QoS Guarantees for Real Time Applications in 802.11 WLANs

M.Tech Dissertation

Submitted in partial fulfillment of the requirements
for the degree of

Master of Technology

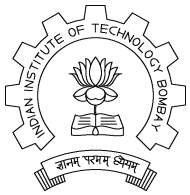
by

Kiran P Diwakar

Roll No: 02329026

under the guidance of

Dr. Sridhar Iyer



Kanwal Rekhi School of Information Technology
Indian Institute of Technology, Bombay
Mumbai

2004

To my **mother**

Mrs. Kshitija P Diwakar,

who has always been my source of love and inspiration.

Acknowledgements

I would like to thank my advisor, Prof. Sridhar Iyer for the consistent directions he has fed into my work. His constant assistance and overwhelming interest in the project has made it so special and successful. I express my deep sense of gratitude and appreciation towards him.

I would also like to thank Kalpesh Patel, Kalyan Rao D and Ranjeeth Kumar for all the support. I would also like to mention that the frequent discussions with the SIGNET group of KReSIT helped me to get valuable feedback for my work. I take this opportunity to thank my batchmates and labmates for being incredibly supportive.

Finally I would like to thank my family and all my friends for helping and believing in me.

Contents

List of Figures

Abstract

IEEE 802.11 standard specifies the MAC and PHY layers of the OSI Seven Layer Model for Wireless LANs. 802.11 is by far the most widely used and popular of the suite of WLANs. Channel access is both distributed as well as centralized called DCF and PCF respectively.

The expansion of IEEE 802.11 based WLANs has created interest in providing Quality of Service (QoS) guarantees in such networks. 802.11e suggests a flow-based solution based on priority queues at the node, for providing QoS guarantees. We propose a different approach, using a variant of TDMA, called Dynamic Time-division Multiple Access (DTMA). It is based on the observation that the ratio of data transmission time to control packet (poll or acknowledgement) transmission time is typically 6:1, and drops down almost to 2:1 when the data packet size becomes smaller than 500 bytes. DTMA focusses on reducing control information and thereby increasing time available for data transmission. More time available for data transmission implies increased throughput.

We expunge the overhead of control packets in the HCF by using cummulative acknowledgements and piggybacking.Taking advantage of 802.11's inherent limited range, DTMA, the modified TDMA is used for data transfer in HCF, without encountering the problem of distributed time synchronization. In this dissertation, we do the analysis of 802.11e and DTMA using probability models and also simulations to support these models. Thus, aided by simulations and analytical methods, we prove that DTMA has stricter and lesser delay bounds than 802.11e, for real-time and QoS sensitive applications. We also show that DTMA enhances the overall throughput by almost 20%, thereby implying better QoS guarantees in the 802.11 domain.

Chapter 1

Introduction

1.1 The Wireless World

Wireless computing is a rapidly emerging technology providing users with network connectivity without being tethered off of a wired network. Wireless local area networks (WLANs), like their wired counterparts, are being developed to provide high bandwidth to users in a limited geographical area. WLANs are being studied as an alternative to the high installation and maintenance costs incurred by traditional additions, deletions, and changes experienced in wired LAN infrastructures. Physical and environmental necessity is another driving factor in favor of WLANs.

The operational environment may not accommodate a wired network, or the network may be temporary and operational for a very short time, making the installation of a wired network impractical. Examples where this is true include ad hoc networking needs such as conference registration centers, campus classrooms, emergency relief centers and tactical military environments. However, to meet these objectives, the wireless community faces certain challenges and constraints that are not imposed on their wired counterparts.

1.2 Need for Separate MAC for Wireless

Existing MAC schemes from wired networks like, CSMA/CD are not directly applicable to wireless medium. In CSMA/CD sender senses the medium to see if it is free. If medium

is busy, the sender waits until it is free. If the medium is free, sender starts transmitting data and also continues to listen into the medium. It stops transmission as soon as it detects collision and sends a jam signal. In wired medium, this works because more or less the same signal strength can be assumed all over the wire. If collision occurs somewhere in the wire, everybody will notice it. This assumption gets invalidated in wireless medium, as the signal strength decreases proportionally to the square of distance to the sender.

In wireless medium, sender may apply carrier sense and detect an idle medium. Thus, the sender starts sending, but a collision happens at the receiver due to a second sender. Second sender may or may not be audible to first sender. Hence the sender detects no collision, assumes that data has been transmitted without errors, but actually a collision might have destroyed the data at the receiver. Besides that, wireless devices are half duplex and battery operated. They are unable to listen to the channel for collision while transmitting data.

1.3 Challenges

1. Global operation: WLAN products should sell in all countries, therefore, many national and international frequency regulations have to be considered.

2. Low Power: Devices communicating via a WLAN are typically also wireless devices running on battery power. Hence, WLAN must implement special power saving modes and power management functions.

3. License-free operation: LAN operators do not want to apply for a special license in order to be able to use the product. Thus, the equipment must operate in a license-free band, such as the 2.4 GHz ISM band.

4. Bandwidth: Bandwidth is the one of the most scarce resource in wireless networks. The available bandwidth in wireless networks is far less than the wired links.

5. Link Errors: Channel fading and interference cause link errors and these errors may sometimes be very severe.

6. Robust transmission technology: Compared to their wired counterparts, WLANs operate under difficult conditions. If they use radio transmission, many other electrical

devices may interfere.

7. Simplified spontaneous co-operation: To be useful in practice, WLANs should not require complicated setup routines but should operate spontaneously after power up. Otherwise these LANs would not be useful for supporting e.g., ad hoc meetings, etc.

8. Easy to use: LANs should not require complex management but rather work on a plug-and-play basis.

9. Protection of investment: A lot of money has already been invested into wired LANs. Hence new WLANs must protect this investment by being inter operable with the existing networks.

10. Safety and security: Most important concern is of safety and security. WLANs should be safe to operate, especially regarding low radiation. Furthermore, no users should be able to read personal data during transmission i.e., encryption mechanism should be integrated. The network should also take into account user privacy.

11. Transparency for application: Existing applications should continue to run over WLANs. The fact that wireless access and mobility should be hidden if not relevant.

1.4 QoS in Networks

Quality of Service (QoS) is a broad term used to describe the overall experience the user or application will receive over a network [26]. With the increase in real-time applications like streaming, video conferencing etc over the networks, there arises the need to give assured service quality to these applications. For example, if QoS is not given to say streaming video application traffic, there could be distortion of the images and the user might not be able to perceive the video properly.

Network architects now have to design QoS aware networks that can fulfill the service requirements of both, the newer time sensitive as well as the conventional applications. Channel utilization also has to be almost perfect leaving very little margin for wastage of the channel time. Many architectures and schemes are being used in the wired as well as the wireless networks to provide the required quality of service to the end user. Architectures like Integrated Services (IntServ), Differentiated Services (DiffServ), Multi Protocol Label Switching

(MPLS) will be covered briefly in the next chapter. Also we will give an overview of the various real time protocols like Session Initiation Protocol (SIP), Realtime Transmission Protocol (RTP/RTCP) and have a look at the extra features in them to handle realtime traffic. Before going to our proposed scheme, we will have a detailed analysis of the working of 802.11b and 802.11e.

1.5 Problem Statement

There is a tremendous increase in the real time applications on the wireless networks. These applications require some bounded delay guarantees and fixed minimum transmission opportunities. So, providing Quality of Service has become an important issue.

The 802.11 Working Group introduced a variant 802.11e that suggests a flow based solution using priority queues at the node, for providing QoS guarantees. It provides delay guarantees for the priority nodes that run real time applications. But with the increase in the number of nodes associated with a particular Access Point, the delay increases. So, there needs to be some architecture or scheme that will provide stricter and lesser delay guarantees and give a steady output performace even with more number of nodes.

The time the channel spends in control information transmission is a large portion of the total time. For small packet sizes, this time is almost half of the actual data tranmission time. It is clear that any non-data transmission is a wastage of the channel time as this does not directly increase the throughput. So, if the control information is reduced, it would directly increase the time available for actual data transmission in a given time frame. This would help in increasing the throughput of the system and also give more transmission opportunities for some privileged nodes, thereby reducing the delays. So, we propose a protocol, DTMA, that makes use of implicit control messages to improve the performance.

The significant contributions of this thesis are :

1. An insight into the current works in the field of WLANs.

2. A protocol granting stricter and lesser QoS guarantees.

3. Analysis of 802.11e and DTMA using theoretical models.

4. Simulation results supporting the theoretical models.

1.6 Thesis Outline

In this thesis, the next chapter summarizes the various QoS architectures and also briefly looks at the various real time protocols that are currently prevalent. Chapter 3 delves into the details of 802.11b WLAN standard, followed by an insight into the 802.11e MAC. Chapter 4 deals with the related works in the field of the providing QoS guarantees in WLANs.
Chapter 5 portrays our proposed scheme DTMA, with a detailed theoretical analysis of both 802.11e as well as DTMA in the following chapter. The analysis shows that the delay bound given by DTMA is lesser and stricter than that provided by 802.11e. Chapter 7 discusses the implementation details of our scheme. It gives the minutes of the various functions we have implemented and the changes in Network Simulator code. Chapter 8 does an evaluation of the proposed scheme DTMA. It gives the simulation results and compares the DTMA results with the existing ones. We also have some of the graph results in the appendix. Finally, we discuss the future work and give the conclusion of our work.

Chapter 2

QoS Architectures and Real Time Protocols

Quality of Service is the suite of new technologies and standards to provide resource assurance and service differentiation to a particular application or user type [26] . As time sensitive applications are being widely used over networks, the need for minimum service guarantees is necessary. In the first part of this chapter we discuss the various QoS architectures deployed in the wired and the wireless networks. The remaining part gives a brief insight into the various real time protocols serving time sensitive applications.

