NS2 code. The static ARP doesn't send any ARP packets and the dumb routing protocol does not use any routing mechanism. It just sends the data packets down the stack to the link layer. As a result, there is no overhead due to the routing protocol and ARP. The analytical models were implemented in C++ with the same set of parameters as that used in the NS2 simulator.

In our study, we simulate a single hop WLAN where each station can hear all the other stations. The WLAN operates in the IEEE 802.11a mode with a data transmission rate of 54Mbps. Our simulation scenario consists of a circular topology of N nodes, where node ' i ' sends all its traffic to node ' $i+1$ ', $1 \leq i < N$, and node ' N ' sends its traffic to node ' 1 '. The MAC layer system parameters are set as shown in Table 4.1.

Table 4.1: 802.11 MAC Parameters

tslot	9 μ s
SIFS	16 μ s
DIFS	34 μ s
Data Rate	54 Mbps
PLCP Data Rate	6 Mbps
Preamble Length Bits	144 bits
PLCP Header Length	48 bits
ACK Length Bytes	14 bytes
MAC Header Length Bytes	28 bytes

4.7.2 Correctness of the P-Persistent 802.11 MAC in NS2

In this section, we verify the correctness of the P-Persistent 802.11 MAC implemented in NS2. The standard 802.11 MAC and the P-Persistent 802.11 MAC are similar in all aspects except the back-off interval generation. So, we used the standard 802.11 MAC implementation in the NS2 and extended it to implement the P-Persistent 802.11 MAC using the C++ feature of a derived class in NS2. We can verify the correctness of the P-Persistent 802.11 in NS2 if we can find a close match between the results for the back-off time in the analytical model and the NS2 simulation. The second approach is to see if there is a close match between the normalized throughput from Eq.4.37 and the NS2 simulation results. The closed form for normalized throughput is the same for both the standard 802.11 and P-Persistent 802.11. In this section, we use the second approach to verify the correctness of the model. We use saturated load conditions as described in section 4.7.3.

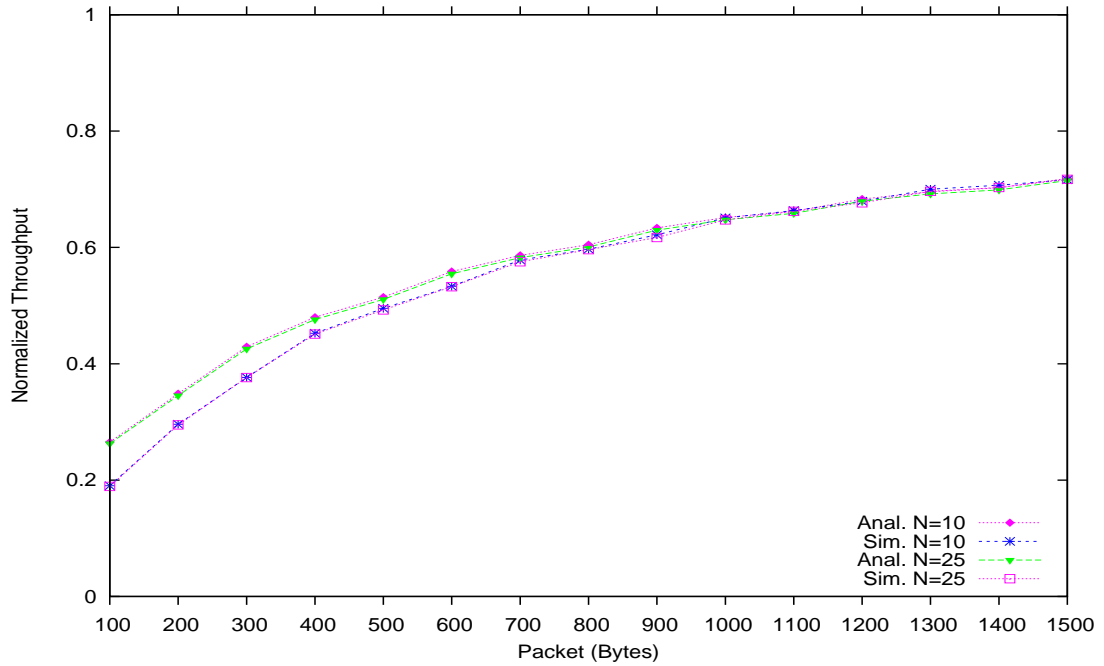


Figure 4.3: Correctness of the P-Persistent 802.11 for variable packets under saturated load.

In the first experiment, we varied the packets from 100 bytes to 1500 bytes for $N=10$ and $N=25$. Fig. 4.3 shows a close match for the normalized throughput for both $N=10$ and $N=25$. The difference between the analytical and the simulation results decreases with the increase in packet size. Overall, the simulation results closely follow the trend in the analytical results.

In the second experiment, we varied the network size from $N=10$ to 100 for packet sizes of $P=300, 600, 900, 1200$ and 1500. Fig. 4.4 shows that the simulation results follow very closely the trend in the analytical results. The difference between the analytical and simulation results decreased with the increase in packet size. Since the error in the results is very small (less than 2%), we can conclude that our NS2 implementation of the P-Persistent 802.11 MAC protocol is

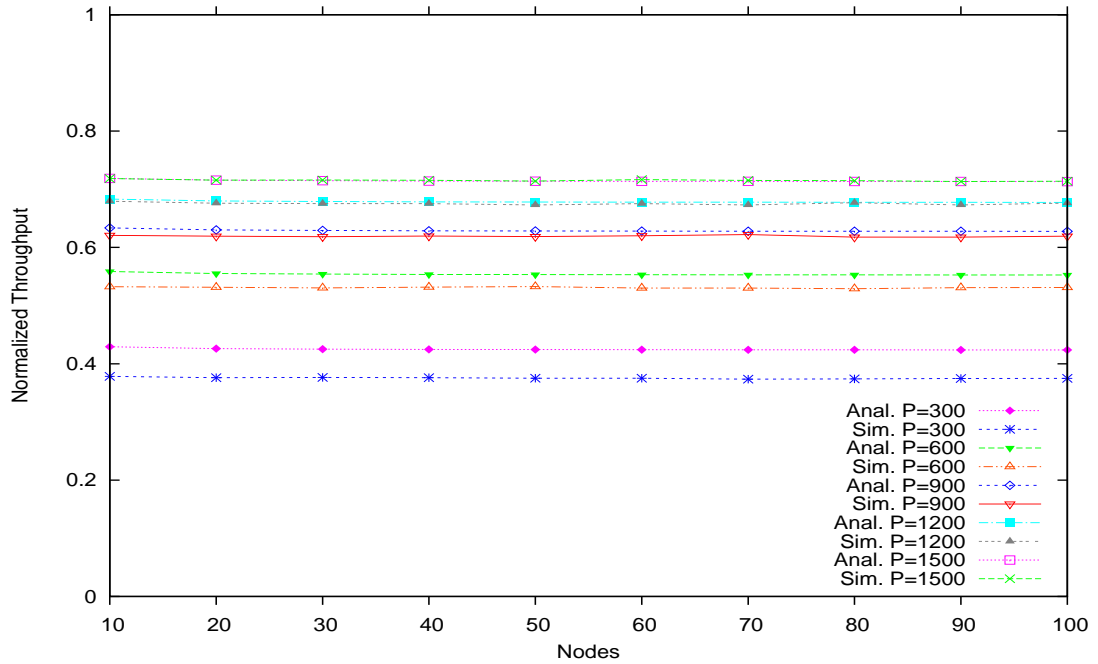


Figure 4.4: Correctness of the P-Persistent 802.11 for variable nodes under saturated load.

correct and can be used to validate the analytical models we developed for the P-Persistent 802.11 MAC protocol.

4.7.3 Saturated Load

In the saturated load case, the simulations are run with the total number of nodes varied from 10 to 100 and the data packet sizes varied from 100 to 1500 bytes. We ran each simulation for 5 seconds and activated all the traffic sources at $t = 0.000000001$ seconds and deactivated them at $t = 5$ seconds. In each station, the exponential traffic is generated at a rate where a queue size of 5

packets is never empty. For example, when we used a load rate of 25 Mbps in each node, the queue was never empty for all the packet sizes.

We evaluated the performance of our model with the simulation results for both the no retry limit case and the finite retry case (using the basic transmission mode without RTS/CTS). In the finite retry case, we set the retry limit to 1. In the first experiment, we varied the nodes from 10 to 100 for packets sizes of 300 and 600 bytes respectively. This was done to verify the correctness of the model over wide range of nodes for a given packet size. Figs. 4.5 and 4.6 show that the simulated results for both the service time delay and the end-to-end delay are almost identical to our analytical models for both cases of no retry limit and finite retries. We conducted a second

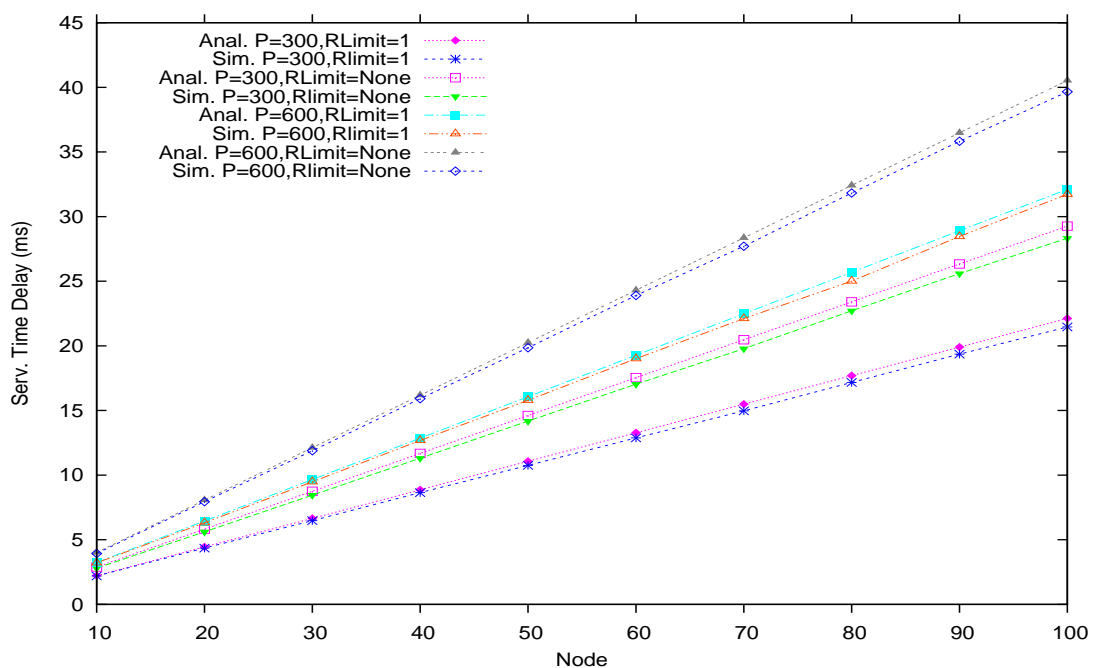


Figure 4.5: Validation of service time delay for variable nodes under saturated load.

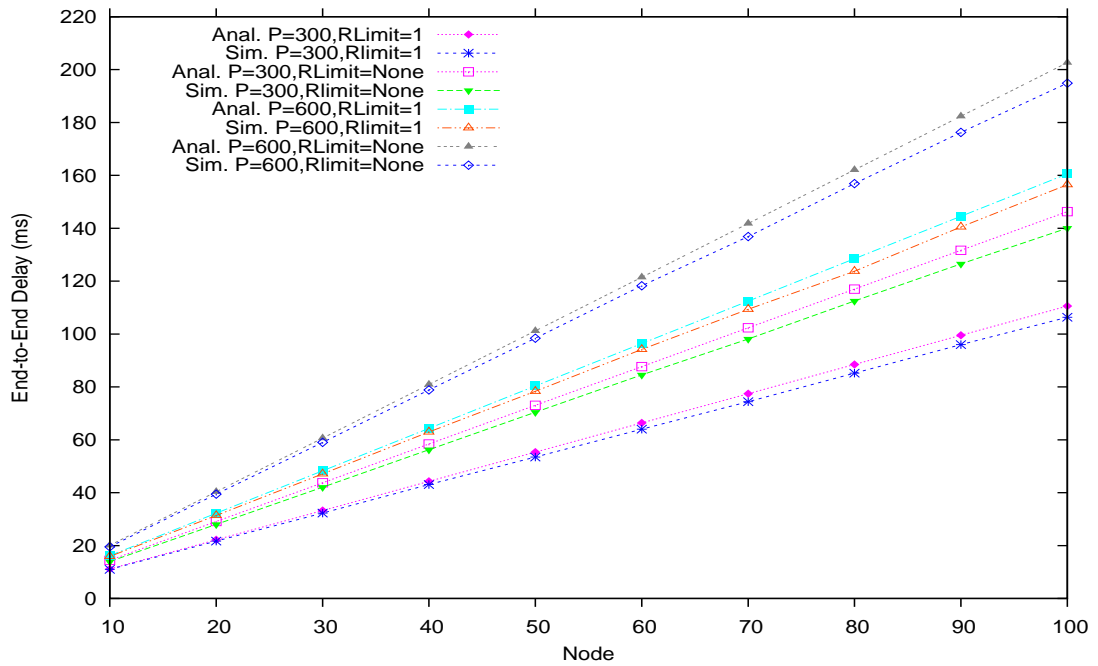


Figure 4.6: Validation of end-to-end delay for variable nodes under saturated load.

experiment in which we varied the packets from 100 to 1500 bytes for network sizes of 10 and 25 respectively. In this experiment, we wanted to verify the correctness of our model over wide range of packets for a give network size. Our simulation results from the Figs. 4.7 and 4.8 show that they are a close match for both retry and no retry limit cases. Therefore, we conclude that our model for saturated load is very accurate.

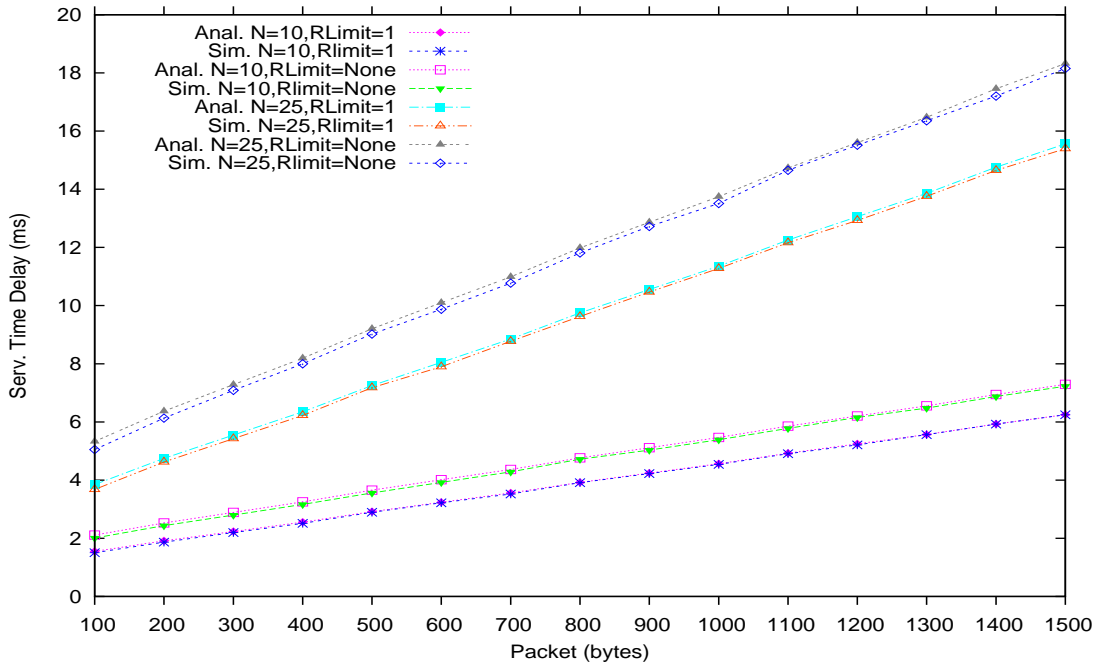


Figure 4.7: Validation of service time delay for variable packets under saturated load.