The following QoS architectures will be discussed during the course of this chapter :

1. Integrated Services (IntServ)
2. Differentiated Services (DiffServ)

Also, the following Real Time Protocols will be discussed :

1. Session Initiation Protocol (SIP)
2. Real-Time Transmission Protocol (RTP)

2.1 QoS Architectures

Both the architectures to be discussed here were motivated from the Internet scenario. Efforts are being made to adapt these architectures to the different forms of networks like wireless.

2.1.1 Integrated Services

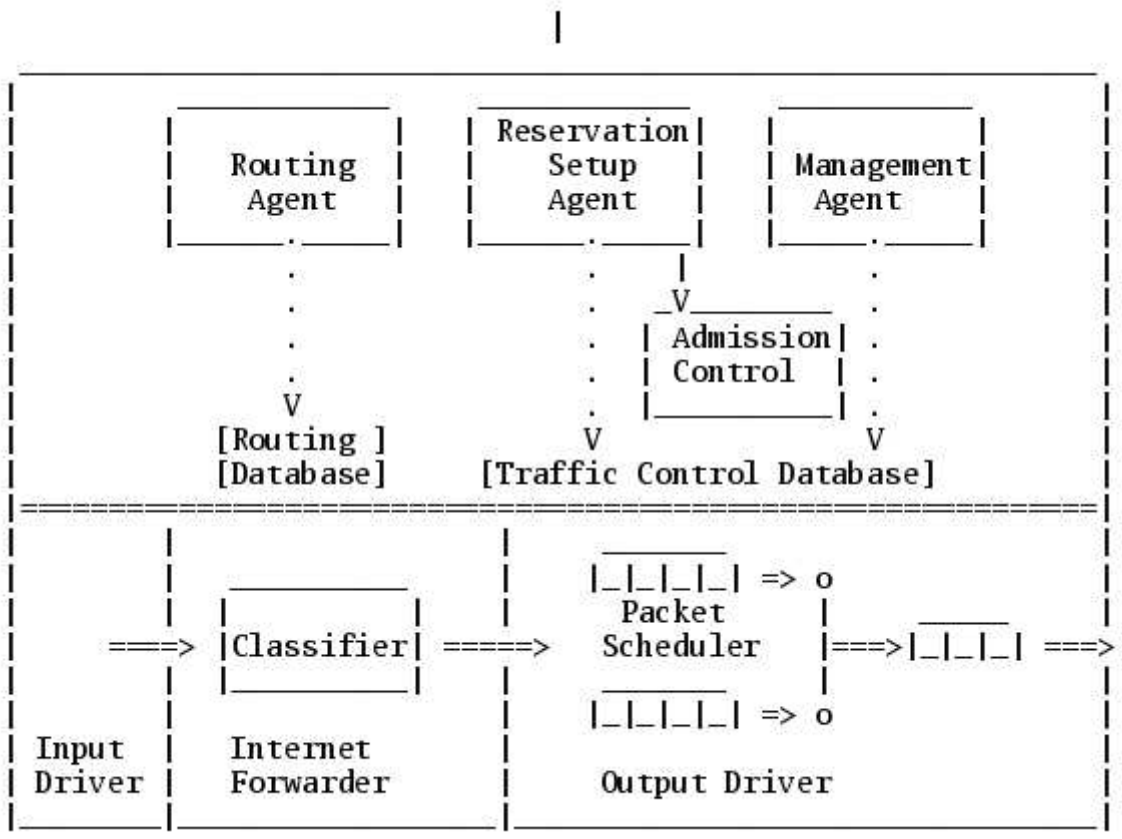


Figure 2.1: IntServ Architecture

This model provides individualized QoS guarantees to individual application sessions [26]. The philosophy of this model is that, to provide special QoS for specific user packet streams or flows, the router should have the resource reservation capability. Two key features of IntServ architecture are:

a. Reserved Resources :

   A router is required to know what amounts of its resources (buffers, link bandwidth) are already reserved for ongoing sessions.

b. Call setup:

   Applications requiring QoS guarantees must set up the paths and reserve resources before transmitting their data. RSVP signaling protocol is used for setting up paths and

reserving resources. This is also called as Call Admission process. Traffic Specification (Tspec) and Reservation Specification (Rspec) defines the specific QoS being requested by a connection.

Functional Components

Integrated Services is implemented by four Components.

1. Signaling Protocol (RSVP).
2. Admission Control routine.
3. Classifier.
4. Packet Scheduler.

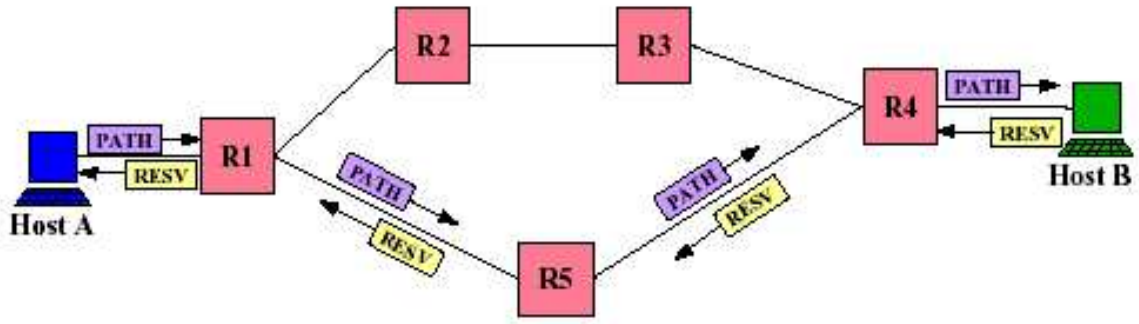


Figure 2.2: RSVP Signalling

Resource Reservation Protocol (RSVP) is used to reserve resources. The sender sends a PATH message to the receiver specifying the characteristics of the traffic. Every intermediate router along the path forwards the PATH message to the next hop determined by the routing protocol. Upon receiving a PATH message, the receiver responds with a RESV message to request resources for the flow. Every intermediate router along the path can reject or accept the request of the RESV message. If the request is rejected, the router will send an error message to the receiver, and the signaling process will terminate. If the request is accepted, link bandwidth and buffer space are allocated for the flow and the related flow state information will be installed in the router. The Admission Control routine decides whether

a request for resources (link bandwidth and buffer space) can be granted or not. Classifier classifies the packets based on the content of multiple fields such as source address, destination address, TOS byte, protocol ID, source port number, destination port number and put the packet in a specific queue based on the result. The packet scheduler then schedules the packet accordingly to meet its QoS requirements.

Services Classes

In addition to the Best Effort service, IntServ provides two more service classes:

a. Guaranteed Service.

b. Controlled Load Service.

Guaranteed Service provides strict bounds on the queuing delays that a packet will experience in a router. Controlled Load Service is provided for those applications requesting reliable and enhanced best effort service. A session receiving controlled load service will receive a quality of service close to the QoS that same flow would receive from a lightly loaded network element.

IntServ Issues

i. Too Complex architecture. Per Flow Information has to be maintained at each node. The amount of state information stored at each node is proportional to the number of flows.

ii. All routers must implement RSVP, Admission Control, Multifield Classification and Packet Scheduling.

iii. It doesn't support gradual deployment.

iv. This is not a scalable architecture.

v. It supports a limited number of services classes.

2.1.2 Differentiated Services

Differentiated Sevices (DiffServ) architecture was developed as an alternative resource allocation scheme [26]. In DiffServ, the users' traffic is divided into small number of forwarding classes. For each class, the amount of traffic pushed into the network is limited at the edge of the network. By changing the total amount of traffic allowed in the network, service providers can adjust the level of resource provisioning and hence control the degree of resource assurance to the users.

This architecture is based on a simple model where traffic entering a network is classified and possibly conditioned at the boundaries of the network, and assigned to different behavior aggregates. It is composed of a number of functional elements implemented in network nodes, Per Hop Behaviours (PHBs), packet classification functions and traffic functions like metering, marking, shaping and policing. The main advantage of this architecture is per application flow or per customer state need not be maintained within the core of the network.

Differentiated Architecture Goals

1. To allow Internet Service Providers (ISPs) to provide different service levels.

2. Keep the forwarding path simple. The packet processing at the interior of the network must be as simple as possible.

3. Push complexity to the edges. Any per flow activity should be kept at the edge of the network. Core must be as simple as possible.

4. Flexible service model. Ability to charge differently for different services. Based on the payment people will get service.

5. Provide a service that avoids assumptions about the type of traffic using it.

Traffic Classification

The Classification module contains a Classifier and a Marker. Classifier is an entity that selects packets based on the content of packet headers according to predefined rules. Marker sets the DS code point in a packet based on defined rules.

There are basically two types of classifiers.

1. Behavior Aggregate Classifier

2. Multi field Classifier

Behavior Aggregate Classifier selects packets based only on the contents of the DS field. A Multi field classifier selects packets based on the contents of source address, destination address, DS field, protocol ID, source port and destination port.

Location of Classifiers and Conditioners

Traffic classifiers and conditioners are usually situated with DS ingress and egress nodes where traffic goes across domains. They may also be present within the interior of a DS domain or within a non-DS capable domain. A domain which is capable of implementing differentiated services as described in the DiffServ architecture is called a DS capable domain. Otherwise it is called non-DS capable domain.

1. Within the source domain :

   Source domain is the domain in which packet originates. Pre-marking can be performed within this domain. Pre-marking is setting the DS code point before the packet leaves the source domain. Pre-marking allows the source domain to classify packets based on the local policies.

2. At the boundary of a DS domain :

   Traffic streams may be classified, marked and conditioned at the boundary of the two domains. The Service Level Agreement (SLA) between the domains should specify which domain has responsibility of marking the packets with appropriate DS codepoints. At ingress, node traffic is checked with the TCA in accordance with local policy.

3. In Interior DS nodes:

   DiffServ implements complex functionalities at the boundary nodes. Core router classifies packets based on DS codepoint and applies the forwarding treatment based on that value.

2.2 Real Time Protocols

2.2.1 Session Initiation Protocol (SIP)

SIP was designed as a multimedia protocol that could take advantage of the architecture and messages found in popular Internet applications. By using a distributed architecture with universal resource locators (URLs) for naming and text-based messaging, SIP attempts to take advantage of the Internet model for building VoIP networks and applications [20]. In addition to VoIP, SIP is used for videoconferencing and instant messaging. SIP is a textual client-server based protocol and provides the necessary protocol mechanisms so that the end user systems and proxy servers can provide different services.

As a protocol, SIP only defines how sessions are to be set up and torn down. It utilizes other IETF protocols to define other aspects of VoIP and multimedia sessions, such as Session Description Protocol (SDP) for capabilities exchange, URLs for addressing, Domain Name Systems (DNSs) for service location, and Telephony Routing Over IP (TRIP) for call routing. SIP is addressing neutral.

SIP is independent of the packet layer and only requires unreliable datagram service, it provides its own reliability mechanisms. SIP supports 5 facets of establishing and terminating multimedia communication.

1. User Location : Determination of the end system.
2. User Availability : Determination of willingness of the called party to engage in communication.
3. User Capabilitites : Determination of media and the media parameters to be used.
4. Session Setup : Ringing, establishment of session parameters to be used at both the calling and called parties.
5. Session Management : This includes transfer and termination of sessions, modifying session parameters and involving services.

SIP itself does not offer conference control services such as floor control and does not prescribe how a conference is to be managed. Just need to initiate session that uses other control protocol. It cannot and does not provide resource reservation capabilities. SIP is

an alternative to H.323. It has been proved that there are concerns about H.323 regarding scalability and extensibility. SIP uses client-server architecture. The SIP user agents are the SIP phones. The SIP servers can be either proxy or redirect servers. Also the SIP servers can be made stateful or stateless. There are the entities called as the 'SIP Gateways'. These are the gateways to PSTN for telephony internetworking and to H.323 for IP telephony.

Working of SIP

Sip works as follows:

Callers and callees are identified by SIP addresses. When making a SIP call, a caller first locates the appropriate server and then sends a SIP request. The most common SIP operation is the invitation. Instead of directly reaching the intended callee, a SIP request may be redirected or may trigger a chain of new SIP requests by proxies. Users can register their location with SIP servers. SIP messages can be transmitted either over TCP or UDP. Much of the message syntax and header field are similar to HTTP. Messages can be request messages or response messages.

Request Messages

The format of the Request packet header is shown in the following figure.

Figure 2.3: Request Message Header

1. Method : The method to be performed on the resource. Possible methods are Invite, Ack, Options, Bye, Cancel, Register.

2. Request-URI : A SIP URL or a general Uniform Resource Identifier, this is the user or service to which this request is being addressed.

3. SIP version : The SIP version being used; this should be version 2.0.

The format of the Response message header is shown in the following figure.

1. SIP version : The SIP version being used.

Figure 2.4: Response Message Header

2. Status-code : A 3-digit integer result code of the attempt to understand and satisfy the request.

3. Reason-Phrase : This is the third field of the response message format.

2.2.2 Real Time Transmission Protocol (RTP)

Both TCP and UDP have their inherent drawbacks and cannot be used directly for streaming multimedia data. So, higher level protocols have been developed to alleviate the associated problems.

RTP [19] is an application level protocol, that provides end-to-end delivery services for data with real time characteristics. It provides ordering and timing information and is used over UDP. RTP actually consists of two protocols, RTP for real time transmission of data packets and RTCP for QoS monitoring and conveying participants' identities in a session. Because TCP is slow due to the 3-Way Handshake, it is not suitable for real time transport. So, UDP is used, but it is inherently unreliable and does not support retransmission. Still, UDP has features like multiplexing and checksum services, that favor real time services. To get over these drawbacks, RTP is proposed at the application level.

RTP itself does not provide any mechanisms to ensure timely delivery or other QoS guarantees, but relies on lower layer services to do so. It also does not guarantee or prevent out-of-order delivery, nor does it assume that the underlying network is reliable and delivers packets in sequence. The data arriving is continuously monitored by RTCP which informs the RTP layer to adjust its coding and transmission parameters for the proper delivery of data. For example, if RTP detects severe packet loss, it may inform RTP layer to slow down the rate of transmission. Thus, the data transmission is monitored by sending the RTCP packets between the participants. There is a chance that the RTCP traffic may exceed the RTP traffic. To keep this situation under check, the RTCP packet transmission rate is changed dynamically as a function of the number of participants, RTCP packet size,

RTCP bandwidth. According to the standard, 20% of the session bandwidth is allocated to RTCP. For small sessions, there would be an RTCP transmission for every 5 seconds. Thus, as mentioned earlier, RTP is responsible for the real time transmission of packets and RTCP is responsible to provide feedback on the quality of data distribution. Other RTCP functions include carrying transport level id for RTP source, called canonical name, which is used to synchronise the audio and video.

The next chapter discusses the 802.11 and 802.11e MAC in detail. It portrays the enhancements of 802.11e over the legacy 802.11 MAC.

Chapter 3

IEEE 802.11 - WLAN Standard

After having seen the various QoS architectures and real time protocols, in this chapter we briefly look at the MAC of one of the most popular WLANs - 802.11b. After that we look at the QoS provided in the 802.11 domain through the detailed analysis of 802.11e MAC.

In 2000, the 802.11b became the standard wireless ethernet networking technology. The WiFI organization was created to ensure interoperability between 802.11b products. The standard gives specifications for the Medium Access Control (MAC) and Physical Layer (PHY).

The primary goal of the standard was the specification of a simple and robust WLAN that offers time bounded and asynchronous services. The MAC should also be able to operate with multiple physical layers, each of which might have a different medium sense and transmission characteristics. Additionally the WLAN should have the support for power management, handling of hidden nodes etc.

3.1 Architecture

The basic service set (BSS) is the fundamental building block of the IEEE 802.11 architecture. A BSS is defined as a group of stations that are under the direct control of a single scheme (either DCF or PCF). The geographical area covered by the BSS is known as the basic service area (BSA), that is analogous to a cell in a cellular communications network. Conceptually, all stations in a BSS can communicate directly with all other stations in a BSS.

An ad hoc network is a grouping of stations into a single BSS for the purposes of com-

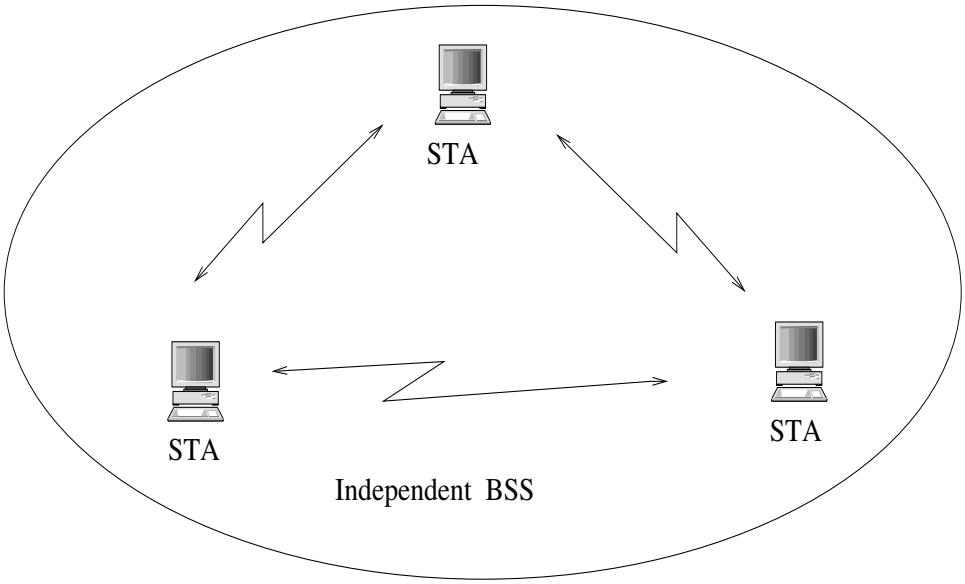


Figure 3.1: 802.11 AdHoc Network

munications without the need for a centralized controlling entity. The IEEE 802.11 standard specifies the term Independent BSS (IBSS) for such an adhoc network. Figure 3.1 shows an IBSS. Any station can establish a direct communication session with any other station in the BSS, without the requirement of channeling all traffic through a centralized entity Access Point (AP).

Infrastructure networks are established to provide wireless users with specific services and explicit channel access. Infrastructure networks are established using APs. The AP is analogous to the base station in a cellular communications network. Range extension is provided by introducing the integration points necessary for network connectivity between multiple BSSs. The entity thus formed is called an Extended Service Set (ESS). The ESS consists of multiple BSSs that are integrated together using a common distribution system (DS). The DS is kind of a backbone network that is responsible for MAC-level transport of MAC service data units (MSDUs). The DS, as specified by IEEE 802.11, is implementation independent i.e it could be either wired or wireless. Thus, the DS could be a wired IEEE 802.3 Ethernet LAN, IEEE 802.4 token bus LAN, IEEE 802.5 token ring LAN, Fiber Distributed Data Interface (FDDI), Metropolitan Area Network (MAN) or another IEEE 802.11 wireless medium.

An ESS can also provide gateway access for wireless users into a wired network such

as the Internet. This is accomplished via a device known as a portal. The portal is a logical entity that specifies the integration point on the DS where the IEEE 802.11 network integrates with a non-IEEE 802.11 network. If the network is an IEEE 802.X, the portal incorporates functions which are analogous to a bridge; that is, it provides range extension and the translation between different frame formats.