4.7.4 Unsaturated System with Equal Load

In the case of unsaturated load with equal load, we ran simulations for different network sizes varying from 10 to 100 and packet sizes varying from 100 to 1500. However, we present results only for network size of 10 and 25 nodes and packet sizes of 300 and 600 bytes. To verify our analytical models, we ran each simulation for 5 seconds and activated all the traffic sources at $t = 0.000000001$ seconds and deactivated them at $t = 5$ seconds. The system load is equally divided among all the given nodes. For instance when 10 nodes are used, the exponential traffic is generated

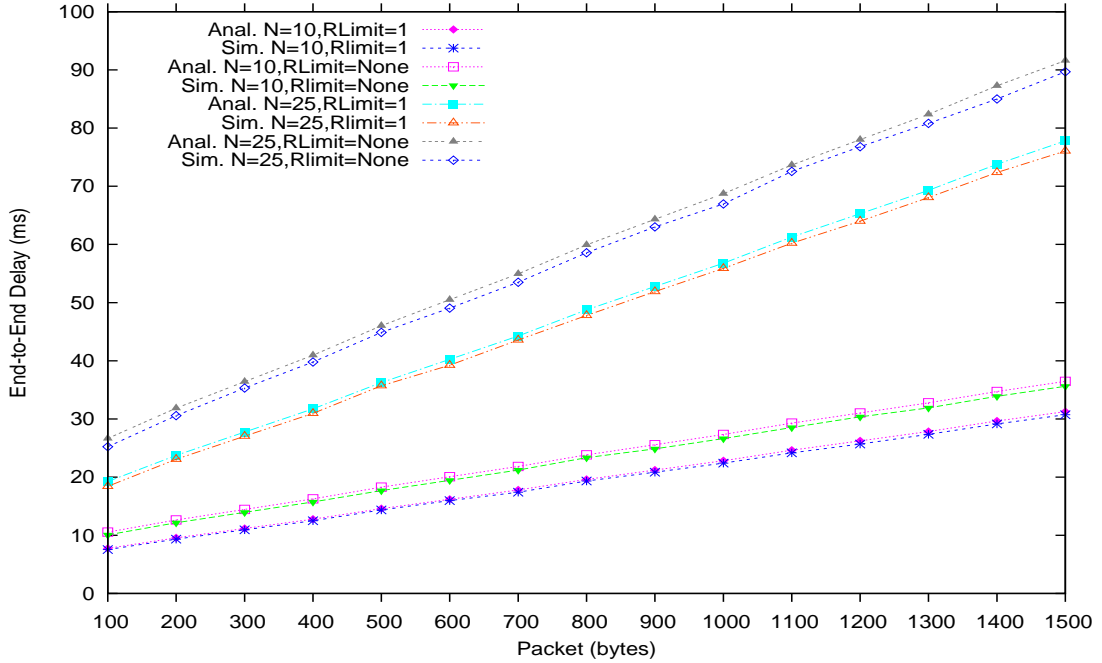


Figure 4.8: Validation of end-to-end delay for variable packets under saturated load.

with a load ranging from 0.5 Mbps to 10 Mbps in each node which is equivalent to a system load of 5Mbps to 100 Mbps. We used single class traffic in our validation.

We validate our model in terms of service time delay and end-to-end delay by varying amount of the system load. In this section, we validate only the retry limit case. Though we conducted simulations for limits of 1 to 7, we present results for a retry limit of 7. In Figs. 4.9 and 4.10, we evaluated the results for $N = 10$ and 25 for $P = 300$ bytes. The simulation results are a very close match to our analytical results. In fact, there are within 1% error. In Figs. 4.11 and 4.12, we evaluated the results for $N = 10$ and 25 for $P = 600$ bytes. The simulation results are a very close match to both our analytical service time delay and end-to-end delay. Therefore, we conclude with

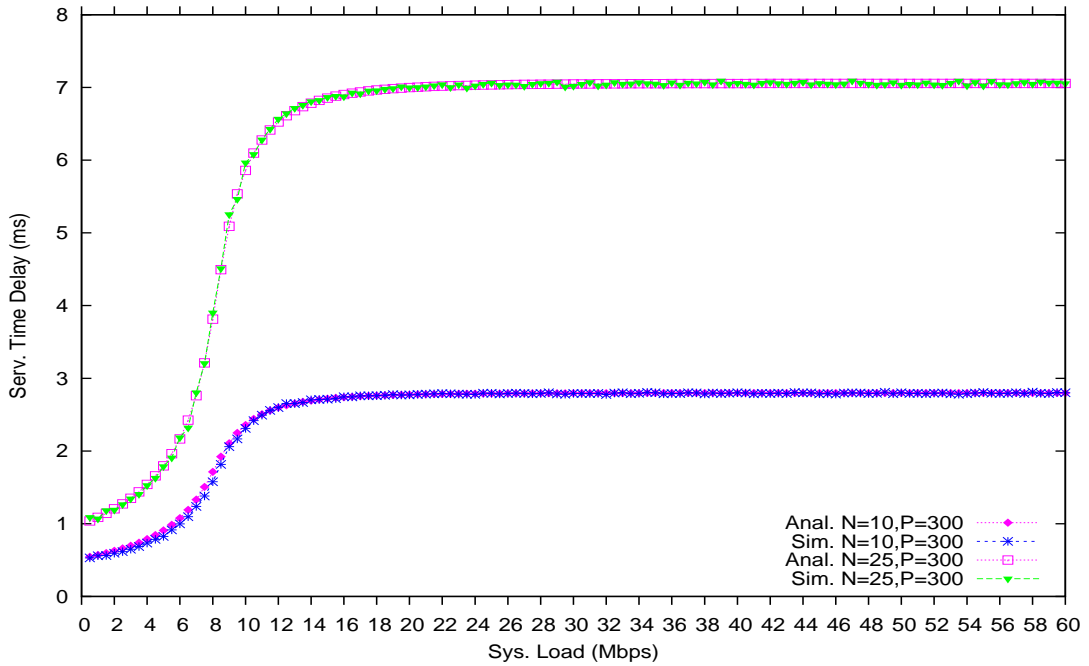


Figure 4.9: Validation of service time delay in an unsaturated system with eqload for $P=300$.

confidence that our analytical model accurately describes the P-Persistent 802.11 MAC protocol in unsaturated system with equal load among the nodes.

4.7.5 Unsaturated System with Unequal Load

In the case of unsaturated load with unequal load among the nodes, the simulations are run for 5, 12, 30 and 48 nodes with data packet sizes of 100, 300, 600, 900, 1200 and 1500 bytes respectively.

In order to evaluate the unequal load condition, we ran two different experiments. In the first experiment we used a network of 5 nodes and the total system load is divided among these 5 nodes

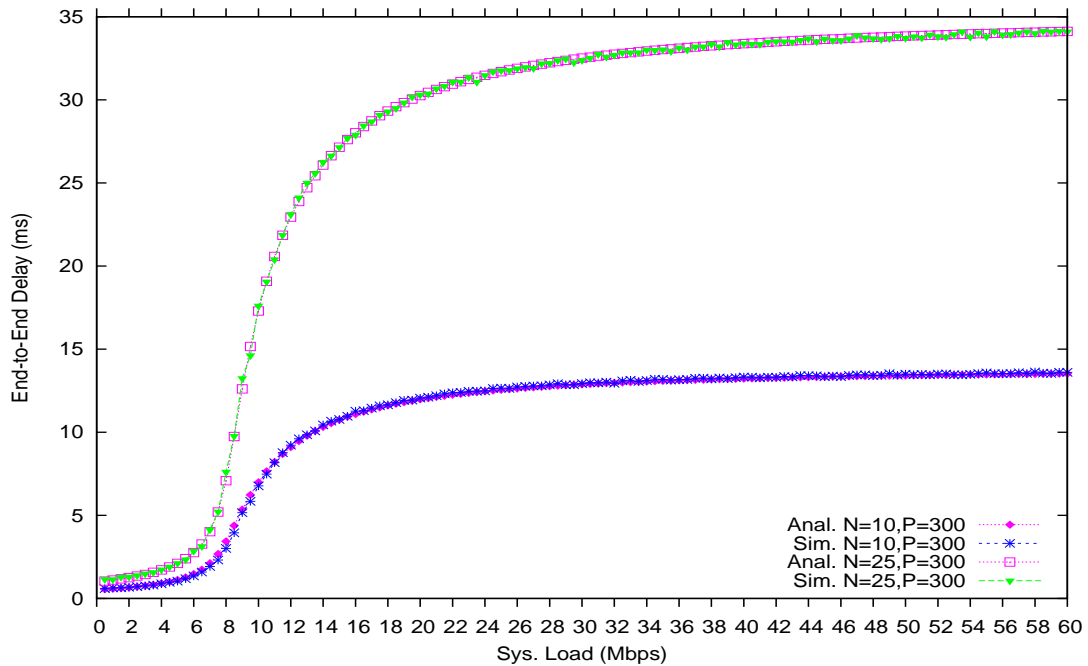


Figure 4.10: Validation of end-to-end delay in an unsaturated system with eqload for $P=300$.

in the ratio of 10:15:20:25:30. That way, each node has a different load from the other nodes. Since it was difficult to come up with different load for every node for a larger network size, we divided the total nodes into three groups and the system load is allocated equally among each group of nodes in the ratio of 50:30:20 in our second experiment. For instance, in the case of 12 nodes and a system load of 10Mbps, four nodes are allocated fifty percent of the 10Mbps (1.25Mbps each), the second four nodes are allocated thirty percent of the 10Mbps (0.75Mbps each) and the last four nodes are allocated twenty percent of the 10Mbps (0.5Mbps each).

We ran each simulation run for 5 seconds and took an average of five runs. The Poisson traffic is activated at $t = 0.000000001$ seconds and deactivated them at $t = 5$ seconds. We present our results

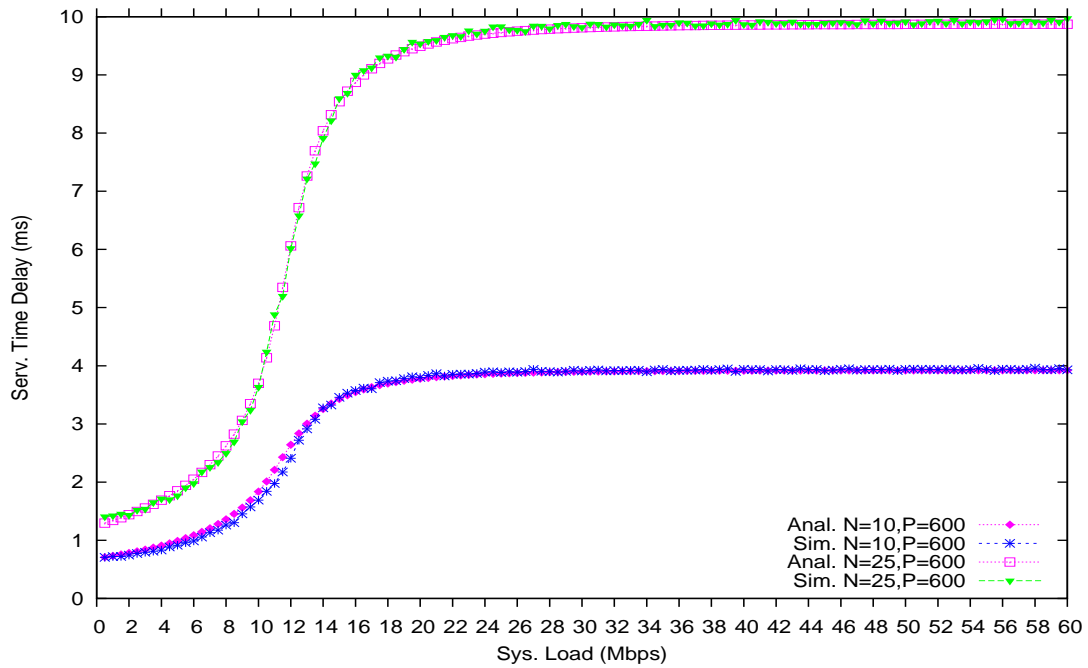


Figure 4.11: Validation of service time delay in an unsaturated system with eqload for $P=600$.

for $N=5$, and $N=30$ and for $P=300$ bytes and 600 bytes. The results for all the other node sizes and packet sizes gave similar performance results. Figs. 4.13 and 4.14 gave a close match to our analytical results in unsaturated system with unequal load distribution among the nodes for $P=300$ bytes. Similarly, we observed a close match for $P=600$ bytes in Figs. 4.15 and 4.16. Therefore, we conclude that our analytical model accurately describes the P-Persistent 802.11 MAC protocol irrespective of the load and its distribution among the nodes.

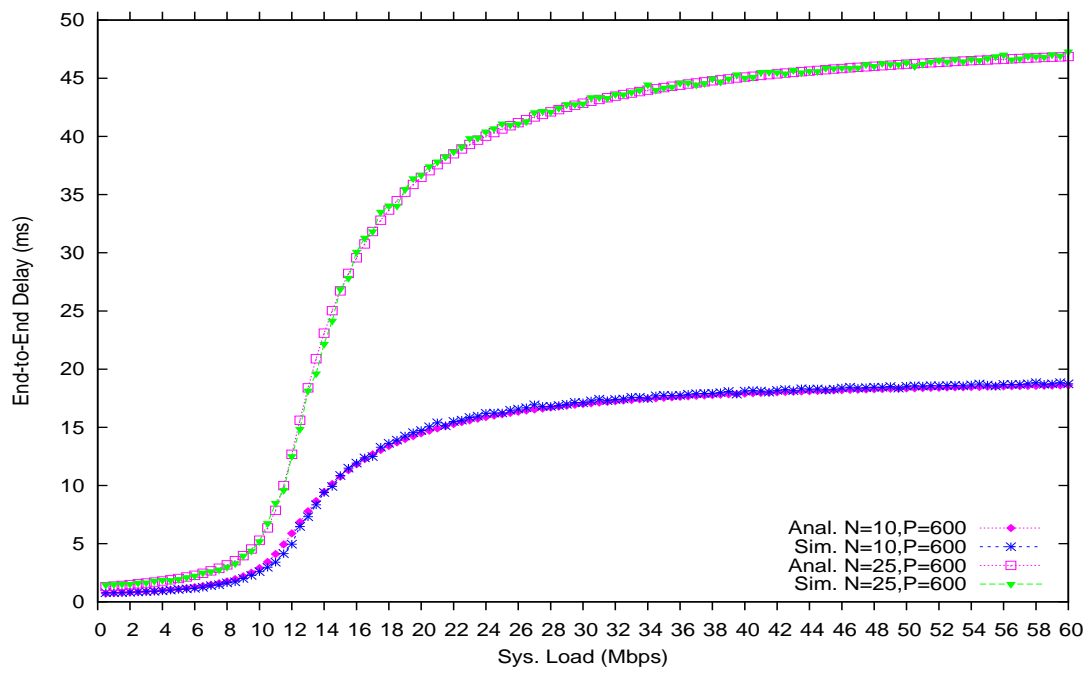


Figure 4.12: Validation of end-to-end delay in an unsaturated system with eqload for P= 600.

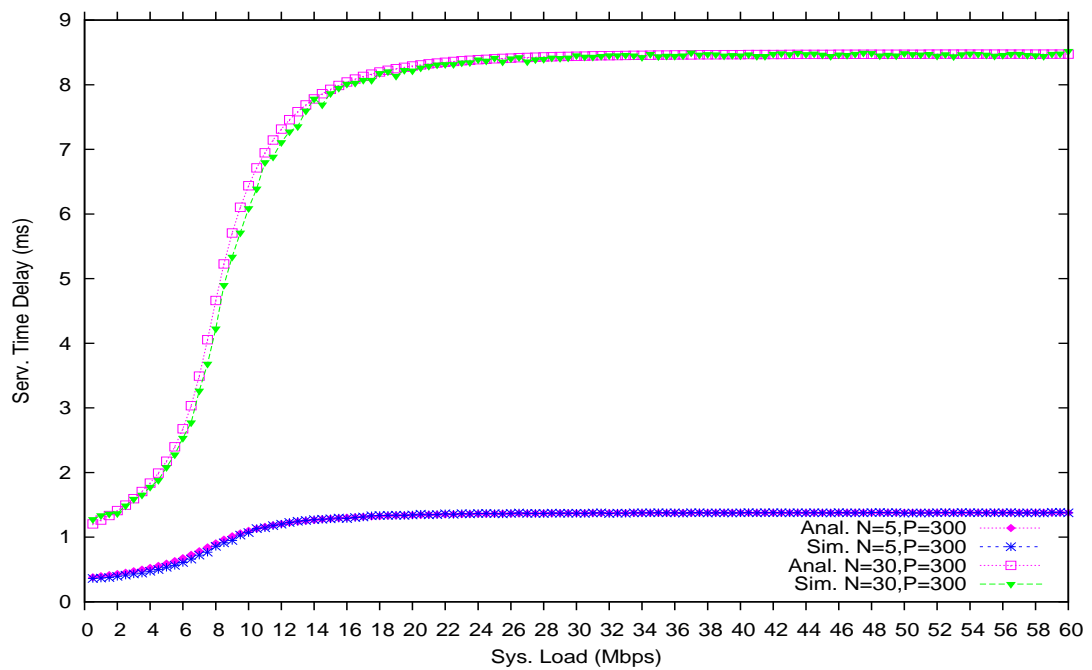


Figure 4.13: Validation of service time delay in an unsaturated system with uneqloud for P= 300.