3.2 802.11 MAC Operation Modes

IEEE 802.11 MAC [8] features two mode of operations:

1. Distributed Coordinating Function (DCF) and

2. Point Coordinating Function (PCF).

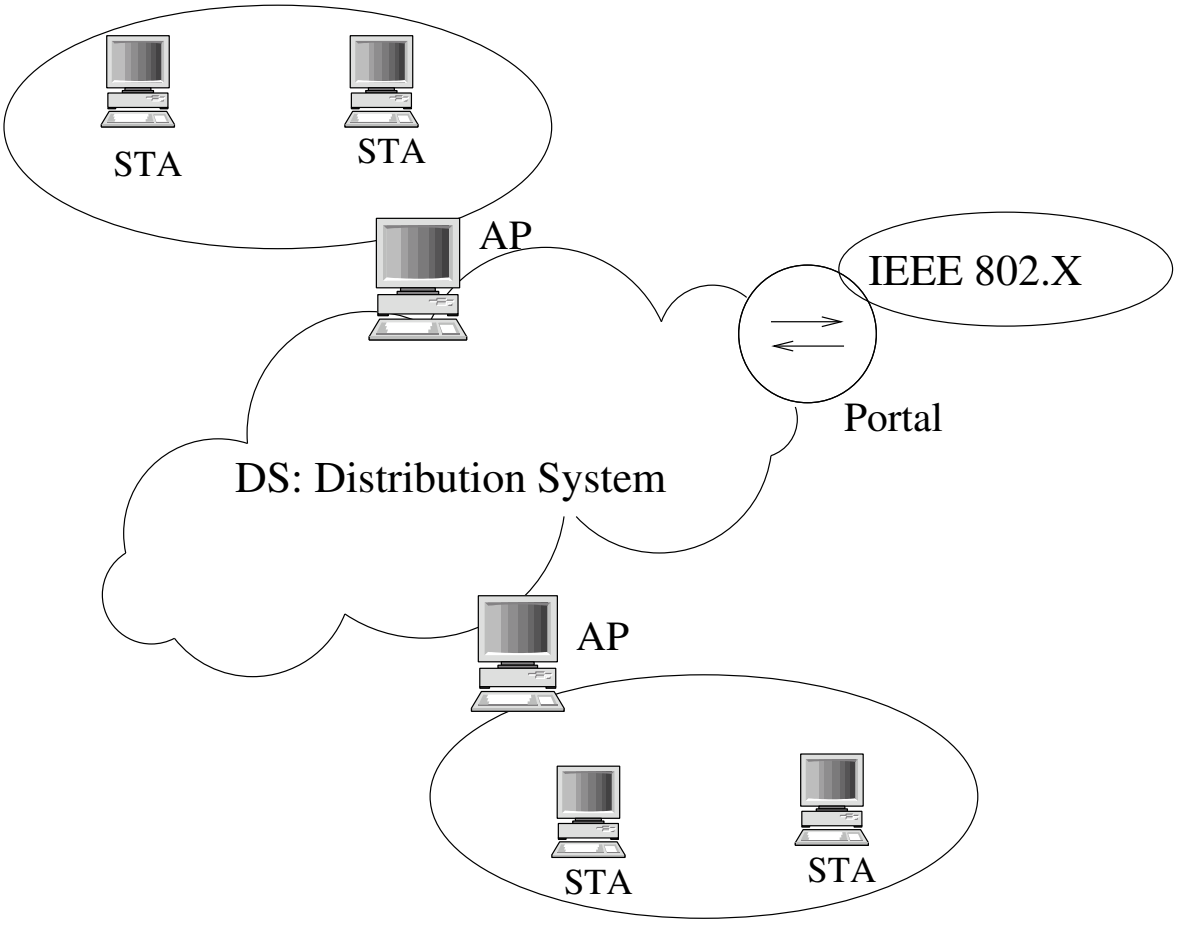


Figure 3.2: 802.11 Infrastructure Based Network

3.2.1 Distributed Coordination Function

The IEEE 802.11 draft standard describes mandatory support for asynchronous data transfer and optional support for distributed time-bounded services (DTBS). Asynchronous data transfer refers to traffic that is relatively insensitive to time delay. This asynchronous data transfer happens through the Distributed Coordination Function, wherein all users with data to transmit have an equally fair chance to transmit.

DCF is CSMA/CA based random access protocol that uses random backoff to avoid collision. It is mandatory for the DCF to be implemented in all stations, for use within both IBSS and infrastructure network configuration. It operates solely in the ad hoc network (IBSS) , and either operates solely or coexists with the PCF in an infrastructure network and supports contention based services. Contention services imply that each station with a MAC Service Data Unit (MSDU) queued for transmission must contend for access to the

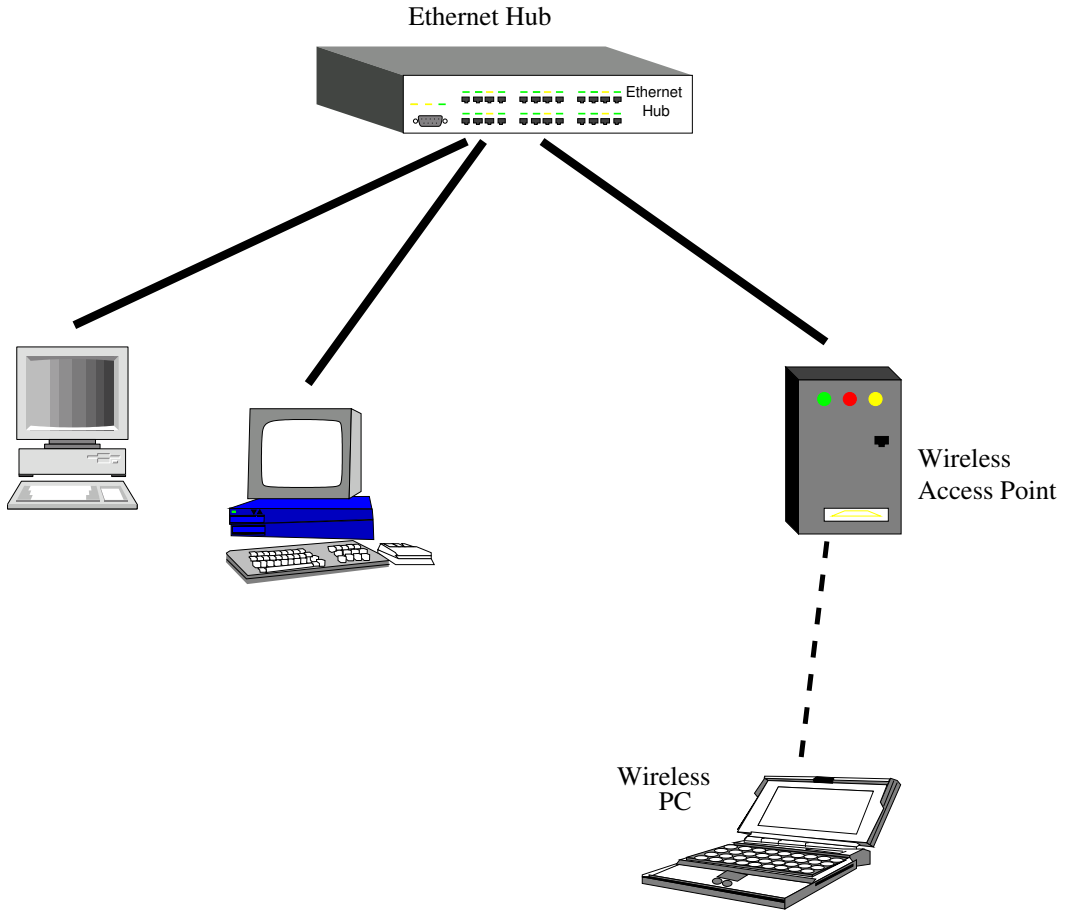


Figure 3.3: 802.11 Wireless Access Point

channel and, once the MSDU is transmitted, must recontend for access to the channel for all subsequent frames. Contention services promote fair access to the channel for all stations.

Station that needs to transmit data, first senses the carrier. In IEEE 802.11, carrier sensing is performed at both, the air interface, referred to as physical carrier sensing and at the MAC sublayer, referred to as virtual carrier sensing. Physical carrier sensing detects the presence of other IEEE 802.11 WLAN users by analyzing all detected packets, and also detects activity in the channel via relative signal strength from other sources. A virtual carrier-sense mechanism is also provided by the MAC. This mechanism is referred to as the network allocation vector (NAV). The NAV maintains a prediction of future traffic on the medium based on duration information that is announced in RTS/CTS frames prior to the actual exchange of data. The duration information is also available in the MAC headers of other frames sent during the CP. The Duration ID field defines the period of time that the medium is to be reserved to transmit the actual data frame and the returning ACK frame. Stations in the BSS use the information in the duration field to adjust their network allocation vector (NAV), which indicates the amount of time that must elapse until the current transmission session is complete and the channel can be sampled again for idle status. The channel is marked busy if either the physical or virtual carrier sensing mechanisms indicate the channel is busy.

DCF uses RTS/CTS mechanism to reserve channel when packet size is above a limit, the RTSthreshold. It reduces the hidden terminal effect.

3.2.2 Point Coordination Function

The PCF provides contention-free frame transfer. The Point Coordinator (PC) is collocated with the AP. All stations inherently obey the medium access rules of the PCF, because these rules are based on the DCF, and they set their Network Allocation Vector (NAV) at the beginning of each Contention Free Period (CFP). The operating characteristics of the PCF are such that all stations are able to operate properly in the presence of a BSS in which a PC is operating, and, if associated with a point-coordinated BSS, are able to receive all frames sent under PCF control. It is also an option for a station to be able to respond to a contention-free poll (CF-Poll) received from a PC. A station that is able to respond to CF-Polls is referred to as being CF-Pollable, and may request to be polled by an active PC. CF-Pollable stations

and the PC do not use RTS/CTS in the CFP. When polled by the PC, a CFPollable station may transmit only one MPDU, which can be to any destination (not just to the PC), and may piggyback the acknowledgment of a frame received from the PC using particular data frame subtypes for this transmission. If the data frame is not in turn acknowledged, the CF-Pollable station cannot retransmit the frame unless it is polled again by the PC, or can decide to retransmit during the Contention Period (CP) that is the DCF. If the addressed recipient of a CF transmission is not CF-Pollable, that station acknowledges the transmission using the DCF acknowledgment rules, and the PC retains control of the medium. A PC may use contention-free frame transfer solely for delivery of frames to stations, and never to poll non-CF-Pollable stations.

Polling Procedure

PCF sends a CF-Poll to at least one station during each CFP when there are entries in the polling list. During each CFP, the PC issues polls to a subset of the stations on the polling list in order by ascending Association-Id value.

A station indicates its CF-Pollability using the CF-Pollable subfield of the Capability Information field of Association Request and Reassociation Request frames. If a station desires to change the PC's record of CFPollability, that station performs a Reassociation. During association, a CF-Pollable station may also request to be placed on the polling list for the duration of its association, by setting the CF-Poll Request subfield in the Capability Information field. If a CF-Pollable station desires never to be placed on the polling list, that station performs Association with CF-Pollable subfield false. Never being polled is useful for CF-Pollable stations that normally use power-save mode.

Thus, DCF rules are followed in the CP and in the CFP PC polls the nodes one by one and grants access to channel. New stations that need to get added to the poll list, send request in the CP.

3.3 802.11e - Bringing QoS in WLANs

3.3.1 Motivation for Quality of Service

With the increase in the popularity of the wireless devices and the availibility of real time applications over wireless networks, it has become important to extend the QoS guarantees provided in the wired domain to the wireless domain. An end user given some guarantees of delay and bandwidth over the normal wired networks would want to receive similar guarantees when he / she switches from the wired network to some wireless counterpart. This switching should be transparent to the end users. With the availability of applications like VoIP, videoconferencing over wireless networks in the market, guaranteeing finite delays has become a necessity for the service providers.

The enhanced wireless MAC for 802.11 is not suitable to give QoS guarantees because there is absence of any kind of prioritization between the nodes or the flows. The MAC lacks the ability to differentiate between traffic streams. All the packets are at the same level in both PCF and DCF. So there is no scope for service differentiation in the current MAC. As an example, a bursty email traffic stream might choke out and cause delays in a real time video stream.

3.3.2 Architecture

There is not much difference in the architecture of MAC of the legacy 802.11 and 802.11e. A station operating under 802.11e is called as an enhanced station or QoS-enabled Station (QSTA). 802.11e has a BSS that is termed as QoS-Supporting BSS (QBSS), that consists of the QoS-compliant Hybrid Coordinator (HC) and QSTAs.

An ad hoc network is a grouping of stations into a single QBSS for the purposes of communications. There is no AP involved. The operation is controlled by Enhanced Distributed Coordination Function (EDCF). As in the legacy predecessor, there is the mode where there is the central entity present that controls the channel access. The central entity is termed as the Hybrid Coordinator (HC) in the 802.11e jargon.

3.4 802.11e Working

802.11e has two phases of operation within the superframe:

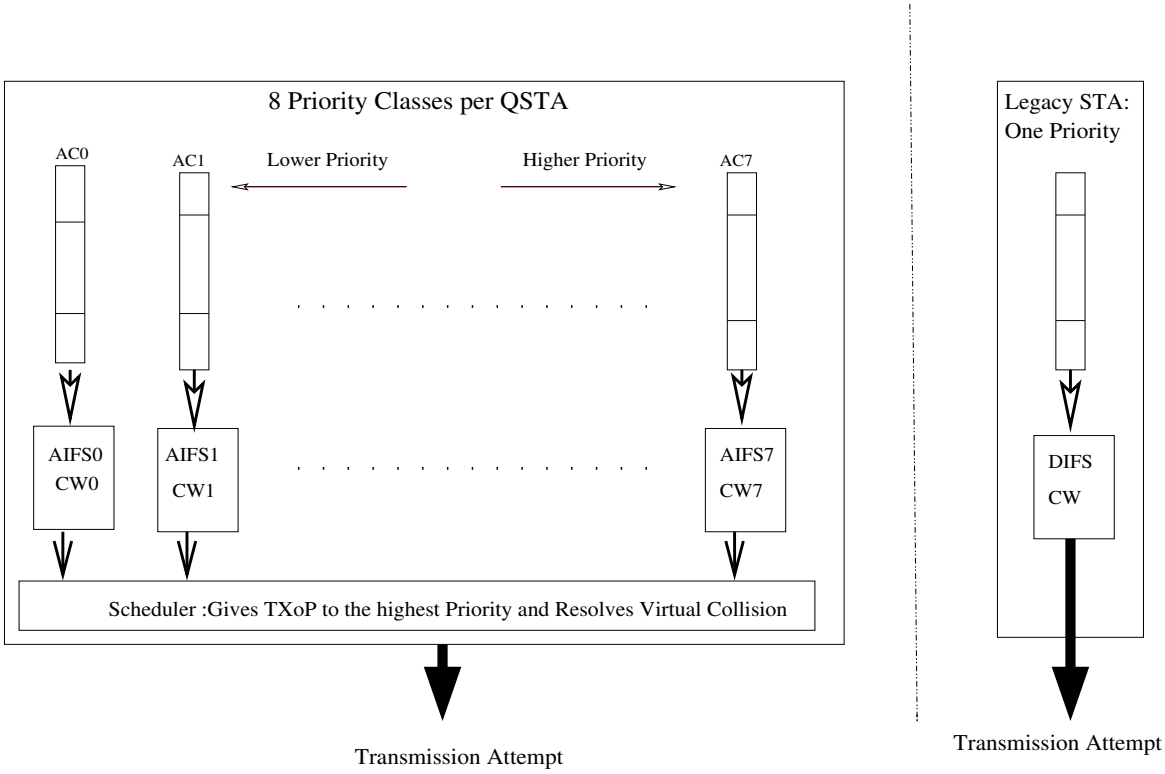


Figure 3.4: EDCF Vs. DCF

1. Enhanced Distributed Coordination Function

2. Hybrid Coordination Function

These modes make use of the concept of priority classes called Traffic Classes (TCs) . Currently the standard supports upto eight TCs [9]. We will see both the schemes in detail.

3.4.1 Enhanced Distributed Coordination Function

Enhanced Distributed Coordination Function (EDCF) is DCF with some elements of the MAC parmeterized. It is part of the Hybrid Coordination Function (HCF) to be discussed later. It is a flow based approach in which there is a queue for each of the TCs. These queues act as virtual stations and their priority parameters decide the ability of that particular queue to transmit, thereby inducing classification. The scheduler, if finds that the backoff counters of two classes have reached zero simultaneously, internally treats it as a virtual collision and then according to the priority of the queue, schedules one with the higher priority for transmission

while the other one backs off as if there is a collision. Thus, MSDUs are delivered through multiple backoff instances within one station. In CP each station contends independently for a Transmission Opportunity (TXoP) and starts backoff after detecting the channel idle for a period of an Arbitration Interframe Space (AIFS). The AIFS is atleast DIFS and can be increased as per TC-specific characteristics.

Figure 3.4 compares the 802.11e EDCF architecture that supports queue based differentiation with original one queue based DCF access mechanism. Each queue signifies one TC, with each having different priorities based on the AIFS timing and backoffs.

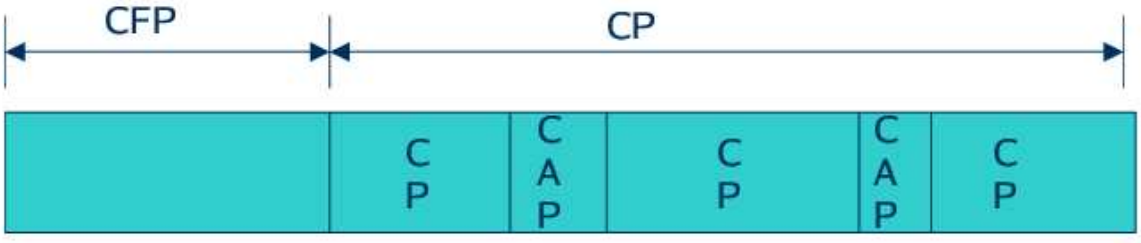


Figure 3.5: Transmission Opportunities in HCF

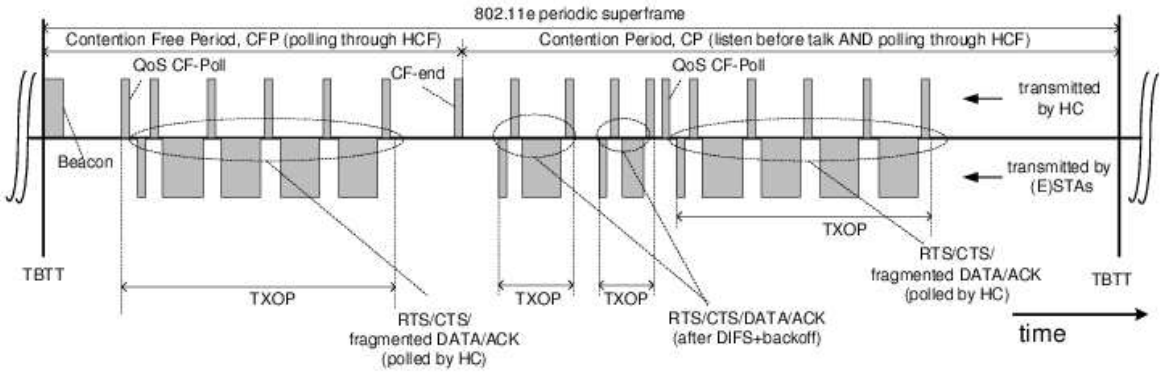


Figure 3.6: Typical 802.11e Superframe

3.4.2 Hybrid Coordination Function

Hybrid Coordination Function is an extension to the PCF idea of the 802.11. The superframe starts with a contention free period and is followed by the contention period, where the channel access is distributed and contention based. However like the PC in the legacy 802.11,

there is a Hybrid Coordinator (HC) that can initiate a CFP even in the DCF to provide the committed QoS guarantees. The HC can allocate TXoPs to itself to initiate MSDU transfers , after detecting the channel idle for PIFS amount of time. In a given CFP, the working is like the normal legacy approach, with the HC polling the stations that need to be given the QoS guarantees. A station has to respond to a poll in SIFS amount of time, failing to do which, the HC takes control of the channel and sends either the poll to another station in its polling list or sends a CF-End frame that signifies that the CFP has ended.