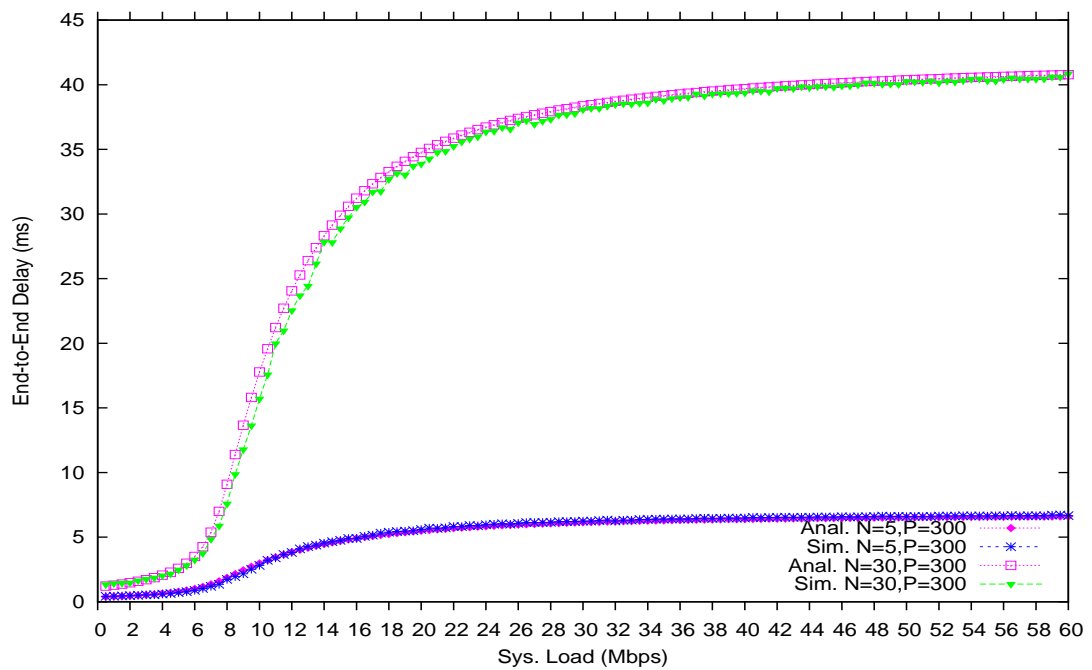


Figure 4.14: Validation of end-to-end delay in an unsaturated system with uneqlod for P= 300.

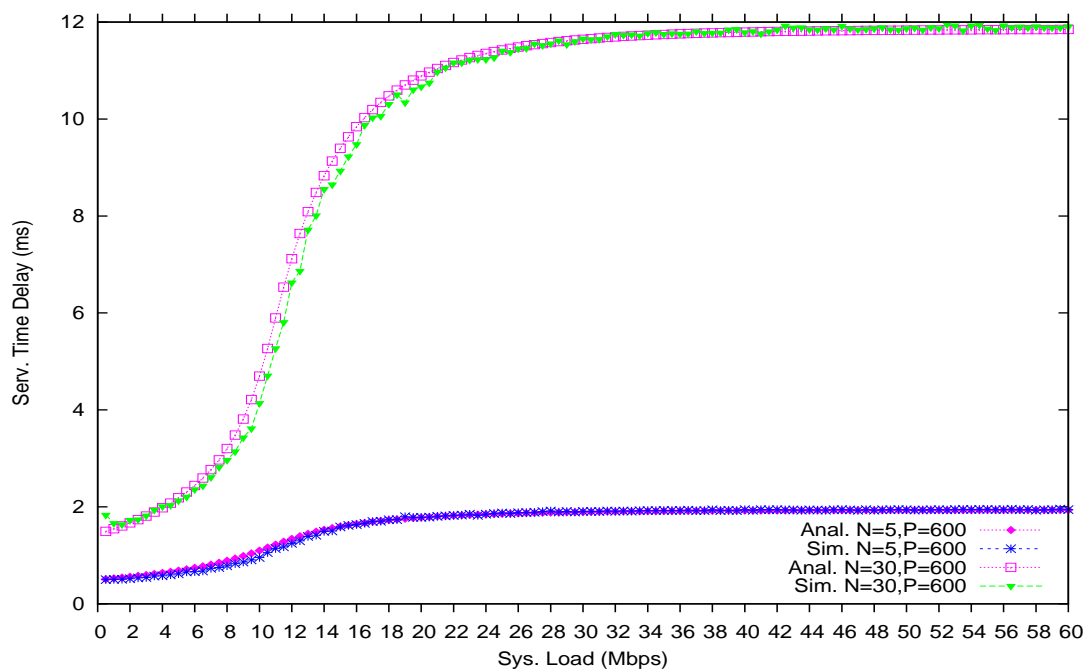


Figure 4.15: Validation of service time delay in an unsaturated system with uneqloud for P = 600.

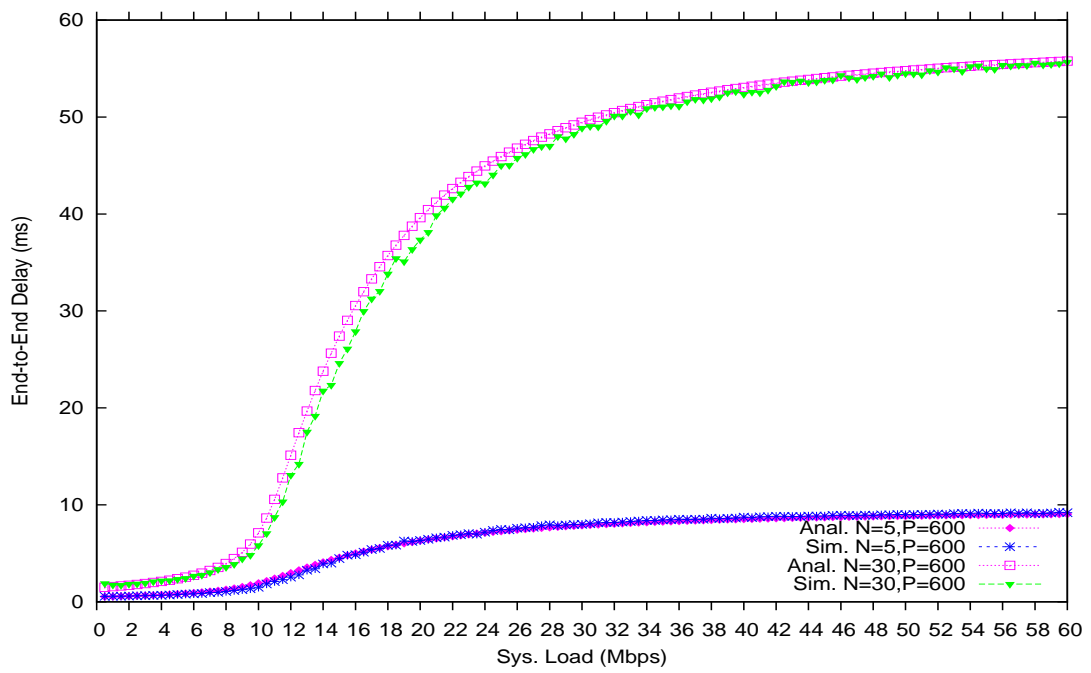


Figure 4.16: Validation of end-to-end delay in an unsaturated system with uneqload for P = 600.

CHAPTER 5 : ANALYSIS AND MODELING OF THE SATURATION THROUGHPUT IN 802.11 WLANS

5.1 Introduction

There are several research efforts in the literature to study the performance of legacy 802.11 and 802.11e [CCG00b,Bia98,Bia00,WPK02,ZA02,CBV03,GK04,EV05,AB06a,AB08b]. These performance studies were done not only by simulations but also by analytical modeling to understand the behavior of these protocols and predict their performance metrics. Bianchi's seminal papers [Bia98, Bia00] introduced a Markov model of saturated WLANs employing 802.11. Most of the models found in the literature are based on Bianchi's Markov model [WPK02, ZA02, CBV03, GK04, EV05].

Evaluation of the P-Persistent 802.11 MAC protocol was also done [AB06a, AB08b]. In this chapter, we study the performance evaluation of the P-Persistent 802.11 in terms of the saturation throughput metric. Particularly, we look into the load conditions at which the throughput attains saturation and also study contention among the nodes at this saturation throughput [AB10].

The rest of the chapter is organized as follows. In section 5.2, we explain the saturation throughput and present the analysis for the P-Persistent 802.11. We validate our findings through simulation results presented in section 5.3.

5.2 Saturation Throughput

In this section, we look into the *Saturation Throughput* for the P-Persistent 802.11 protocol. The saturation throughput is defined as the limit reached by the system throughput as the offered load increases, and represents the maximum load that the system can carry in stable conditions [Bia00]. It is one of the fundamental parameters used in the performance analysis of the 802.11 MAC protocol.

In [Bia00], Bianchi studied the throughput performance of 802.11. The results showed that as the system offered load increases, the throughput of the 802.11 protocol also increases until it reaches a maximum value, defined as *maximum throughput*. When the system offered load is further increased, the throughput drops to lower values and it finally stabilizes to a constant throughput called the *saturation throughput*.

We conducted experiments for the throughput performance of P-Persistent 802.11 similar to that presented in [Bia00] for the standard 802.11. The system load is increased in increments of 1.0 Mbps at every 20 seconds and we call this load the strict-linear load. We also generated another

system load, called the zigzag load, which closely follows the strict-linear load but zigzags above and below it. Both the system load and system throughput are measured every 20 seconds and are normalized with respect to the channel data transmission rate. The simulation parameters for a WLAN of size 10 and 20 stations are given in Table 5.1 and the simulation models and results are described in Section 5.3.

Fig. 5.1 shows that the throughput corresponding to the strict-linear system load closely followed the system load for the first 220 seconds of the simulation and remained constant at the normalized value of 0.19 for the remaining simulation time. For the zigzag load, we also find that the throughput closely followed the zigzag load till the first 220 seconds and then remained constant at 0.19 irrespective of the increased value of the zigzag load. Our results show that the maximum throughput for the P-Persistent 802.11 protocol always coincides with the saturation throughput. Specifically, there is no peak value followed by a drop in the throughput as in the case of the standard 802.11 presented in [Bia00]. Therefore, the beginning of the saturation throughput is both the maximum throughput and the stable saturation throughput achieved under overloaded conditions. Based on this observation, we define the system load which corresponds to the beginning of the stable saturation throughput as the initial saturation load. Any system load value higher than the initial saturation load is simply called the saturation load and gives the same saturation throughput. Once the system reaches the saturation throughput, any increase in the offered load leads to increasing the queue buildup. We observed that the capacity (size) of the buffer used for queuing packets has an influence on the value of the initial saturation load. In Fig. 5.2, we studied

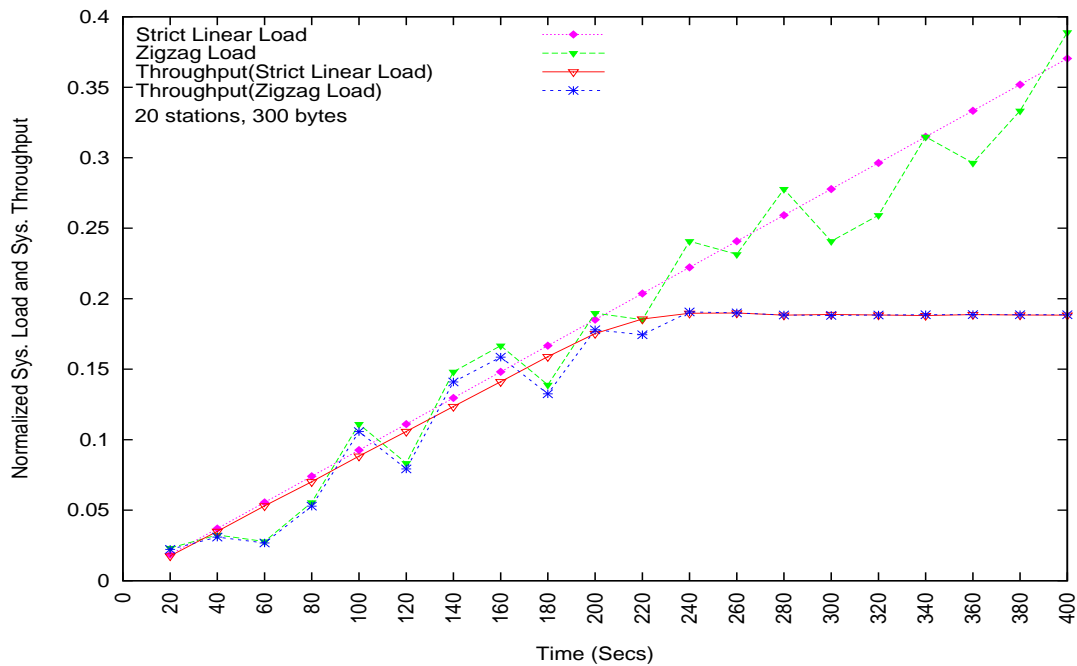


Figure 5.1: Throughput of P-Persistent 802.11.

the throughput for queue lengths of 2, 3, 4 and 5, respectively. The results show that the bigger the queue size, the lower the initial saturation load.

The above behavior can be explained as follows. At a smaller buffer size, the queue gets full quickly at lower loads and incoming packets are dropped. The buffered packets eventually get transmitted leaving an empty or partially empty queue, which takes more time to fill as the inter-arrival time between packets at this low load is relatively large. Thus, even though the queue gets full, the low load is not sufficient to make the system saturated. So, a higher load is needed to continuously fill the packets when the earlier packets are removed from the queue. This results in a higher initial saturation load for buffers with small sizes.

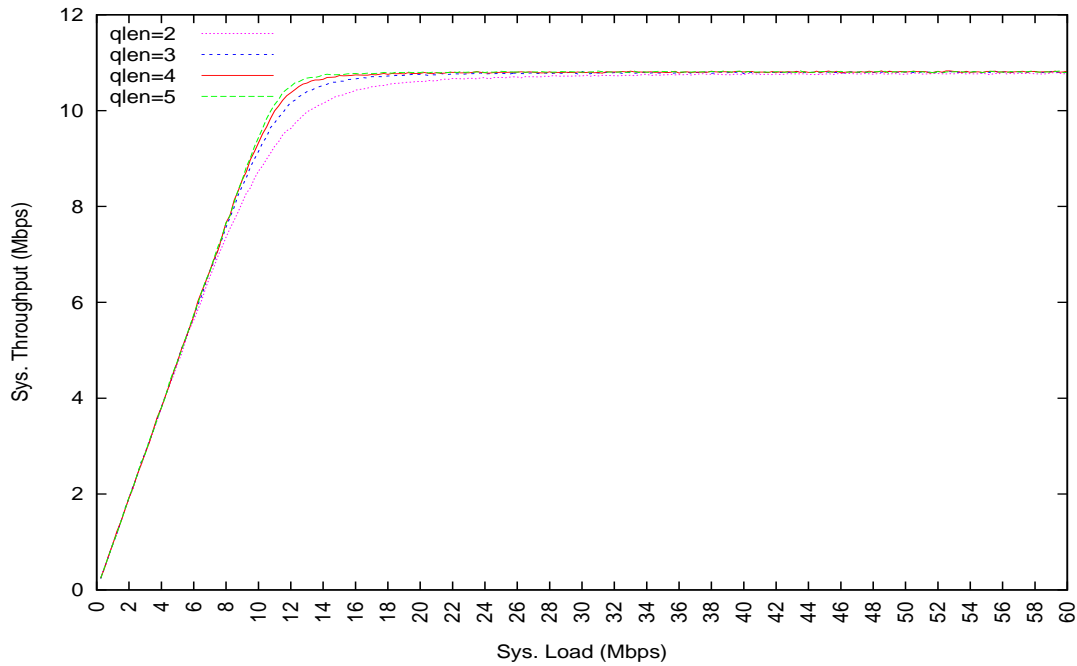


Figure 5.2: System throughput for different queue sizes.