The figure 3.5 shows the Controlled Access Period (CAP) in the CP, where the HC takes control of the channel.

Thus the main change proposed in the HCF based approach is that the HC has the total control over the channel and can give prioritized access. There is a new QoS field added to the conventional MAC frame through which the stations convey the queue status, delay information to the HC. Based on this information the HC takes the scheduling decisions, giving TXOPs in the CFP to the stations in need.

The figure 3.6 shows a typical 802.11e superframe. It shows EDCF and HCF alternating. Also shown is the HC taking control of the channel and giving TXoPs to itself or some other node.

Chapter 4

Related Work

In this chapter we present some of the works done in the 802.11 domain to provide better services. Before the introduction of the 802.11e standard, the research community tried to optimize the PCF and DCF time values. Stress was also given on attempts to make better scheduling algorithms that would have lesser time complexity.

The survey done in this work presents the various optimizations done for the configurable parameters of the 802.11e standard. We also provide an insight into the analysis and evaluation of 802.11e.

4.1 Dynamic Adaptation of PCF and DCF

A. Goliya et al [1] in their work propose a scheme wherein the 802.11 MAC adapts itself to the varying loads. The main motivation for their work is that PCF and DCF do not perform equally well under different traffic scenarios. DCF performs well for low traffic and PCF works well for high traffic and large sized networks by reducing the contention. Their Dynamic Switching Protocol (DSP) monitors the network traffic and decides a threshold. Based on this it switches the mode of dynamically to either PCF or DCF as the case might be, to suit the network load and traffic at that point.

The authors also present a scheme that optimizes polling in the PCF mode. Polling has heavy overheads when only a small number of the CF-Pollable nodes actually have data to send. The scheme proposed by them, the Priority Round Robin Scheduling (PRRS) maintains an active list of nodes that have data to send [1]. These nodes are polled with higher priority.

The nodes that do not have data to send are added to the passive list. Further, the authors also present a CFP Repetition Interval Adaptation Algorithm that adapts the value of the CFP repetition interval to optimize the channel usage and increase network throughput.

The DSP scheme can indeed perform very well and increase the throughput. But the authors used static configuration parameters for switching between PCF and DCF. There should a mathematical function at the AP to decide the threshold to switch between the modes. Also the scheme PRRS uses priority lists for polling. Similar approach is used in the 802.11e Hybrid Coordination Function (HCF) that is a variant of the PCF.

4.2 Performance Comparison of PCF and DCF

In their work, Andreas Kopsel et al [3] compare the performance of PCF and DCF under various traffic loads. Using simulations they show that DCF is useful when the traffic is low and the number of mobile nodes is less. On the other hand, PCF should be used for the scenario of high load and large number of nodes.

Similar to the work in the earlier section, the authors propose a scheme to reduce the polling overheads. The mobile nodes have to provide the Access Point with suitable polling information. They propose two schemes :

1. Explicit Signalling : In the CP, a short frame is sent to the AP by the mobile station indicating its transmission requirements.

2. Implicit Signalling : Transmission requirements are conveyed by the mobile node to the AP through the most recently transmitted frame. The *more* flag of the IEEE 802.11 data frame is used.

The authors used simulations to prove the results.

In the proposal made, the explicit signalling scheme will have non-trivial overhead on the DCF part. In the event of high number of mobile nodes, this overhead will be quite huge. Also this necessitates a new frame type to be added. Implicit signalling using the *more* flag is inherently used by 802.11.

4.3 Blackburst

To improve the performance of real time streams in wireless LANs, Sobrinho and Krishnakumar proposed a scheme called *Blackburst* [21]. The main goal of Blackburst is to minimize the delay for real time traffic, and it is somewhat different from the other schemes since it imposes certain requirements on the traffic to be prioritized. Blackburst requires that all high priority stations try to access the medium with constant intervals, t_{sch} (this interval has to be the same for all high priority stations). Further, Blackburst also requires the ability to jam the wireless medium for a certain period of time. When a high priority station wants to send a frame, it senses the medium to see if it has been idle for a PIFS time and then sends its frame. On the other hand, if the medium is found busy, the station waits until the channel has been idle for a PIFS and then enters a black burst by jamming the channel for a period of time. The length of the blackburst is determined by the time the station has been waiting to access the medium and is calculated as a number of black slots. After transmitting the black burst, the station listens to the medium for a short period of time less than black slot to see if some other station is sending a longer black burst. That would imply that the other station has been waiting for a longer time, and thus should access the channel first. If the medium is idle, the station will send its frame, otherwise it will wait until the medium becomes idle again and enter another black burst contention period. By using slotted time and imposing a minimum frame size on real time frames, it can be guaranteed that each black burst contention period will result in an unique winner.

After the successful transmission of a frame, the station schedules the next access instant of transmission, t_{sch}, seconds in the future. This has the nice effect that real-time flows will synchronize and share the medium in a TDM fashion. This means that unless there is a transmission by a low priority station when an access instant for a high priority station occurs, very little blackbursting will need to be done once the stations have synchronized. Low priority stations use the ordinary DCF access mechanism.

4.4 AEDCF

Romdhani et al [15] propose an Adaptive Enhanced Distributed Cordination Function (AEDCF). This scheme propses a Service Differentiation Scheme for 802.11 WLAN. The authors claim

that the static reset of the CW in the EDCF significantly reduces the throughput. Their scheme gives prioritised access to the channel. The difference is that they consider the application requirements and network traffic to adjust the size of the CW for a particular service class.

The results show that their goodput is 25% more than EDCF. Thus the authors achieve delay differentiation and low jitter with increased channel utilization.

4.5 Performance Evaluation of 802.11e

Literature provides pointers to many works involving the performance evaluation of 802.11e. Yunli Chen et al [25] provide the delay analysis for 802.11e EDCF. They support their analysis with simulation results. They also propose a priority scheme that allows the user to continuously send real time packets.

Priyank Garg et al [14] also conduct a performance evaluation of 802.11e through computer simulations. Their results prove that HCF does reduce channel contention and allows better channel utilization. The authors state that HCF and EDCF are very sensitive to protocol parameters and conduct experiments to optimize these parameters.

Another computer simulation based evaluation is provided by Antonio Grilo and Mario Nunes [4]. They evaluate using a scenario of 802.11b/e access to an IP core network through an AP in an infrastructure WLAN. They provide the comparisons of average delay and throughput in the presence of varying bursty data sessions and real time flows.

We verify the simulation results of our proposed scheme DTMA against the results of this work.

Thus, not much work has been done for the analysis of the new 802.11e standard. Also, most of the stress is on the optimization of EDCF part as it involves more probabilistic approaches. In the next chapter we propose our enhancement to the HCF part and do an anaysis of it against the EDCF in the chapter after that.

Chapter 5

Proposed Scheme : DTMA

This chapter gives the details of our proposed scheme, DTMA - Dynamic Time Division Multiple Access, for providing better QoS guarantees.

5.1 Motivation for a new scheme

We observe that the HCF mode polling is similar to that of legacy 802.11. The data transmission time for a minimum packet size of 512 bytes for a station is approximately 100μsec and the transmission time for the poll or acknowledgement is almost 44μsec [13]. In the worst case, that is when data packet size in small, the system spends almost half of the time sending just the control information. Even when the packet size is as large as 2000 bytes, the ratio of data to control frame transmission time is almost 6 : 1, which becomes a substantial overhead when the number of priority flows is high. So, keeping polls and acknowledgements implicit or removing them altogether would mean saving 44μsec per station. This implies having more time for useful data transmission, increasing the overall performance of the system.
Also, EDCF does not give any guarantees for prioritized traffic. At high load there are high number of collisions even for flows with high priority [5].

Providing better QoS than existing schemes is our main goal. For achieving it we propose to use a dynamic slot allocation version of *TDMA approach* for transmission of high priority data. We eliminate the process of polling each station by having each node transmit in a defined time slot, the allocation of which is dynamic. Individual acknowledgements are

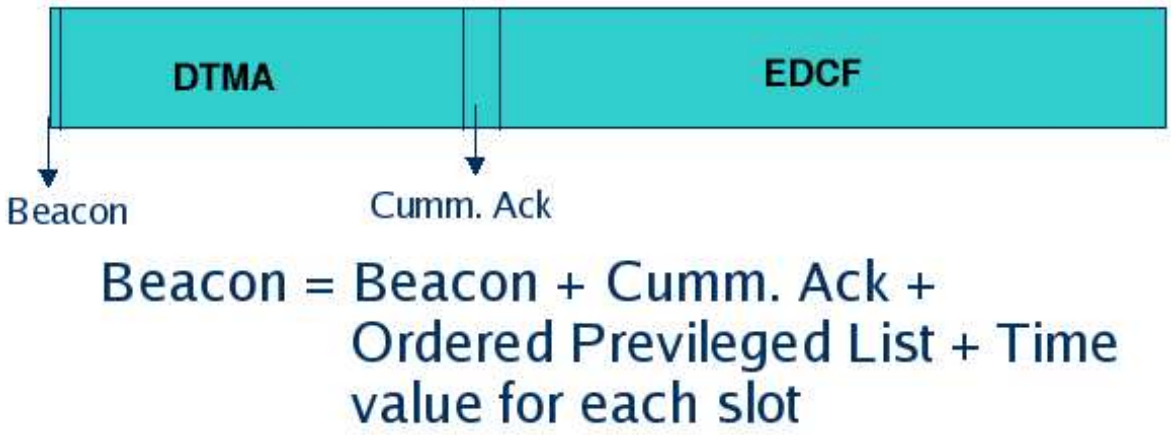


Figure 5.1: DTMA Superframe

removed by using a single cummulative acknowledgement.