5.3 Results and Analysis

Analytical performance studies of 802.11 have assumed that the saturation throughput is calculated under overload conditions. Under these conditions, every node is contending for transmission, i.e., whenever the MAC layer of a node successfully transmits a data frame, it immediately gets another frame ready for transmission. In [Bia00], Bianchi defined the analytical model for saturation throughput in terms of normalized system throughput S . It is defined as the fraction of time the channel is used to successfully transmit payload bits. The normalized saturated throughput S is expressed in the following equation:

$$S = \frac{P_S \cdot P_{tr} \cdot E[P]}{(1 - P_{tr}) \cdot \sigma + P_{tr} \cdot P_S \cdot T_S + P_{tr} \cdot (1 - P_S) \cdot T_C} \quad (5.1)$$

Where, P_{tr} is the probability that there is at least one transmission in the considered slot time; P_S is the probability that a transmission occurring on the channel is successful given there is at least one transmission; $E[P]$ is the average packet size; T_S is the average time the channel is sensed busy due to the successful transmission and T_C is the average time the channel is sensed busy by each station during collision. Also, the slot time is defined as the time duration between two consecutive successful transmissions. The assumption of the overloaded condition for the above equation is that every node in the network has a packet contending for transmission. However, our analysis shows that the saturation throughput for the P-Persistent 802.11 can be reached well before this extreme overloaded condition. Specifically, the 802.11 wireless LAN can attain a saturation throughput for an overloaded system when some of its nodes are idle.

To validate our claim, we conducted extensive simulation tests using the NS2 simulator [13-14]. In our study, we simulated a single hop WLAN where each station can hear all the other stations (i.e., the hidden node problem does not exist) and the WLAN operates in the IEEE 802.11a mode with a data transmission rate of 54Mbps. In our tests, the Poisson traffic is generated with a system load varying from 0.25 to 60 Mbps. For each system load, we ran the simulation for 5 seconds activating all the traffic sources at $t = 0.00001$ seconds and deactivating them at $t = 5$ seconds. The

results are collected by taking an average of 5 simulation runs. We studied the results for two data packet sizes of 300 and 600 bytes and for two network sizes of N=10 and N=20 nodes. The MAC layer system parameters used in our simulations are set as shown in Table 5.1.

Table 5.1: 802.11 MAC Parameters

tslot	9 μ s
SIFS	16 μ s
DIFS	34 μ s
Data Rate	54 Mbps
PLCP Data Rate	6 Mbps
Preamble Length Bits	144 bits
PLCP Header Length	48 bits
ACK Length Bytes	14 bytes
MAC Header Length Bytes	28 bytes

We use two performance metrics in our analysis, *viz.*, *Fraction of Contention Nodes* and *Frequency of Fraction of Contention Nodes*. The fraction of contention nodes is the percentage of the nodes contending for the wireless medium. For example, if 7 nodes have packets and 3 nodes are idle in a 10-node WLAN, then the fraction of contention nodes is 0.7 (or 70%). We measured the fraction of contention nodes every 5 milliseconds and therefore collected 1000 data points during a simulation test of 5 seconds. The frequency of fraction of contention nodes is the percentage of occurrences of a given fraction of contention nodes within the 1000 collected data points. These two parameters capture a good picture of the medium contention in a wireless LAN as discussed below.

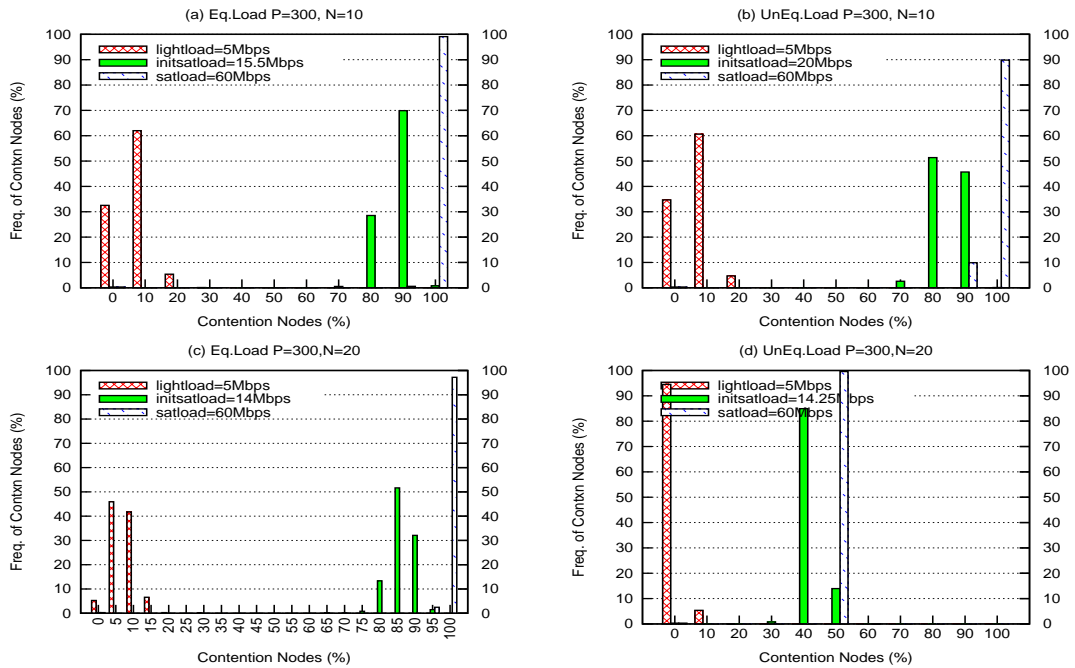


Figure 5.3: Contention Nodes P=300 bytes.

The performance results obtained for packet sizes of 300 and 600 bytes are similar in all respects and we will therefore present the results for packet size of 300 bytes only. In Fig. 5.3(a), the wireless LAN has 10 nodes and the system load is equally divided among these 10 nodes. For example, a load of 0.5 Mbps is assigned to each node when the total system load is 5 Mbps. In Fig. 5.3(b), we distribute the system load unequally among the ten nodes. Specifically, we used three load levels of $0.5 * L$, L , $1.5 * L$ where L is the total system load divided by the number of nodes. Each of these three load levels is assigned to the nodes in the ratio of 2:1:2. For example, when we apply a system load of 5 Mbps, the value of L is 0.5 Mbps and 4 nodes are allocated 0.25 Mbps, 2 nodes are allocated 0.5 Mbps, and the remaining 4 nodes are allocated 0.75 Mbps.

In Fig. 5.3(c), the wireless LAN has 20 nodes and we distribute the system load equally among these 20 nodes. In Fig. 5.3(d), the wireless LAN has 20 nodes but the load distribution is slightly different. We distribute a constant load of 0.5 Mbps equally among 10 nodes out of the 20 nodes. The remaining 10 nodes are assigned load equally from the remainder of the total system load. For example, if the total system load is 5 Mbps, we assign a load of 0.05 Mbps to each one of 10 nodes and assign 0.45 Mbps to each of the remaining 10 nodes.

We now discuss the results shown in Fig. 5.3. In Fig. 5.3(a), at a system load of 5 Mbps, we observe that there is no contention for the channel thirty five percent of the time, 10% (1 node) contention for sixty percent of the time, and 20% (2 nodes) contention for five percent of the time. Fig. 5.3(b) also shows almost a similar behavior for the light load of 5 Mbps, i.e., there is hardly any contention for the wireless medium. Similarly, we observe the same phenomenon in Figs. 5.3(c) and 5.3(d) for the light load of 5 Mbps.

Next, we consider the initial saturation system load which corresponds to the beginning of the stable saturation throughput. Load levels of 15.5, 20, 14, and 14.25 Mbps represent the initial saturation load for Figs. 5.3(a), 5.3(b), 5.3(c) and 5.3(d), respectively. In Fig. 5.3(a), we observed that there is 80% (8 nodes) contention 30% of the time and 90% (9 nodes) contention 70% of the time. Thus, at the initial saturation load, the system does not have all nodes contend.

We now examine the unequal load case in Fig. 5.3(b). We observed that there is 70% (7 nodes) contention five percent of the time, 80% (8 nodes) contention fifty five percent of the time and 90%

(9 nodes) contention forty five percent of the time. We observe that it needed 20 Mbps to reach the initial system saturation load in Fig. 5.3(b) compared to 15.5 Mbps in the equal load case of Fig. 5.3(a). Thus, the initial system saturation load of the network depends on the load distribution among the nodes in the system. It is clear that reaching the initial system saturation load does not require that every node needs to have a packet contending for transmission.

In the case of 20 nodes in Fig. 5.3(c), the system exhibits a similar behavior with 85% and 90% contention amounting to the major chunk of the time. Finally, we examine Fig. 5.3(d). As mentioned earlier, one half of the 20 nodes are assigned a collective small amount of 0.5 Mbps. Due to this, we observe that at the initial saturation load there is 40% (8 nodes) contention for eighty percent of the time and 50% (10 nodes) contention for twenty percent of the time. This confirms again that saturation can be arrived in the wireless medium without all the nodes in contention for the medium. This is further supported by the contention results at a higher system load of 60 Mbps which is way higher than the initial saturation loads in the four graphs of Fig. 5.3. The results show that even at such a high load, there is no 100% contention all the time.

CHAPTER 6 : ADMISSION CONTROL FOR P-PERSISTENT 802.11

6.1 Introduction

Internet has become a significant part of people's lives due to its reliable functionality and availability. In addition, the demand for wireless connectivity has grown rapidly resulting in IEEE 802.11 wireless local area networks (WLANs) being successfully deployed in many public and commercial areas [HER09]. It is envisioned that further growth of WLAN usage will take place and that they will continue to have a major impact on society [SV07].

Exciting new applications and networked services are putting great demand on Next Generation Networking (NGN). With the increasing popularity of multimedia applications, such as voice and video, these networks are then required to support this new emerging multimedia traffic. These multimedia applications have quality of service (QoS) needs typically expressed in terms of the maximum allowed delay and/or the minimum required bandwidth (throughput). Therefore, the network must make sure that it can meet the QoS requirements of a multimedia application before accepting it [HER09].

Bandwidth reservations may not always be implemented at the access networks to aid QoS support. This is especially the case with IEEE 802.11 WLANs since they are initially designed to

provide accessibility to all forms of best-effort Internet data, such as web browsing, e-mailing, and file transferring. For this reason end users of WLANs would not experience predictable network performance [SV07].

Moreover, since DCF is a contention-based protocol, it suffers from collisions in the shared channel, which leads to poor resource utilization. Several research efforts have investigated various approaches to enhance the throughput of DCF by controlling the number of collisions and idle slots. One common approach is to dynamically tune one of the Channel Access Parameters (CAPs)-the contention window size. However, this approach relies on accurate estimation of the number of active stations and the distribution of frame lengths at run time, which may be quite difficult to realize in practice [ZZ04].

The goal of any QoS support is to provide applications with guarantees in terms of bandwidth, delay, or jitter. To provide such guarantees in a networked environment, the MAC layer is responsible for resource allocation at individual nodes, while the network layer must consider resources along the entire route of communication [YK05]. When the network becomes heavily loaded, it becomes less capable of satisfying the QoS requirements of time bounded multimedia traffic due to increased congestion. This results in the need for an effective admission control scheme in 802.11 WLANs.

Admission control is a mechanism deciding if a new flow can be admitted to the network without degrading the performance of already admitted flows, while also satisfying its own ser-

vice quality requirements. Without admission control, there is no protection against congestion in the network, because the new requests may always be accepted into the network regardless of the available resources. However, it is difficult to quantify the available resources within the 802.11 WLANs due to the probabilistic nature of the contention based channel access mechanism. Considering the error-prone, time-varying characteristics of wireless channels, providing hard QoS guarantee in IEEE 802.11 WLAN is an *impossible task*. Even under the assumption of interference-free, providing QoS guarantees while simultaneously achieving optimal channel utilization is still challenging [ZZ04].

The admission control in IEEE 802.11 differs from that in Ethernet due to the characteristics of WLANs. While most of the admission control algorithms for wired networks are based on the end-to-end QoS, admission control in wireless networks should mainly protect the QoS of flows between the AP and clients in a BSS. Therefore, most of the legacy admission control algorithms for wired networks are not applicable in WLANs. Furthermore, the admission decision in wireless networks is more difficult than that in wired networks. Even though we can estimate the delay using the queue size of the AP, we cannot wait until the flow is actually admitted, as this would interfere with the QoS of existing flows. Thus, we need to know the effect of new flows before we admit them [SS08].

There are two main approaches for admission control in 802.11: measurement-based and model-based admission control [GCN05, BSE07]. Measurement-based admission control algorithms use measurements of some network parameters to decide if a flow should be accepted

[XL04a, XL04b, XLC04, GZ03]. The model-based admission control approach deploys an analytical Markov model (e.g., Bianchi's model and its variants [PM03, KLW03, KTB04, CSS05]) to determine if flows should be admitted. In this chapter, we develop a measurement-aided model-based admission control to provide statistical QoS guarantees in both the distributed and centralized scenarios in the contention period of IEEE 802.11. Our admission control uses the analytical model for the P-Persistent 802.11 developed in Chapter 4. It also uses the measured values of the collision probability, service time, and the probability of finding an empty queue, q_0 in the decision process.

The rest of the chapter is organized as follows. In section 6.2, we give a literature review of admission control methods in WLANs. We present two load estimations schemes for our distributed admission control in section 6.3. In sections 6.4 and 6.5, we present our distributed and centralized admission control methods, respectively. We evaluate the performance of our admission control methods in section 6.6.

6.2 Related Work

Several admission control schemes were developed for the 802.11 MAC protocol [VL02, BSE07, KTB04, EV05, ZCF06, PM03, PA07, BKT08, SV07]. These schemes can be classified into three

categories: model-based, measurement-based and hybrid measurement-based and model-based schemes [LM09]. Most of the schemes use throughput as the admission criteria instead of delay.

Valaee developed a distributed admission control based on a universal service curve which is distributed at the time of registration [VL02]. The universal service curve is different for different network sizes and packet distribution. Similarly, Bai [BSE07] developed a model based admission control which is based on delay instead of throughput. It uses regression equations in the calculation of the transmission probability τ which varies with each scenario. The newly calculated τ along with the total offered load is used to find ρ . The call is admitted based on the value of ρ .

Pong developed a centralized admission control for EDCA of 802.11e where he predicts the new throughput of each existing flow when a new flow is admitted. He also suggests optimum CW_{min} values for efficient use of the system. The admission decision for a flow is still based on throughput as the admission criterion [PM03].

Hamdaoui et al. [HER09] developed a delay based centralized admission control in EDCA of IEEE 802.11e. They developed an analytical model using G/G/1 system to estimate the delays. The estimated delays are then used in the admission control at the AP to make an admission decision. This work is one of the few works which employed delay as an admission criterion. Yasukawa developed a distributed delay based admission control for 802.11 WLANs [YFS07]. His work is based on a simple idea that any STA can monitor the medium shared in a given BSS. Each station

calculates the time between two idle times on the channel to estimate the amount of congestion seen by the AP. Based on this, the delay in the system is estimated and is used to admit a new flow.

There are very few works using hybrid measurement-assisted and model-based admission control schemes [KTB04,BKT08,PA07,SV07]. In [KTB04,BKT08] Bensaou proposed a measurement-assisted model-based admission control in a centralized scenario. They approximate each flow as a new terminal and estimate the throughput possible for each terminal. The problem with this method is that each terminal uses throughput as a threshold, which will not help delay based traffic flows. They also suggested a slightly modified scheme that works in the distributed scenario.

In [PA07], Patil proposed a hybrid measurement and model based distributed admission control called BUFFET for the DCF of 802.11. It uses measurements as an input to the model and predicts the WLAN saturation thereby maintaining average delay within acceptable limits. However, the criterion used for saturation is still throughput based.

In [SV07], Smith proposed a hybrid measurement and model based scheme for EDCA of 802.11e in a centralized scenario. In this scheme, the queue size is used to calculate the number of virtual terminals and the unsaturated model is used to calculate the throughput threshold for the admission. We notice that most of the models reviewed in this section were developed with throughput as an admission criterion.

6.3 Load Estimation

The admission of a new flow into the system depends on the current load. It becomes quite difficult to know the current system load accurately in a distributed network. In this chapter, we present two load estimation methods.

6.3.1 Heuristic based Load Estimation

In Chapter 5, we observed that the system throughput linearly increases with the system load and reaches saturation. Therefore, the ratio of the unsaturated normalized throughput to that of the normalized saturated throughput can be approximated with linear equations. Similarly, we also observed that for a given packet, the system throughput reaches saturation approximately at the same system load irrespective of the network size N when an optimum p for the P-Persistent MAC protocol is used for a given network size N and packet size P . Figs. 6.1 and 6.2 show our observations for a packet size of 300 and 600 bytes respectively. The system load at which the initial saturation occurs is defined as $Sys. Load_{sat}$. Therefore, we use the above two observations to estimate the system load as follows:

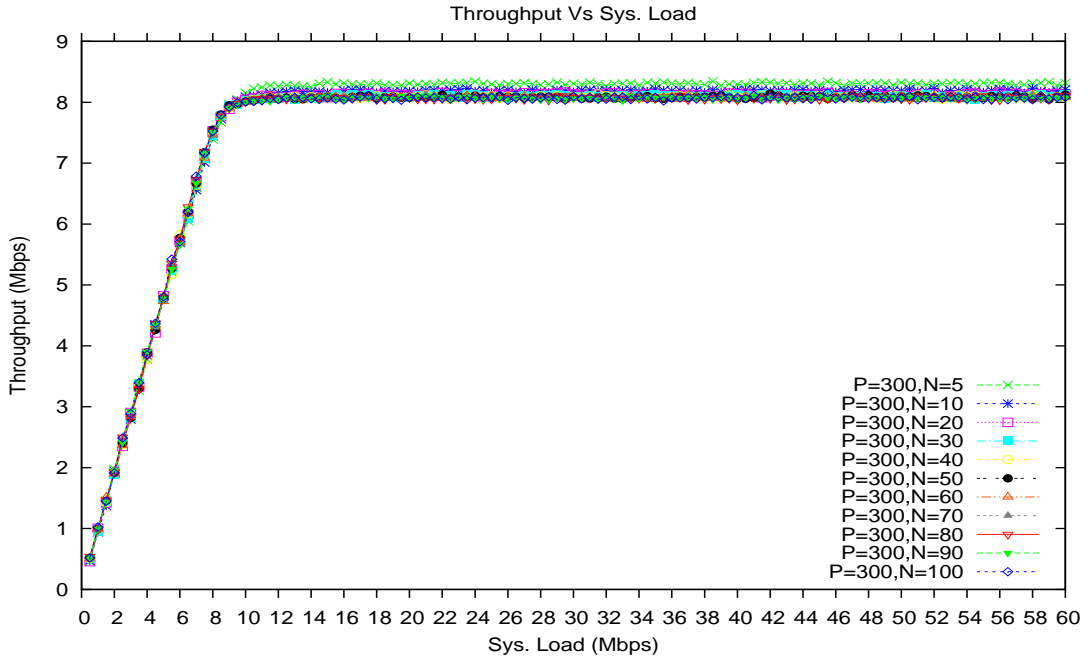


Figure 6.1: System throughput for different network sizes for P = 300 bytes.

$$Sys.Load_{est} = \frac{Normalized.Throughput_{unsat}}{Normalized.Throughput_{sat}} \times Sys.Load_{sat} \quad (6.1)$$

All the nodes will maintain a table of packet sizes and their corresponding saturated system load [AB08a]. The normalized saturated throughput can be calculated from Bianchi's work [Bia00]. Each node can measure the normalized throughput, successful transmission length, collision length, packet size and other parameters from the channel. The nodes maintain the averages of the measured values using the following exponential weighted moving average equation . Eq. 6.2

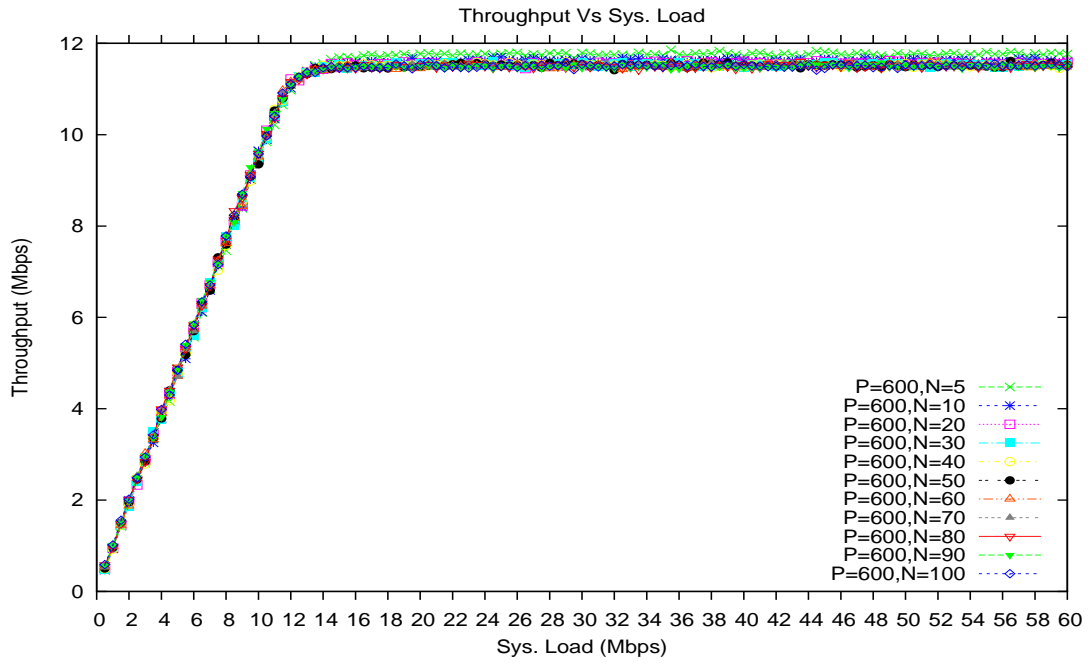


Figure 6.2: System throughput for different network sizes for P = 600 bytes.

gives an example with normalized throughput.

$$Norm.Sys.Thruput_{est} = 0.8 \times Norm.Sys.Thruput_{est} + 0.2 \times Mean.Norm.Sys.Thruput_{cur} \quad (6.2)$$

6.3.2 Analytical based Load Estimation

In this section, we use a closed form for the service time of a packet in estimating the system load.

In [?], Ziouva calculated the average service time of a packet as follows:

$$\bar{L} = \left(\frac{G}{S} - 1 \right) \cdot (T_F + \bar{Y} + \bar{R}) + (T_S + \bar{R}) \quad (6.3)$$

Where \bar{R} is the average back-off time, T_S is the length of the successful transmission, T_F is the length of the unsuccessful transmission, Y is a random variable that represents the time a station waits until a frame collision has been detected, G is the total load in the system, and S is the system throughput. We noticed that Eq. 6.3 is similar to our model for service time for no retry limit. Eq. 4.19 for the equal load case is given below to compare.

$$\bar{T} = S_C \cdot (T_B + T_C) + (T_B + T_S) \quad (6.4)$$

where S_C is the average number of transmission attempts ending with a collision before a packet is successfully transmitted. We observe from Eq. 6.3 and Eq. 6.4, that $\frac{G}{S} - 1 = S_C$. Thus, if we can measure the average packet size ($E[P]$), average transmission time of the packet ($TxTime(E[P])$),

S_C and the normalized throughput S , we can estimate the system load as follows

$$Sys.Load_{est} = \frac{(1 + S_C) \cdot S \cdot E[P]}{TxTime(E[P])} \quad (6.5)$$

Since, the normalized throughput S is expressed as a ratio of the $TxTime(E[P])$ and total time taken for the successful transmission of the packet, we eliminate the transmit time of the packet by dividing the S in Eq. 6.5 with $TxTime(E[P])$ and multiplying with $E[P]$ to get load in Bps or bps.

6.4 Distributed Admission Control

In this section, we present a measurement-assisted model-based distributed admission control scheme (MM-DAC) for the P-Persistent 802.11 MAC in the contention based channel access. In [KTB04], a measurement-assisted model-based call admission control scheme was proposed. It is a centralized scheme which can function in a distributed scenario with some modifications and is based on the bandwidth availability predictions.

In [BSE07], Bai et al made an important observation that an admission control algorithm that employs delay predictions as the threshold for call admission will in theory achieve better channel utilization than those that consider throughput as the threshold. Therefore, we develop our scheme based on delay predictions. The scheme uses our unsaturated delay model with the equal load case from Chapter 4 to estimate the end-to-end delay when a new flow is admitted into the system and

use this estimate end-to-end delay in the admission process. The main goal of an admission control is to make sure that the QoS requirements of the previously admitted flows are not affected while satisfying the QoS requirement of the new flow.

Our unsaturated delay model with equal load case is based on the assumption that all the nodes have same input load rate and thereby the same probability of having an empty queue. We use this model to overcome the difficulty of knowing the load of each node in the distributed wireless network. So, our admission control is based on approximating the system with unequal loads to a system with equal loads. We present a brief summary of our delay model from Chapter 4 below:

In Chapter 4, We defined the service time of a packet as the time interval between the time the packet is at the head of its queue ready for transmission and the time when an acknowledgement for this packet is received or the packet is discarded after the retry limit for collisions is exhausted. The average service time for a node is given by Eq. 4.25. Since we are using an equal load case, the service time for each node is equal.

$$\bar{T} = \bar{T}_j \quad 1 \leq j \leq N \quad (6.6)$$

In unsaturated load conditions, the average service time \bar{T} in Eq. 4.25 depends on the probability for the queue to become empty (q_0). The terms V_C and T_B have the $(1-q_0)$ term in their expressions. In saturated load conditions, the value of q_0 is 0. The service rate μ is the inverse of the average service time \bar{T} . In chapter 4, an end-to-end delay in a single hop WLANS is defined as the

time interval between the time at which a packet is placed in the queue and the time at which a corresponding ACK is received. The end-to-end delay of a packet is given by Eq. 4.29. As in the case of service time, the end-to-end delay is the same for each node.

$$\bar{D} = \bar{D}_j \quad 1 \leq j \leq N \quad (6.7)$$

So, for a given input load λ , the end-to-end delay can be found from Eqs. 4.27, 4.28 and 4.29. In unsaturated load conditions, for a given λ , the end-to-end delay estimate is calculated by using the iterative method described in Algorithm 4.1.

When admitting a new call, the admission control should make sure that the new call will not affect the delay requirements of the existing flows. The remaining nodes might also admit new calls at the same time as the current node in discussion even though it is a rare case, but a possibility. These new calls will not only affect the requirements of the existing flows but also the new call under consideration. Therefore, in a distributed scenario, each node needs to estimate the delay due to admission of calls from all the other nodes as well as the delay due to the addition of a new call within itself. This approach is conservative but will help in a distributed case and fits our model assumption.

To handle the lack of accurate global information in the distributed case, we introduce the concept of equivalent load λ_{equiv} and equivalent probability to find the queue to be empty at each node q_{0equiv} . We estimate λ_{equiv} by estimating the system load using Eq. 6.1 or Eq. 6.5 from the

load estimation methods described in section 6.3 and dividing it with the total number of nodes.

This gives an approximate load at each node.

$$\lambda_{equiv} = \frac{Sys.Load_{est}}{N} \quad (6.8)$$

When a new flow requests for admission at a node with load rate λ^1 and delay requirement D^1 , we estimate the delay in the system due to this flow and similar flows at all the nodes as follows

$$\bar{D}_{est} = I\left(\lambda_{equiv} + \frac{\lambda^1}{N}\right) \quad (6.9)$$

The function $I\left(\lambda_{equiv} + \frac{\lambda^1}{N}\right)$ is the iterative procedure given by Algorithm 4.1. Each node maintains minimum delay D_{min} which is the smallest delay requirement of all the flows it has accepted. The new call is admitted when the value \bar{D}_{est} is less than D^1 and D_{min} . We also add an additional admission criterion at each node. Each node keeps track of the total load rate λ_{total} which is the sum of the load rates of all its accepted flows. If $\frac{(\lambda_{total} + \lambda^1)}{\mu_{est}}$ is less than 1, then we accept the flow and update λ_{total} and D_{min} . Note μ_{est} is found during the estimation of \bar{D}_{est} .

6.5 Centralized Admission Control

In this section, we present a measurement-assisted model-based centralized admission control scheme (MM-CAC) for the P-Persistent 802.11 MAC in the contention based channel access. Compared to the distributed admission control, the centralized admission control has complete information about the total number of nodes (N), total load (λ_{totalj}), probability for the queue to become empty (q_{0j}), and service time (\bar{T}_j) of each node ($j = 1$ to N). Therefore, the end-to-end delay predictions are much accurate. Our model in Chapter 4 takes loads at each node into consideration and the results showed that the model is very accurate for different loads at each node.

As in the previous section, we take Bai's observation that an admission control algorithm that employs delay predictions as the threshold for call admission will in theory achieve better channel utilization than those that consider throughput as the threshold [BSE07]. Therefore, our centralized admission control is also based on end-to-end delay predictions. When a new flow is admitted into the system, it uses our unsaturated delay model for the unequal load case from Chapter 4 to estimate the end-to-end delay and use the estimated delay in the admission process.

Since, it is a centralized admission control, all the admission requests will be sent to one coordinator and we assume it would be the Access Point (AP) in a given Basic Service Set (BSS). When a new flow requests for admission at a node j with load rate λ^1 and delay requirement D^1 , node j sends its current state *viz.*, q_{0j} , and \bar{T}_j obtained by measurements and the new flow details

(λ^1 and D^1) to the AP. The AP keeps track of each flow with all its details at each node. The AP also keeps track of the total load λ_{totalj} , q_{0j} , service time \bar{T}_j and minimum delay D_{minj} for each node. Since a new flow affects the existing flows in the system, we estimate the end-to-end delay using Algorithm 4.1. The AP uses values λ_{totall} , q_{0l} , service time \bar{T}_l , for all the nodes except j ($l = 1$ to N and $l \neq j$) in Algorithm 4.1. However for node j , it uses $(\lambda_{totalj} + \lambda^1)$ as its total load, the $q_{0j(measured)}$ for q_{0j} and $\bar{T}_{j(measured)}$ for \bar{T}_j in Algorithm 4.1. Therefore the delay in the system due to this flow at each node i is given by

$$\bar{D}_{esti} = I\left(\lambda_{totalj} + \lambda^1, q_{0j(measured)}, \bar{T}_{j(measured)}, \lambda_{totall}, q_{0l}, \bar{T}_l\right) \quad 1 \leq l \leq N, i \neq j, 1 \leq i \leq N \quad (6.10)$$

The new call is admitted when the value \bar{D}_{esti} is less than D_{mini} (for $i = 1$ to N , $i \neq j$) and \bar{D}_{estj} is less than D^1 and D_{minj} . All the parameters for each node is updated with the estimated values obtained for q_{0l} and \bar{T}_l from Algorithm 4.1. The node parameters for node j are updated similarly. This way, each node has an estimate of its values when a flow is accepted. The value of D_{minj} for node j is also updated to the minimum value of D^1 and old D_{minj} . We can periodically update these values at each node with the measured values to keep the system information more accurate, but the periodic update adds more overload in the system. Therefore we use the estimates of all the parameters at each node.

6.6 Performance results

We present the performance evaluation of our admission control with the help of the simulation results generated using the NS2 simulator [Pro09]. The admission control component is added to our scheduler-based architecture that was added to the IEEE 802.11 DCF implementation of NS2. Also, we made sure that there is no overhead due to packets from ARP and routing protocols.

Our simulation setup consists of a single hop WLAN where each station can hear all the other stations. The WLAN operates in IEEE 802.11a mode with a data transmission rate of 54Mbps and the MAC layer system parameters are set as shown in Table 6.1. We ran our experiments for two sets of nodes *viz.*, 10 and 25 with data packet sizes of 300 and 600 bytes. The simulation is run for 300 seconds and an exponential traffic with a system load of 5 Mbps is divided into five individual loads as follows: $L = 5 \text{ Mbps}/N$, where N is 10 or 25. Now we generate five heterogeneous loads as follows: $0.5*L$, $0.75*L$, L , $1.25*L$, $1.5*L$. For example, in the case of 10 nodes and 5Mbps system load, we have 0.25 Mbps, 0.375 Mbps, 0.5 Mbps, 0.625 Mbps and 0.75 Mbps. One of these loads is generated at a node at every 5 seconds and in the case of 10 nodes, a system load of 5 Mbps is generated at the end of 50 seconds. The loads are added from the beginning to the end of the simulation. Earlier, we presented two load estimation methods for the distributed admission control and we present results for these two methods and results for the centralized admission control.

In our simulation tests, we observed a large set of data generated by all the nodes in one second and we therefore reduced the data set by averaging the data collected in every 0.25 second interval. However, in the case of interface queue drops, we add the drops collected in every 0.25 second interval. In the discussion below, we use the term delay to denote the end-to-end delay.