5.2 Scenario Considered

We propose a scheme, DTMA, that has been developed for a particular kind of network like an airport kind of wireless LAN. It can consist of a number of wireless nodes and Access

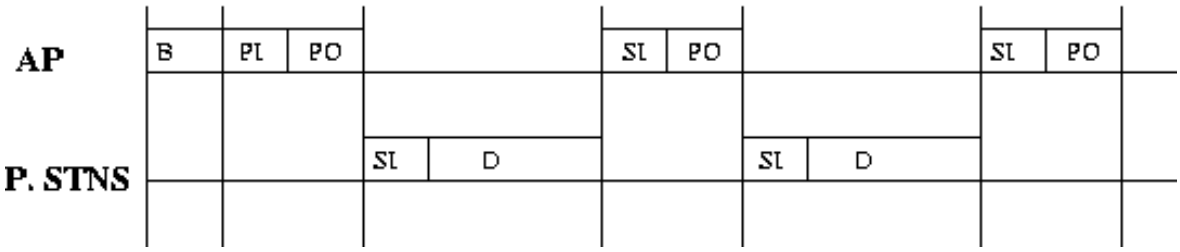


Figure 5.2: 802.11e Transmission

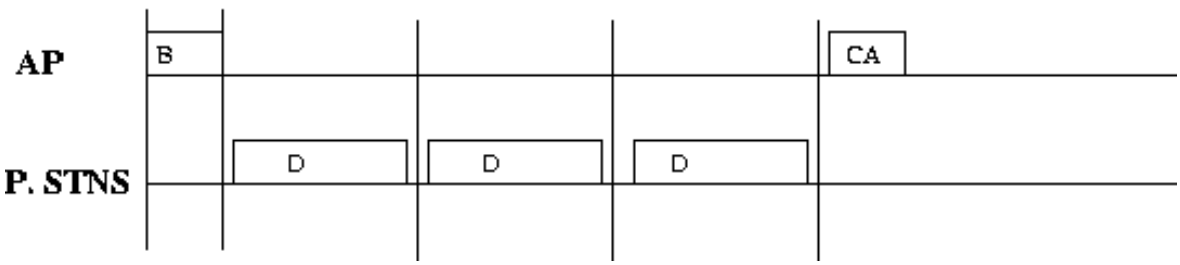


Figure 5.3: DTMA Transmission

Points. A particular AP has a number of nodes in its domain. The APs can communicate with each other. Also there is a chance of a node moving from one AP to another, but we will consider that case at a later stage. Currently we assume that the nodes are not mobile. We claim to provide QoS guarantees to the wireless stations, but these are from the node to the AP i.e the node will be transmitting the data (if it has any) to the AP and also receive data from the AP, in a finite, deterministic, guaranteed amount of time. The QoS guarantees are restricted to the local wireless network and not beyond that.

5.3 Dynamic Time Division Multiple Access (DTMA)

DTMA, Dynamic Time Division Multiple Access, is a variant of TDMA to expunge control information preceding data transmissions in HCF. As discussed in the earlier section, using explicit polls / acknowledgements can have much overhead when the number of pollable stations is high. We have already shown that the ratio of data to poll / acknowledgement transfer time could be as much as 2.5 : 1. Reducing this extra overhead means there is more transmission time for the data, less delay for the stations which in turn means better QoS guarantees to the stations. Thus, we propose a modified TDMA approach for HCF to provide stricter upper bound on the delays for real time traffic.

As shown in the figure 5.1, the CFP starts with a beacon as in the legacy 802.11. However, the beacon frame is modified by us. Along with the usual fields in the beacon, there are a few additional fields to make the DTMA successful. The new additions to the beacon are :

- *Cummulative Acknowledgement* : This is the combined acknowledgement for all the nodes that transmitted in the previous superframe.
- *Ordered Privileged List* : This is the list of nodes as discussed in the earlier part.
- *Time Value of Slot* : This is the value of the time for each DTMA slot.

DTMA has time slot value, changing dynamically depending on parameters like load on the network, average delay and throughput. We use these for making the decision of slot size. A time slot value that optimizes these weighted parameters is chosen. This value adapts to the system so that delay and throughput are balanced and the overall performance of the

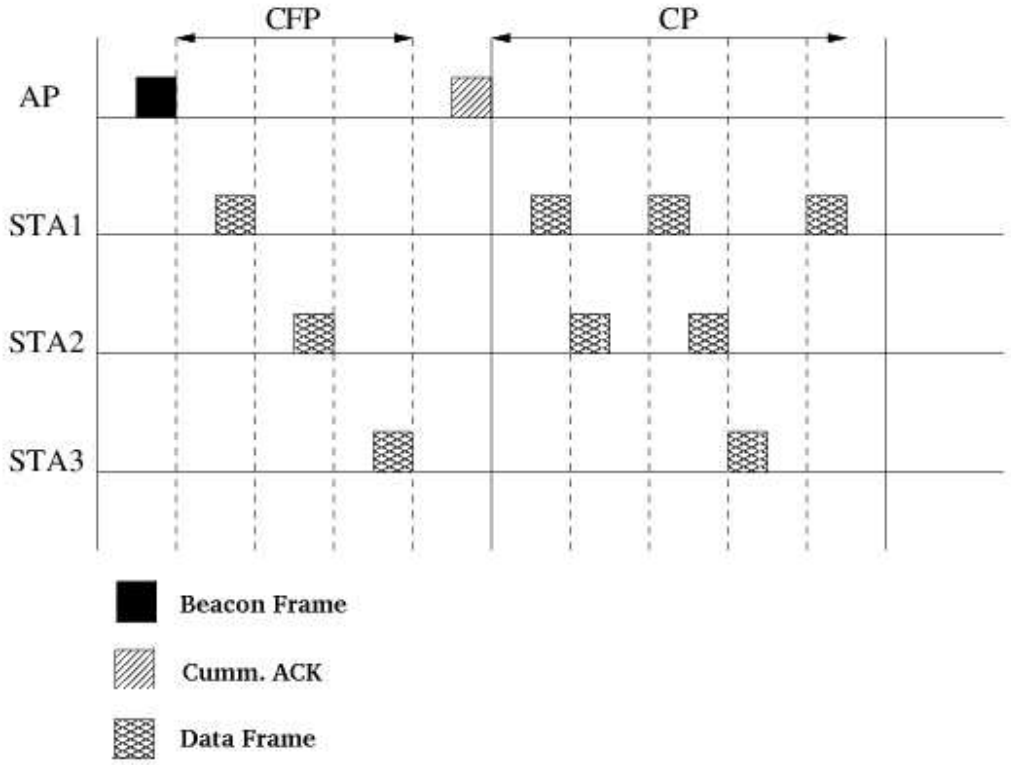


Figure 5.4: Transmission Sequence for DTMA

system is improved. Currently, we manually vary the slot sizes for our experiments. A more complex function can be used for the same.

In addition to the value of time slot, the number of time slots also vary dynamically. If the AP finds that privileged node does not have data to send, it does not allot the node any slot for transmission. Thus we remove the inherent drawback of the TDMA approach of wasting time slots, when there is no data to be sent.

The general sequence of events that take place in DTMA are :

1. As shown in figure 5.1, the frame starts with the reception of the beacon. All the nodes get the beacon frame. All nodes now know the *Time Value of Slot* for the dynamic slot scheme. Also the *Ordered Previleged List* field in the beacon is checked by all the privileged nodes, that gives each the position in the sequence of transmissions.

   From the position and the time slot value, each node calculates the exact time at which

it has to transmit data. Thus no polling is required.

2. After this the nodes start transmitting at their respective time instants. The AP keeps receiving the data, but does not send the acknowledgement to the sender node. Instead it marks the acknowledgement in a *Cummulative Acknowledgement*. If data is received without any errors, 1 is inserted in the position corresponding to the node in the list, else a 0 is inserted.

3. After the last node in the *Ordered Privileged List* has finished its transmission, the AP sends the cummulative acknowledgement for all the nodes. The privileged nodes check for the bit in their corresponding position in the list. If it is '1' the node removes the sent packet from its buffer and prepares to send the next one, else it prepares to retransmit the old packet that was not received properly.

4. Following the acknowledgement is the EDCF part. All the nodes compete as per EDCF backoff and transmit rules.

5. After the EDCF, the superframe ends and the beacon is transmitted indicating the start of a new superframe. The *Cumulative Acknowledgement* in the beacon though redundant, is kept for reliability. If the original acknowledgement is erroneous, the Cummulative Acknowledgement in the beacon can be used to avoid retransmissions.

Figure 5.4 shows the transmission sequence of the proposed scheme.

5.4 DTMA: Protocol Specifications

5.4.1 Functional Description of the Access Point

The figure 5.5 shows the flow chart of operations to be done by the Access Point. Following are the important functions carried out by the Access Point.