We compare the performance of our admission control for a minimum delay requirement of 2 milliseconds and 4 milliseconds in the system with that of no admission control. In this paper, we compare our admission control with no admission control to give a better understanding of the system behavior in the absence of admission control so that we can appreciate the value added by the use of an admission control. As the system load increases, the contention for the channel

Table 6.1: 802.11 MAC Parameters

tslot	9 μ s
SIFS	16 μ s
DIFS	34 μ s
Data Rate	54 Mbps
PLCP Data Rate	6 Mbps
Preamble Length Bits	144 bits
PLCP Header Length	48 bits
ACK Length Bytes	14 bytes
MAC Header Length Bytes	28 bytes

increases and so the end-to-end delay. Figs. 6.3 and 6.4 show the delay results for a packet size of 300 bytes for the 10 nodes using the two load estimation methods for the distributed admission control. Similarly, Fig. 6.5 shows the delay results for the centralized admission control. In the case of no admission control, there is a stepped increase in the delay with time as all the traffic is

allowed into the system. On the other hand, our distributed control stops admitting new flows to

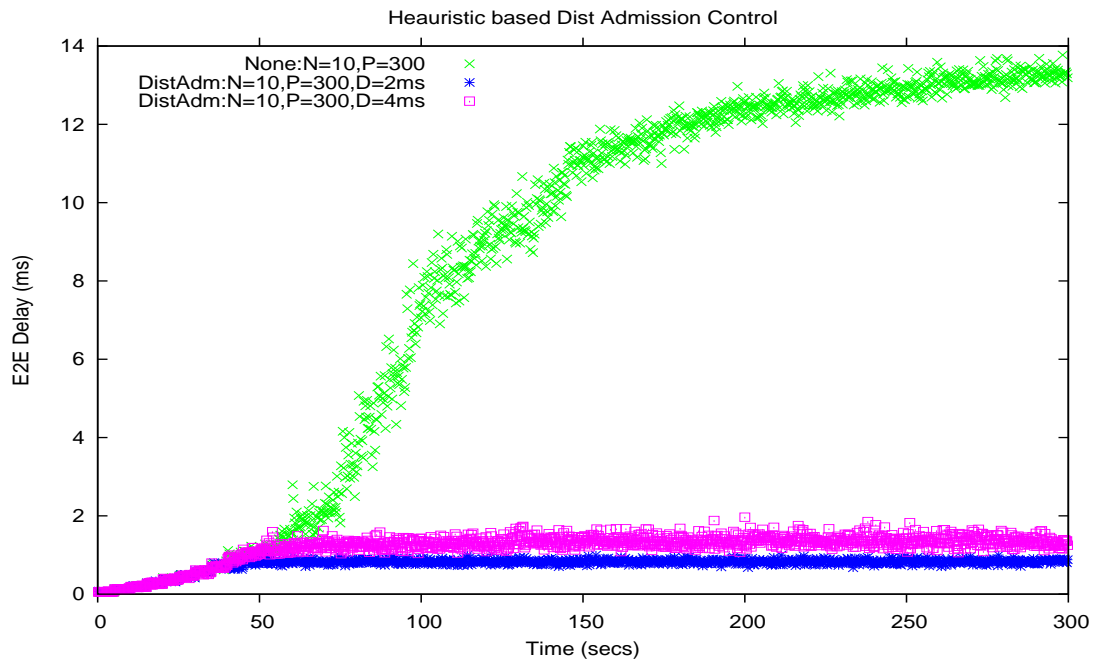


Figure 6.3: End-to-End Delay (N=10, P=300 bytes) Heuristic based Dist. Adm. Control.

keep the delay below the requirement of 2 ms and 4 ms, respectively. Since its difficult to know the accurate load information at each node, we approximated the system load estimation and therefore, the scheme admitted the flows very conservatively. On the other hand, the centralized admission control has complete information about all the nodes and thereby, it did better delay estimates and admitted more flows so that the delay is closer to 2 ms and 4 ms, respectively. We also noticed that the analytical based load estimation gave better performance than the heuristic based load estimation. A similar behavior is observed for a packet size of 600 bytes in Figs. 6.6, 6.7 and 6.8.

In the distributed case, the delay seems to be much lower than 2 ms and 4 ms respectively. This

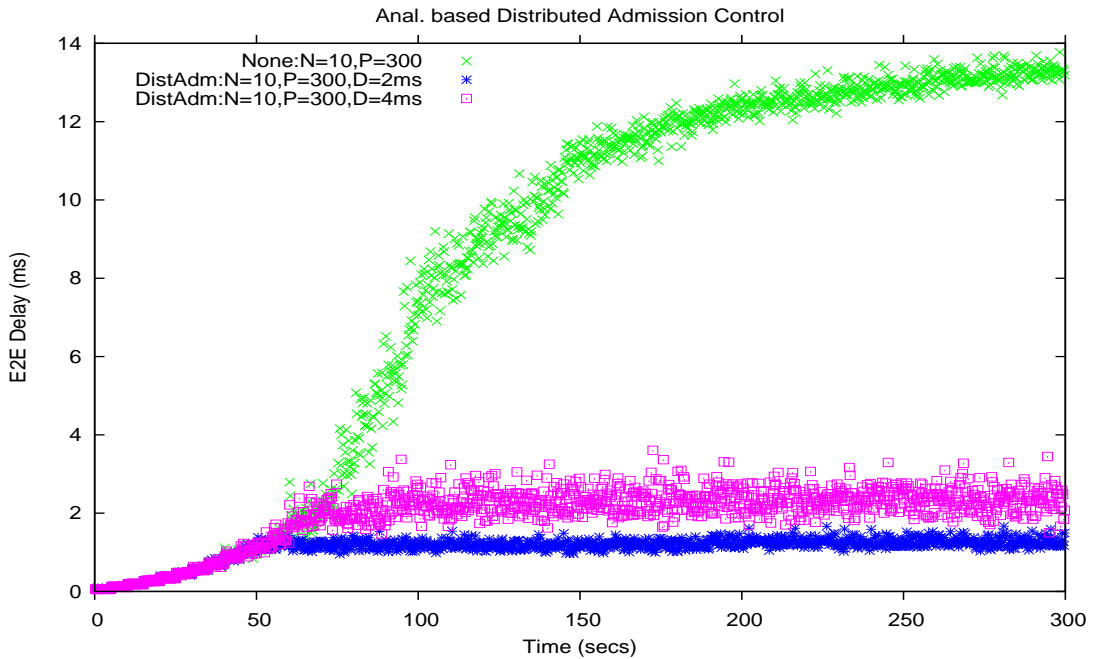


Figure 6.4: End-to-End Delay (N=10, P=300 bytes) Analytical based Dist. Adm. Control.

phenomenon is attributed to the addition of 5.0 Mbps system load in our tests. Also, our model estimates delay assuming there were similar load rates at all the other nodes which amounts to a total system load of 5.0 Mbps. This is a large amount of load and there might be a spike in the delay with this amount of load. Next, we look into the system throughput. As the system load increases, the system throughput also increases until it reaches its saturation limit and thereafter maintains a constant throughput. Figs. 6.9- 6.11 show exactly the same behavior when it came to the no admission control case. The figures also show lower throughput for the admission control compared to the no admission control scheme. This is because the admission control does not admit new flows when it reaches the delay requirements. So the smaller the delay requirement, the

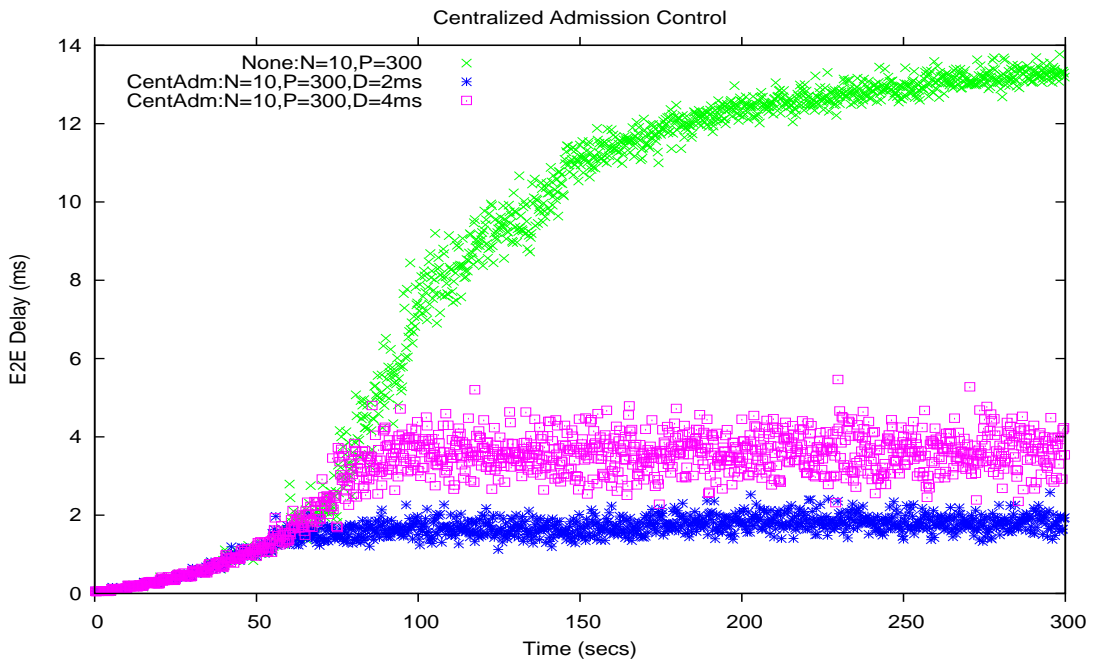


Figure 6.5: End-to-End Delay (N=10, P=300 bytes) Centralized Admission Control.

lower is the throughput. This is verified for the cases of 2 ms and 4 ms delay requirements. Also, we could see that the throughput for the centralized scheme is much better than the distributed scheme because it predicts the delay estimates much more accurately and so admits more traffic flows whereas distributed scheme uses a conservative approach based on the estimated system state. Similarly, Figs. 6.12- 6.14 show the number of packets dropped at the interface queue. As the load increases, the queue gets full and more and more packets gets dropped with increased load. The admission control prevents this by not allowing new flows based on the delay requirements. The smaller the delay, the lesser the flows which results in less contention and more packets gets

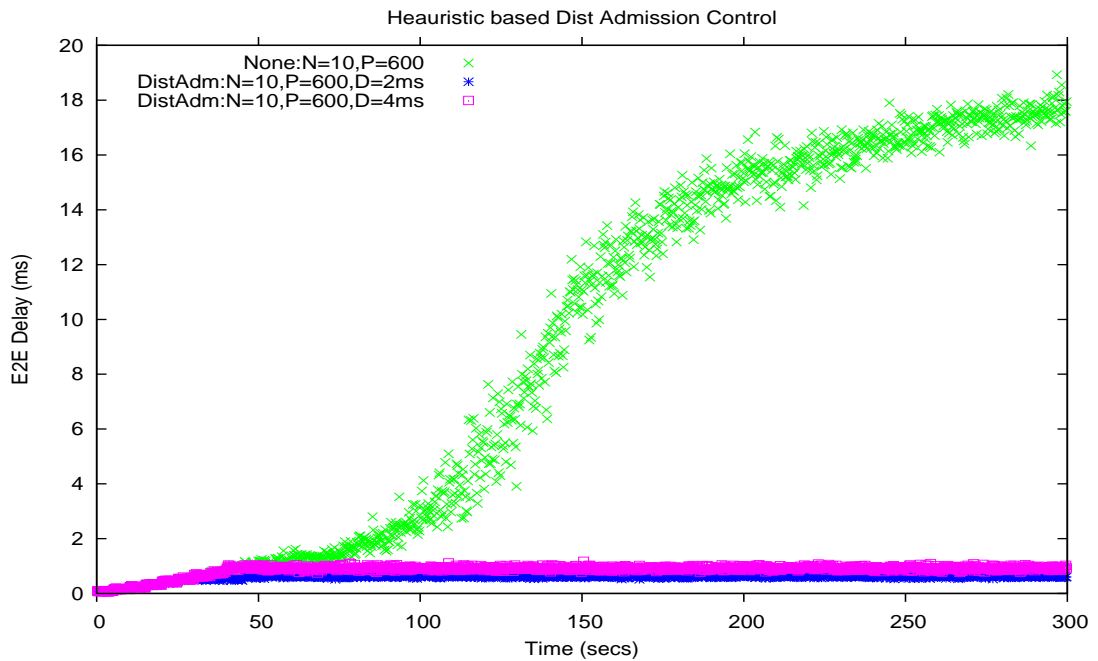


Figure 6.6: End-to-End Delay (N=10, P=600 bytes) Heuristic based Dist. Adm. Control.

transmitted with few or negligible drops. From the results, we can see that the admission control kept the number of packets dropped very low or almost zero.

We notice that there is a considerable IFQ (Interface Queue) drops in the centralized admission control as it admits more flows compared to the conservative distributed admission scheme. But, still the IFQ drops for the centralized scheme is small compared to the no admission scheme. We can clearly see the benefit of an admission control just by looking at the IFQ drops for the no admission case.

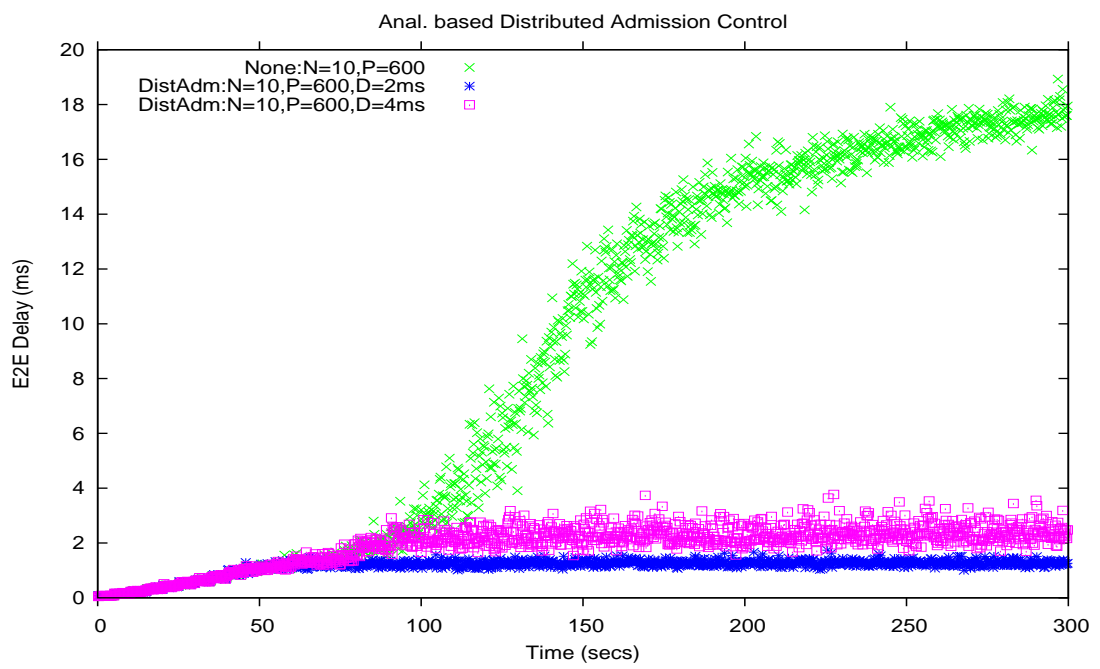


Figure 6.7: End-to-End Delay (N=10, P=600 bytes) Analytical based Dist. Adm. Control.

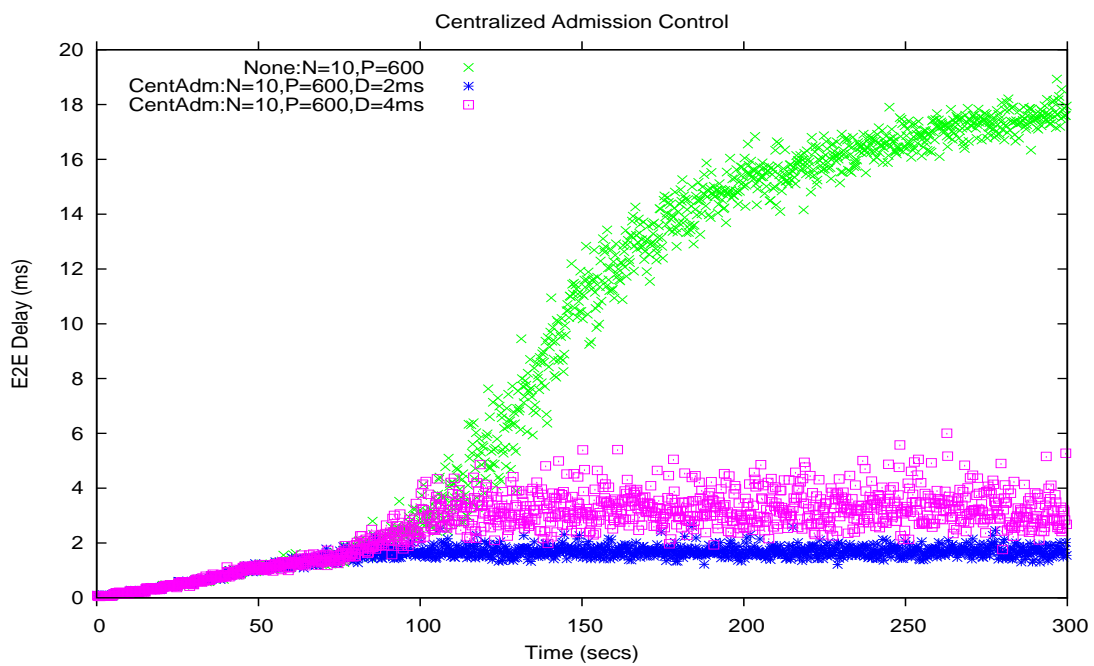


Figure 6.8: End-to-End Delay (N=10, P=600 bytes) Centralized Admission Control.

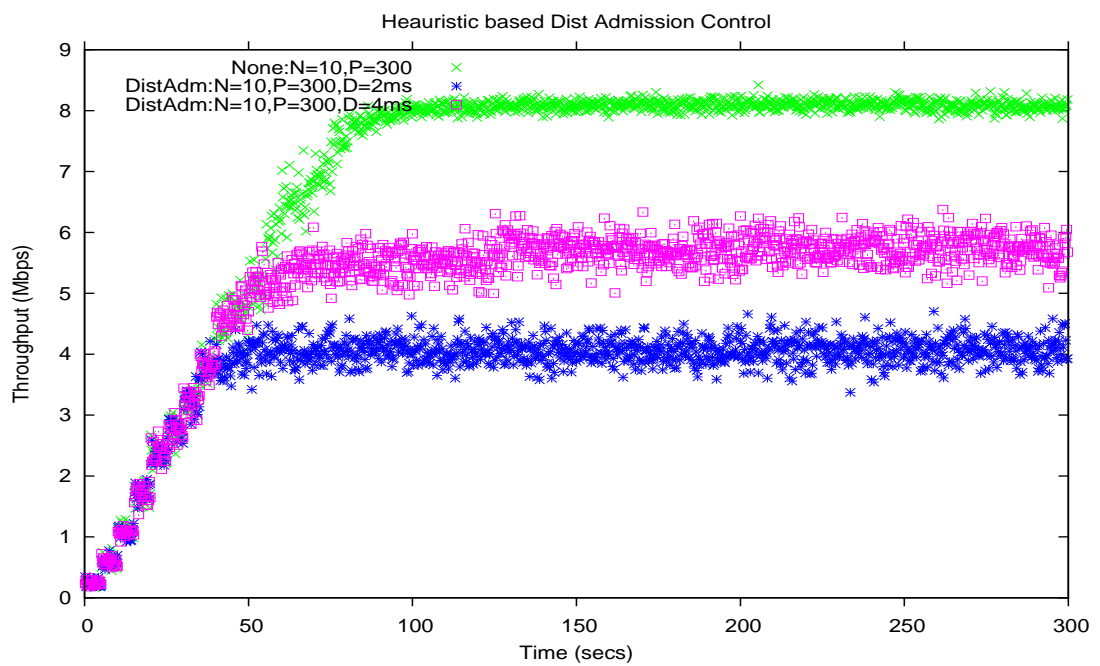


Figure 6.9: End-to-End Delay (N=10, P=300 bytes) Heuristic based Dist. Adm. Control.

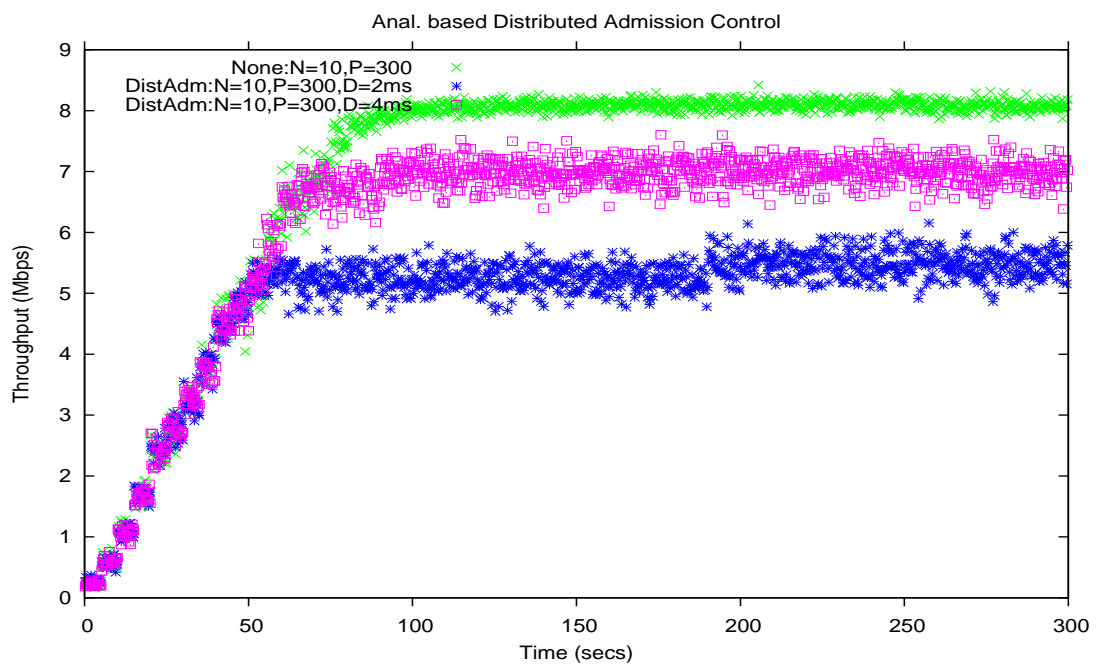


Figure 6.10: End-to-End Delay (N=10, P=300 bytes) Analytical based Dist. Adm. Control.

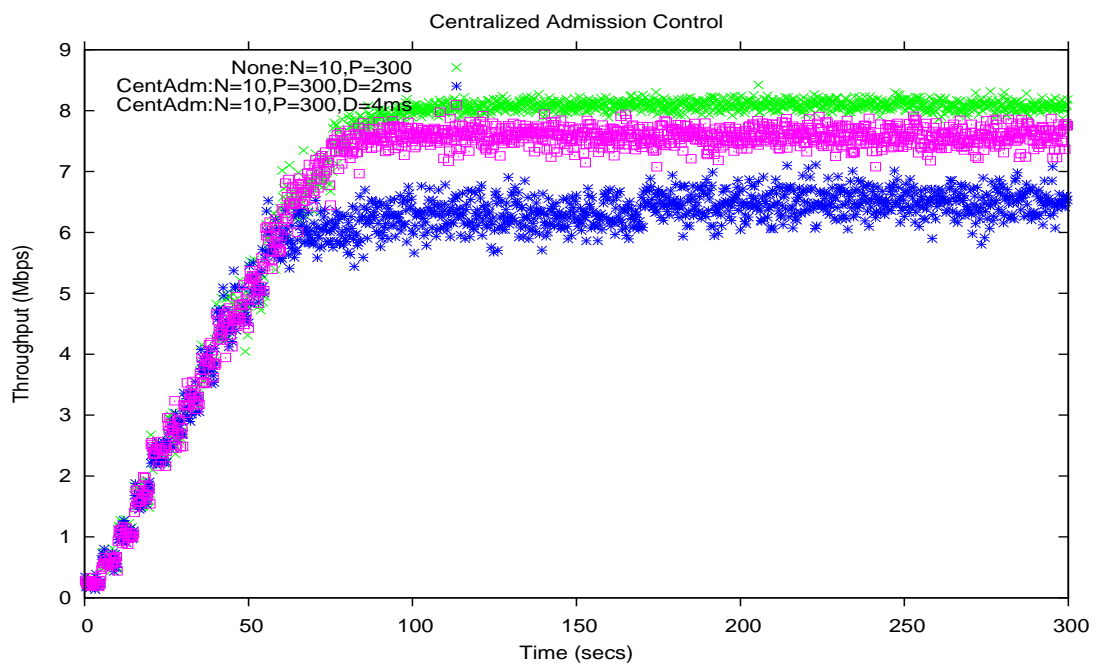


Figure 6.11: End-to-End Delay (N=10, P=300 bytes) Centralized Admission Control.

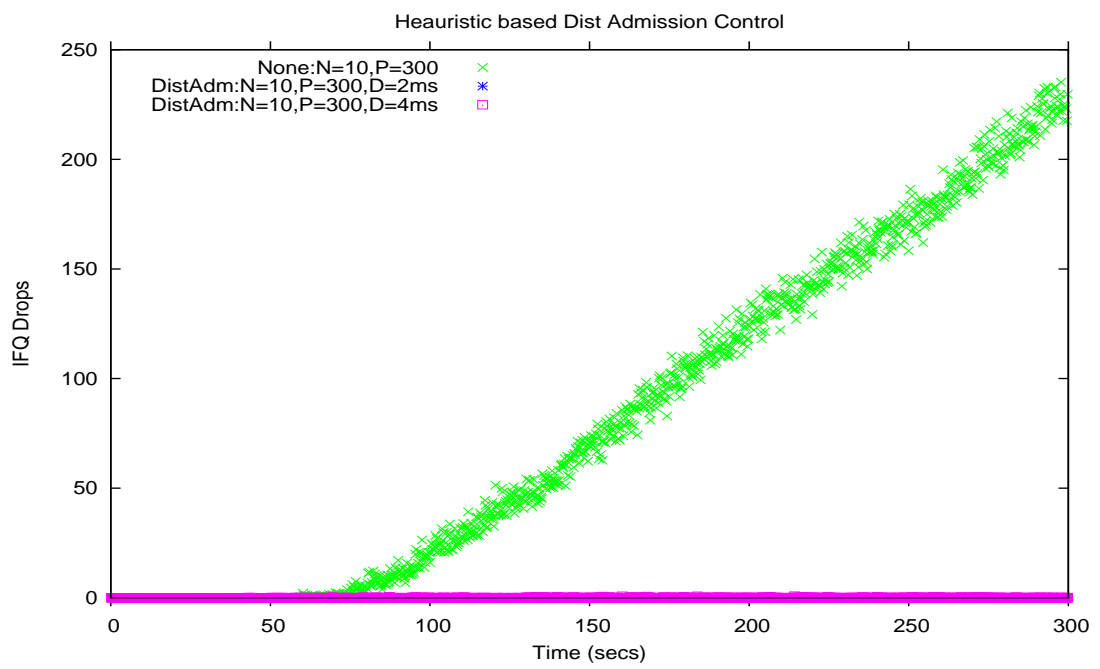


Figure 6.12: End-to-End Delay (N=10, P=300 bytes) Heuristic based Dist. Adm. Control.

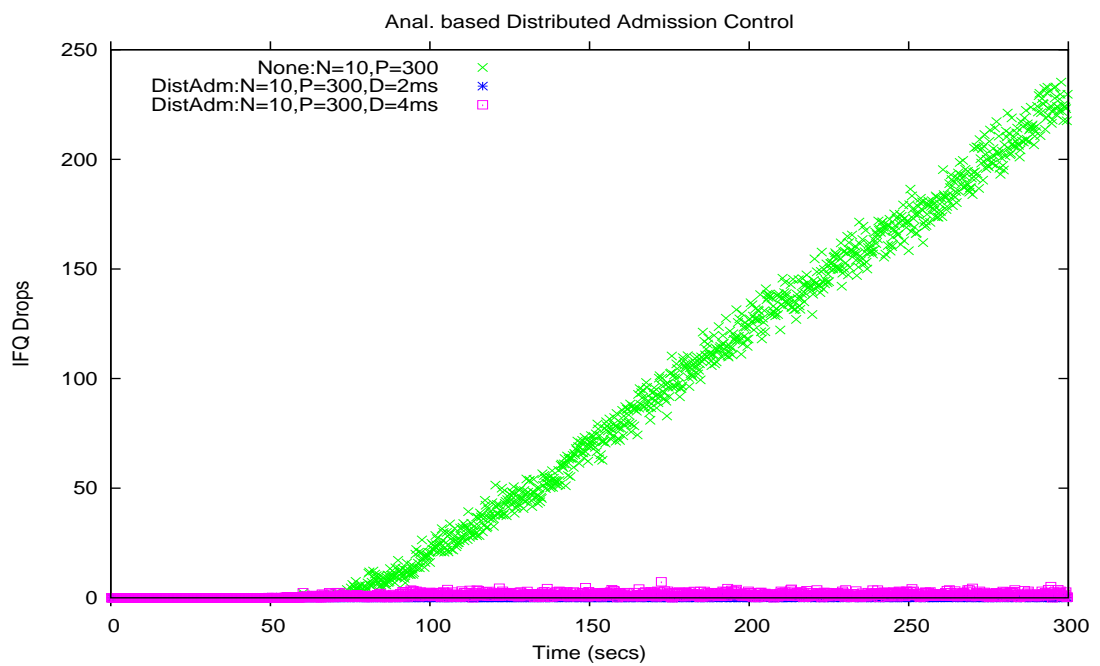


Figure 6.13: End-to-End Delay (N=10, P=300 bytes) Analytical based Dist. Adm. Control.

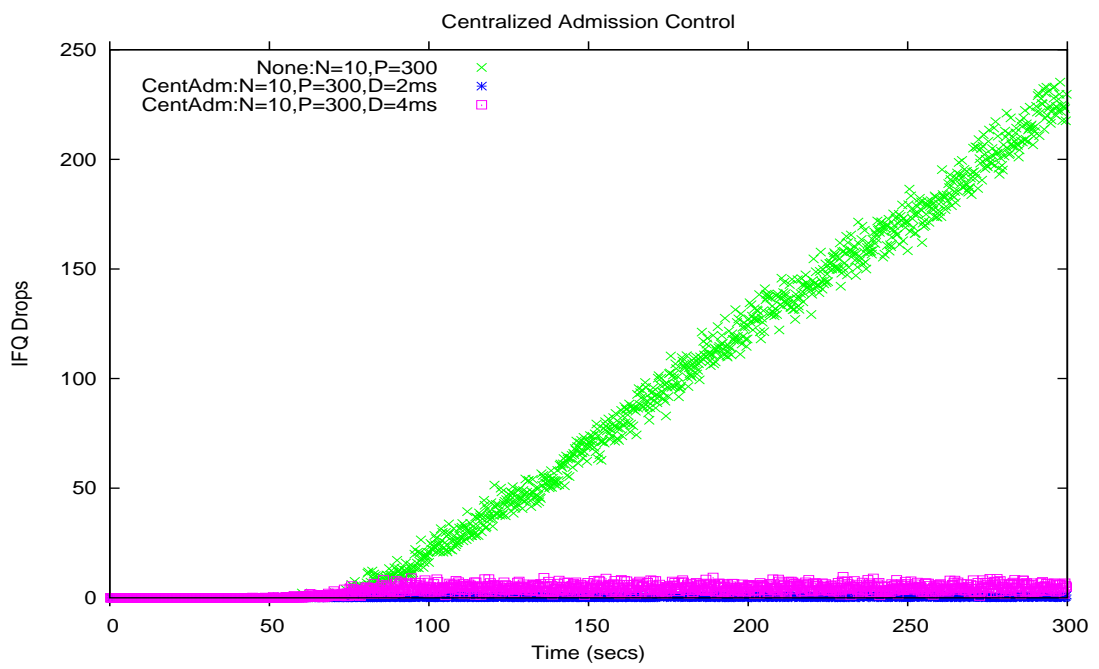


Figure 6.14: End-to-End Delay (N=10, P=300 bytes) Centralized Admission Control.

CHAPTER 7 : CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In this dissertation, we have discussed the inadequacies and problems associated with providing QoS support in contention based channel access of the IEEE 802.11e MAC protocol. The 802.11 MAC protocol was designed for best effort traffic and is extended to support multiple classes of traffic in 802.11e. However, the current design supports only service differentiation but not delay and/or throughput guarantees. We also observed that the same channel access parameters in 802.11e are used for two different components *viz.*, service differentiation and channel access. So, an attempt at realizing certain goals in one component will affect the other component detrimentally. Also, in order to provide a delay and/or throughput guarantees, there needs to be some kind of monitoring component which will make sure that the QoS guarantees are provided to all the existing flows in the system as well as the new flow which is about to enter into the system. Therefore, an admission control is an essential component to provide QoS.

We proposed a new framework that consists of three separate components *viz.*, a scheduler to provide service differentiation, a core MAC layer that provides efficient channel access and an admission control that provides QoS guarantees. The scheduler uses the weights assigned to the

ACs to provide the service differentiation. We proposed using P-Persistent based channel access at the core MAC layer. The P-Persistent access scheme uses only one channel access parameter and is therefore very easy to tune to achieve efficient channel utilization. We evaluated the separation of these two components and observed that it is easy to provide service differentiation while providing optimum channel utilization with little to no overhead. The same thing is a cumbersome process to achieve in the standard IEEE 802.11e protocol due to the need to tune multiple channel access parameters.

In order to develop an admission control, we need a detailed analysis of the P-Persistent 802.11 MAC protocol. In Chapter 4, we developed an accurate analytical model for the P-Persistent 802.11 in terms of three important metrics viz., normalized throughput, service time delay and end-to-end delay. We validated the analytical model with simulation results using NS2. The simulation results are very close to analytical results and the error is within 2%.

During our work on admission control, we came across an interesting observation. When the system load was increased, the throughput increased linearly and became constant at certain system load and remained the same for any higher load thereafter. However, most of the analytical models developed for saturated throughput assumed overloaded conditions, i.e., each node in the network is saturated and always has a packet to send. We performed experiments to prove that the saturation throughput does not require overloaded conditions. When there is sufficient load in the system to provide enough collisions, the system will reach its saturation throughput.

We developed a delay based measurement-assisted model-based admission control for our framework using the analytical model we developed in Chapter 4. It works in both the distributed and centralized scenarios. Our results showed that our admission control schemes provided the required QoS guarantees. However, the performance of the distributed admission control is inferior to the centralized admission control. This is due to the lack of complete state information of all the nodes in the distributed model.

Finally, we have concluded that our framework is very efficient and provides a simple and elegant solution that achieves all the three goals of the 802.11e MAC protocol.

7.2 Future Work

There are many ways to extend the research presented in this dissertation. The proposed framework to provide QoS in 802.11e MAC Protocol can be improved by adding additional features to make it as robust and complete as possible. Below are some possible areas worthy for future investigation.

- In this dissertation, we developed our model under perfect channel conditions and no hidden node problems. But, that doesn't accurately describe the real world scenario. The wireless channels are prone to bit errors due to interference and other factors. Therefore, one possible area for future work is to extend our model to include methods to deal with the hidden node problem and channel errors.

- Our analytical model describes only a single class of traffic. But, our framework supports multiple classes and service differentiation among them. The service time and end-to-end delays for each class depend on the weights assigned to the ACs that provide service differentiation. It will be interesting to extend our analytical model to support multiple classes taking the service differentiation parameters into consideration.
- Another possible area for future work is to extend our admission control schemes to support multiple classes of traffic taking the differentiation parameters into consideration.
- Since the 802.11 MAC protocol is a contention based access mechanism, it is prone to collisions among different classes of traffic. These collisions reduce the channel efficiency. It will be useful to extend our work to use TxOP and develop both the analytical model and admission control that support TxOP access.
- One of the problems with providing QoS for high priority traffic is starvation of low priority traffic such as best effort traffic that requires no QoS guarantees. Extending our proposed framework to include mechanisms to prevent starvation is an area worthy of further research.
- The proposed framework deals with one hop wireless LANS. One possible future task is to extend it to multi-hop wireless networks.

LIST OF REFERENCES

- [AB06a] K. Anna and M. Bassiouni. “A Scheduler Based Architecture for QoS Provisioning in IEEE 802.11 MAC Protocol.” In K Elleithy, T. Sobh, A. Mahmood, M. Iskander, and M. Karim, editors, *Advances in Computer, Information, and Systems Sciences, and Engineering*, pp. 308–315. Springer Netherlands, 2006.
- [AB06b] K. Anna and M. Bassiouni. “WSN07-4: Analytical Modeling and QoS Application of the P-Persistent 802.11 MAC Protocol.” In *IEEE Global Telecommunications Conference (GLOBECOM)*, pp. 1–6, San Francisco, CA, USA, November/December 2006.
- [AB08a] K. Anna and M. Bassiouni. “A Distributed Admission Control to Lower End-to-End Delay in P-Persistent 802.11 MAC protocol.” In *IEEE Wireless Telecommunications Symposium (WTS)*, pp. 316–323, Pomona, CA, USA, April 2008.
- [AB08b] K. Anna and M. Bassiouni. “A New Framework for QoS Provisioning in WLANs using P-Persistent 802.11 MAC.” *Computer Communications*, **31**(17):4035–4048, November 2008.
- [AB10] K. Anna and M. Bassiouni. “Analysis of the saturation throughput and node contention levels in the 802.11 MAC protocol.” In *IEEE Radio and Wireless Symposium (RWS)*, pp. 609–612, New Orleans, LA, USA, January 2010.
- [AS03] Farshid Alizadeh-Shabdiz and Suresh Subramaniam. “A Finite Load Analytical Model for the IEEE 802.11 Distributed Coordination Function MAC.” In *Workshop Modeling Optimization Mobile Ad Hoc Wireless Networks*, pp. 321–322, 2003.
- [BBA05] Y. Barowski, S. Biaz, and P. Agrawal. “Towards the Performance Analysis of IEEE 802.11 in Multi-hop Ad-hoc Networks.” In *IEEE Wireless Communications and Networking Conference (WCNC)*, volume 1, pp. 100–106, March 2005.
- [Bia98] G. Bianchi. “IEEE 802.11-Saturation Throughput Analysis.” *IEEE Communications Letters*, **2**(12):318–320, December 1998.
- [Bia00] G. Bianchi. “Performance Analysis of the IEEE 802.11 Distributed Coordination Function.” *IEEE Journal on Selected Areas in Communications*, **18**(3):535–547, March 2000.

- [BKT08] B. Bensaou, Zhen-Ning Kong, and D.H.K. Tsang. “A Measurement-Assisted, Model-Based Admission Control Algorithm for IEEE 802.11e.” In *International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN)*, pp. 260–265, Sydney, NSW, Australia, May 2008.
- [BP02] A. Banchs and X. Perez. “Distributed Weighted Fair Queuing in 802.11 Wireless LAN.” In *IEEE International Conference on Communications (ICC)*, volume 5, pp. 3121–3127, 2002.
- [BSE07] A. Bai, T. Skeie, and P.E. Engelstad. “A Model-Based Admission Control for 802.11e EDCA using Delay Predictions.” In *IEEE International Conference on Performance, Computing, and Communications Conference (IPCCC)*, pp. 226–235, New Orleans, LA, USA, April 2007.
- [BT03] G. Bianchi and I. Tinnirello. “Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 Network.” In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pp. 844 – 852, March 2003.
- [CBV03] P. Chatzimisios, A.C. Boucouvalas, and V. Vitsas. “IEEE 802.11 Packet Delay-A Finite Retry Limit Analysis.” In *IEEE Global Telecommunications Conference (GLOBECOM)*, volume 2, pp. 950–954, December 2003.
- [CBV05] P. Chatzimisios, A.C. Boucouvalas, and V. Vitsas. “IEEE 802.11 Wireless LANs: Performance Analysis and Protocol Refinement.” *EURASIP Wireless Communications and Networking*, **2005**(1):67–78, 2005.
- [CCG00a] F. Cali, M. Conti, and E. Gregori. “Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit.” *IEEE/ACM Transactions on Networking*, **8**(6):785–799, December 2000.
- [CCG00b] F. Cali, M. Conti, and E. Gregori. “IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism.” *IEEE Journal on Selected Areas in Communications*, **18**(9):1774–1786, September 2000.
- [CG03] M.M. Carvalho and J.J. Garcia-Luna-Aceves. “Delay Analysis of IEEE 802.11 in Single-hop Networks.” In *Proceedings of 11th IEEE International Conference on Network Protocols (ICNP)*, pp. 146–155, November 2003.
- [CSS05] Chun-Ting Chou, S.N. Shankar, and K.G. Shin. “Achieving per-stream QoS with Distributed Airtime Allocation and Admission Control in IEEE 802.11e Wireless LANs.” In *Proceedings of IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, volume 3, pp. 1584–1595, March 2005.

- [DML05] K. Duffy, D. Malone, and D.J. Leith. “Modeling the 802.11 Distributed Coordination Function in Non-saturated Conditions.” *IEEE Communications Letters*, **9**(8):715–717, August 2005.
- [EV05] M. Ergen and P. Varaiya. “Throughput Analysis and Admission Control for IEEE 802.11a.” *Mobile Network Applications*, **10**(5):705–716, October 2005.
- [FZ02] C.H. Foh and M. Zukerman. “Performance Analysis of the IEEE 802.11 MAC Protocol.” In *European Wireless Conference*, pp. 184–190, Florence, Italy, February 2002.
- [GC05] M Garetto and CF Chiasserini. “Performance Analysis of 802.11 WLANs under Sporadic Traffic.” In Boutaba, R and Almeroth, K and Puigjaner, R and Shen, S and Black, JP, editor, *Networking 2005: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication Systems*, volume 3462 of *Lecture Notes in Computer Science*, pp. 1343–1347. Springer-Verlag Berlin, 2005.
- [GCN05] D. Gao, J. Cai, and K.N. Ngan. “Admission Control in IEEE 802.11e Wireless LANs.” *IEEE Network*, **19**(4):6–13, July/August 2005.
- [Ge04] Y. Ge. *QoS provisioning for IEEE 802.11 MAC protocols*. PhD thesis, Ohio State University, September 2004.
- [GHC07] Y. Ge, J. C. Hou, and S. Choi. “An Analytic Study of Tuning Systems Parameters in IEEE 802.11e Enhanced Distributed Channel Access.” *Computer Networks*, **51**(8):1955–1980, June 2007.
- [GK04] N. Gupta and P.R. Kumar. “A Performance Analysis of the 802.11 Wireless LAN Medium Access Control.” *Communications in Information and Systems*, **3**(4):279–304, September 2004.
- [GZ03] D. Gu and J. Zhang. “A New Measurement-based Admission Control Method for IEEE 802.11 Wireless Local Area Networks.” In *14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications (PIMRC)*, volume 3, pp. 2009–2013, September 2003.
- [HER09] B. Hamdaoui, M. Elaoud, and P. Ramanathan. “A Delay-based Admission Control Mechanism for Multimedia Support in IEEE 802.11e Wireless LANs.” *Wireless Networks*, **15**(7):875–886, October 2009.
- [IEE99] IEEE. *IEEE Std. 802.11-1999, Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, 1999.

- [IEE04] IEEE. *IEEE 802.11e/D11.0, Draft Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, 2004.
- [KH04] H. Kim and J. C. Hou. “Improving Protocol Capacity for UDP/TCP Traffic with Model-based Frame Scheduling in IEEE 802.11-operated WLANs.” *IEEE Journal on Selected Areas in Communications*, **22**(10):1987–2003, December 2004.
- [Kle75] L. Kleinrock. *Queueing systems*, volume 1. John Wiley and Sons, Inc., NewYork, 1975.
- [KLW03] Y. L. Kuo, C.H. Lu, E.H.K. Wu, and G.H. Chen. “An Admission Control Strategy for Differentiated Services in IEEE 802.11.” In *IEEE Global Telecommunications Conference (GLOBECOM)*, volume 2, pp. 707–712, December 2003.
- [KTB04] Z. Kong, D.H.K. Tsang, and B. Bensaou. “Measurement-assisted Model-based Call Admission Control for IEEE 802.11e WLAN Contention-based Channel Access.” In *The 13th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN)*, pp. 55–60, April 2004.
- [LLC03] Wei Liu, Wenjing Lou, Xiang Chen, and Yuguang Fang. “A QoS-Enabled MAC Architecture for Prioritized Service in IEEE 802.11 WLANs.” In *IEEE Global Telecommunications Conference (GLOBECOM)*, volume 7, pp. 3802–3807, December 2003.
- [LM09] Y. Liu and M. Meng. “Survey of Admission Control Algorithms in IEEE 802.11e Wireless LANs.” In *International Conference on Future Computer and Communication (FCC)*, pp. 230–233, Chongqing, China, June 2009.
- [Ni02] Q. Ni. “Adaptive EDCAF and its implementation.” <http://www-sop.inria.fr/planete/qni/Research/AEDCAF/>, October 2002.
- [PA07] P. Patil and V. Apte. “An Autonomous Distributed Admission Control Scheme for IEEE 802.11 DCF.” In *the Fourth International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness and Workshops (QSHINE)*, pp. 1–7, Vancouver, Canada, August 2007.
- [PBK02] W. Pattara-Aukom, S. Banerjee, and P. Krishnamurthy. “Starvation Prevention and Quality of Service in Wireless LANs.” In *The 5th International Symposium on Wireless Personal Multimedia Communications*, volume 3, pp. 1078–1082, October 2002.
- [PM03] D. Pong and T. Moors. “Call Admission Control for IEEE 802.11 Contention Access Mechanism.” In *IEEE Global Telecommunications Conference (GLOBECOM)*, volume 1, pp. 174–178, December 2003.

- [Pro09] The VINT Project. “The Network Simulator - ns-2.” http://nslam.isi.edu/nslam/index.php/Main_Page, June 2009.
- [RNT03] L. Romdhani, Qiang Ni, and T. Turletti. “Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks.” In *IEEE Wireless Communications and Networking Conference (WCNC)*, volume 2, pp. 1373–1378, New Orleans, LA, USA, March 2003.
- [SS08] S. Shin and H. Schulzrinne. “Call Admission Control in IEEE 802.11 WLANs Using QP-CAT.” In *the 27th IEEE Conference on Computer Communications (INFOCOM)*, pp. 726–734, Phoenix, AZ, USA, April 2008.
- [SV07] C. Smith and N. Ventura. “A Measurement-Aided Model-Based Admission Control Scheme for IEEE 802.11e EDCA Wireless LANs.” In *Proceeding of SATNAC*, September 2007.
- [TDE09] N.C. Taher, Y.G. Doudane, and B. El Hassan. “A Complete and Accurate Analytical Model for 802.11e EDCA under Saturation Conditions.” In *IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, pp. 800–807, Rabat, Morocco, May 2009.
- [TS04] O. Tickoo and B. Sikdar. “Queueing Analysis and Delay Mitigation in IEEE 802.11 Random Access MAC based Wireless Networks.” In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, (INFOCOM)*, volume 2, pp. 1404–1413, March 2004.
- [VDG05] N. Vaidya, A. Dugar, S. Gupta, and P. Bahl. “Distributed Fair Scheduling in a Wireless LAN.” *IEEE Transactions on Mobile Computing*, **4**(6):616–629, November/December 2005.
- [VL02] S. Valaee and Baochun Li. “Distributed Call Admission Control for Ad Hoc Networks.” In *Proceedings of 56th IEEE Vehicular Technology Conference (VTC)*, volume 2, pp. 1244–1248, 2002.
- [WPK02] H. Wu, Y Peng, Long K., S. Cheng, and J. Ma. “Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LAN: Analysis and Enhancement.” In *Proceedings of IEEE Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, volume 2, pp. 599–607, 2002.
- [XL04a] Y. Xiao and H. Li. “Evaluation of Distributed Admission Control for the IEEE 802.11e EDCA.” *IEEE Communications Magazine*, **42**(9):S20–S24, September 2004.

- [XL04b] Y. Xiao and H. Li. “Voice and Video Transmissions with Global Data Parameter Control for the IEEE 802.11e Enhance Distributed Channel Access.” *IEEE Transactions on Parallel and Distributed Systems*, **15**(11):1041–1053, November 2004.
- [XLC04] Y. Xiao, H. Li, and S. Choi. “Protection and Guarantee for Voice and Video Traffic in IEEE 802.11e Wireless LANs.” In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, (INFOCOM)*, volume 3, pp. 2152–2162, Hong Kong, March 2004.
- [YFS07] K. Yasukawa, A.G. Forte, and H. Schulzrinne. “Distributed Delay Estimation and Call Admission Control in IEEE 802.11 WLANs.” In *IEEE International Conference on Network Protocols (ICNP)*, pp. 334–335, Dresden, Germany, June 2007.
- [YK04] Y. Yang and R. Kravets. “Distributed QoS Guarantees for Realtime Traffic in Ad Hoc Networks.” In *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pp. 118–127, October 2004.
- [YK05] Y. Yang and R. Kravets. “Contention-aware admission control for ad hoc networks.” *IEEE Transactions on Mobile Computing*, **4**(4):363–377, July/August 2005.
- [ZA02] E. Ziouva and T. Antonakopoulos. “CSMA/CA Performance under High Traffic Conditions: Throughput and Delay Analysis.” *Computer Communications*, **25**(3):313–321, February 2002.
- [ZCF06] H. Zhai, X. Chen, and Y. Fang. “A Call Admission and Rate Control Scheme for Multimedia Support over IEEE 802.11 Wireless LANs.” *Wireless Networks*, **12**(4):451–463, July 2006.
- [ZLC04] Hua. Zhu, M. Li, I. Chlamtac, and B. Prabhakaran. “A Survey of Quality of Service in IEEE 802.11 Networks.” *IEEE Wireless Communications*, **11**(4):6–14, August 2004.
- [ZZ04] L. Zhang and S. Zeadally. “HARMONICA: Enhanced QoS Support with Admission Control for IEEE 802.11 Contention-based Access.” In *Proceedings of 10th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pp. 64–71, May 2004.