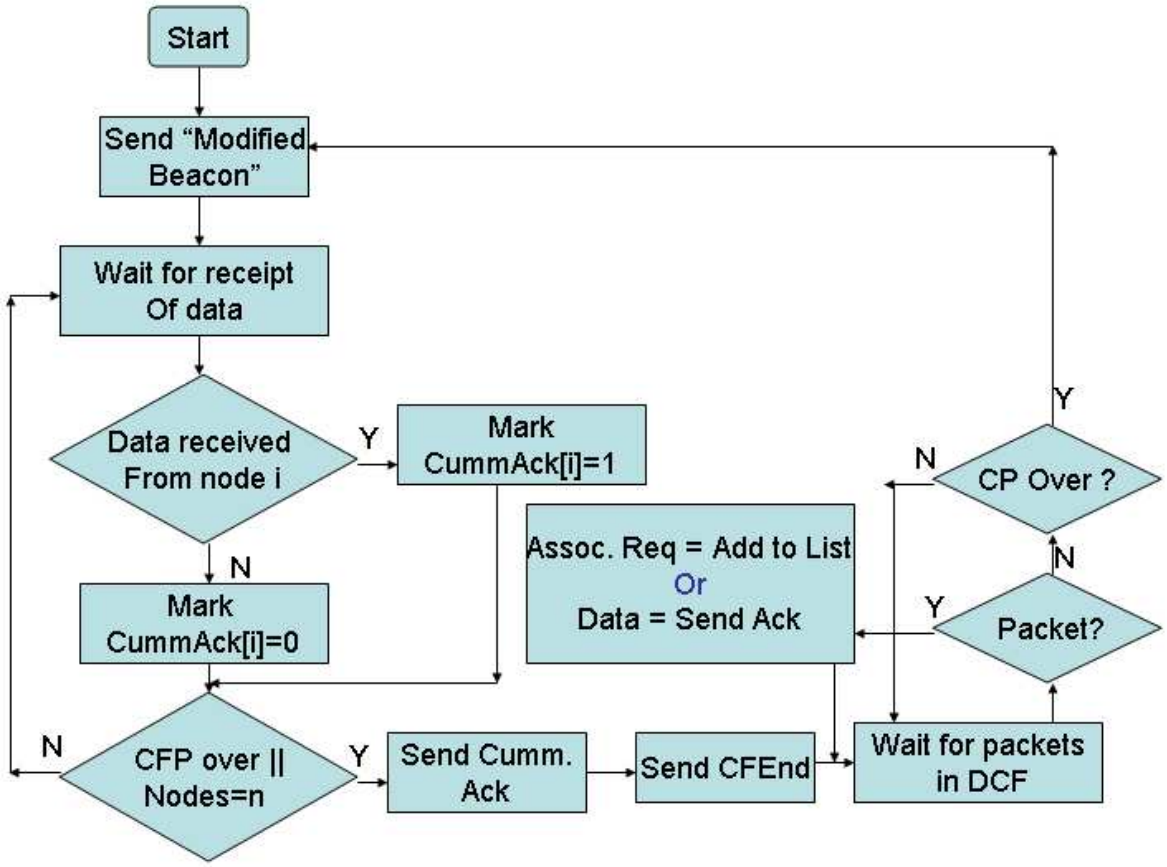


Figure 5.5: Access Point Flow Chart

1. Before the start of each frame and at the end of *CFP* the AP has to send the Beacon and CFEnd respectively.

2. Event : Reception of Association / Dissociation request.

   1: **if** frame_type == Association AND is_Privileged == TRUE **then**
   2: Add the node to the privileged_list ;
   3: **end if**
   4: **if** frame_type = Dissociation **then**
   5: Remove the node from the privileged_list ;
   6: **end if**

3. Event : Reception of Data from the node.

   1: **if** frame_type == Data **then**

2: Find node position i in the privileged list.

3: Set Cumm_Ack[i] = 1.

4: **end if**

4. Event : Reception of acknowledgement from node.

1: **if** frame_type == Ack **then**

2: Remove the corresponding packet from buffer.

3: **end if**

5. Event : Reception of NULL frame from the node.

1: set_mark_flag = 1;

2: mark_flag_count++ ;

3: **if** mark_flag_count >= k **then**

4: set not_to_schedule_flag = 1;

5: **end if**

The variables are set such that, if a node sends NULL frame for a fixed number of times k, the AP interprets it as, the node being either idle or sleeping and hence does not schedule it for the superframes after that time.

Thus the system adapts to the conditions when the stations don't have data to send and dynamically controls the number of time slots available.

6. Event : An out of list node sends a join request.

1: **if** frame_type == join_request **then**

2: set not_to_schedule_flag = 0;

3: Grant a time slot in the next superframe;

4: **end if**

7. If both AP and node have data for each other, then the AP schedules such that both the node and AP transmit in alternate superframes. The scheduling is done depending on the queue status of the node and the AP.

8. One important function of the AP is to calculate the time slot. As discussed earlier, using given parameters, the AP calculates the optimal value of the time slot, using which the overall performance of the network enhances.

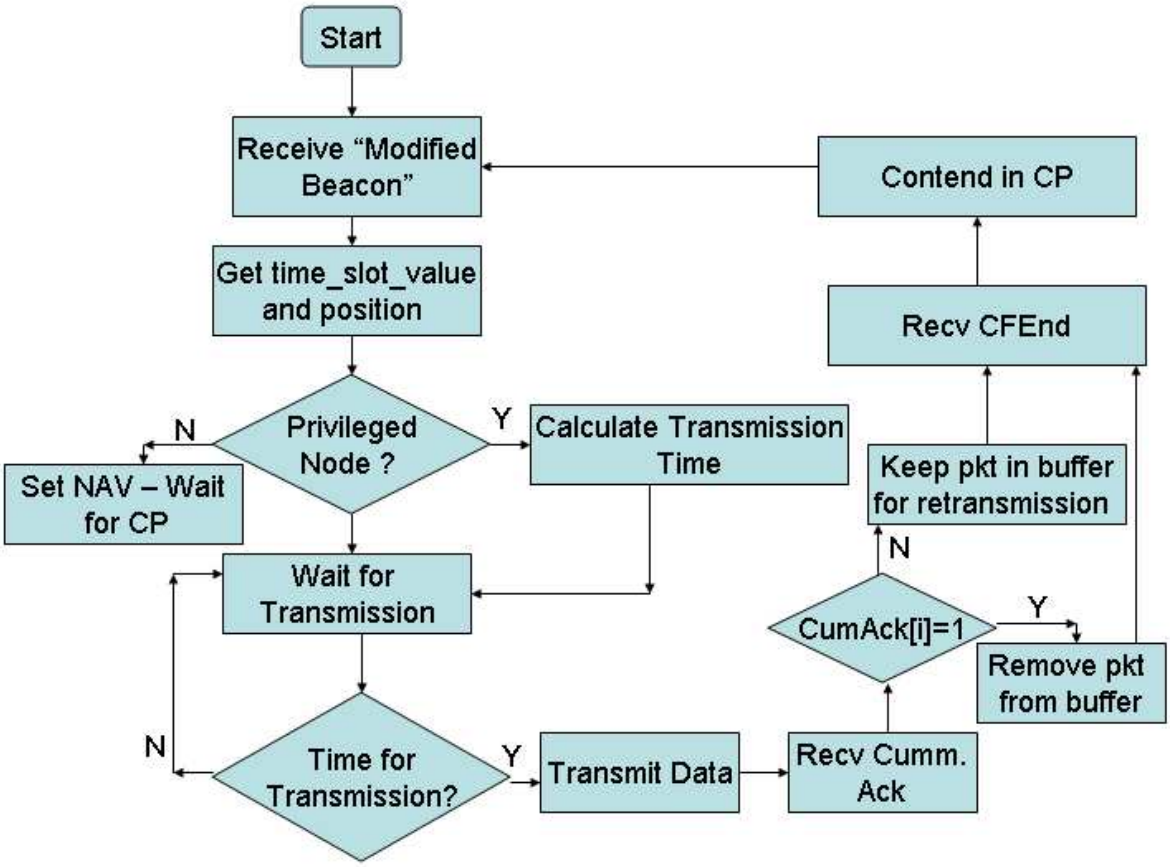


Figure 5.6: Node Flow Chart

Figures 5.2 and 5.3 shows the comparison of working of 802.11e vis-a-vis DTMA. We can see the reduction of the control information in DTMA which contributes to lesser delay and better throughput.

5.4.2 Functional Description of the Node

The modification that has to be done at the node is that the node has to interpret the modified beacon, and according to the values sent, transmit in its given slot. Also the node needs to be changed for interpretation of the *Cummulative Acknowledgement*

Figure 5.6 shows the flow chart of operations of a node compliant to DTMA.

In the next chapter, we analyze 802.11e and DTMA using probability models and find the expressions for the worst case delays for both.

Chapter 6

Analysis of EDCF and DTMA

In this chapter we do the delay analysis of DTMA and 802.11e. We derive the closed form expressions for the worst case delays for both and then compare them. We also show using this analysis that the delay bound for DTMA is lesser and stricter as compared to that of 802.11e.

6.1 Delay Analysis of 802.11e

Yunli Chen et al have done an analysis of 802.11e EDCF [25]. The average delay for any real time flow in EDCF is given in [24]. This analysis considers only EDCF for calculating the average delay. For the complete analysis of 802.11e we need the average delay over both HCF and EDCF. Our handling of the analysis is thus different. We calculate the delay for the privileged node in HCF and EDCF individually and then calculate the average delay using these two, over the entire superframe.

In HCF, all the nodes in the polling list are polled separately and given a chance of data transmission in the *CFP*. In the *CP*, all nodes contend for the channel access and the node with the smallest backoff value transmits first. So to calculate the average delay, for the 802.11e HCF and EDCF together, we calculate the delay suffered in each part of the superframe and then take a weighted average.

Let the number of privileged nodes be x and the unprivileged be y. The backoff interval for legacy DCF is assumed to be [1,N]. For ease of analysis, we assume that each node requires the same amount of time for data transmission, m. The number of transmission time slots

in *CFP* and *CP* are *t1* and *t2* respectively. The superframe time is T units.

For tractability in the analysis, we divide the *CFP* and *CP* time into a number of slots in which data transmissions can take place. So,

$$t1 = \frac{CFPTime}{m}$$

Similarly,

$$t2 = \frac{CPTime}{m}$$

We first start by calculating the average delay a privileged high priority node suffers in the *CP*.

We note that, for the privileged node to transmit successfully in EDCF, it must select the lowest random backoff. That is, if it has a random backoff value i, all the remaining nodes must select a value that is greater than i.

Thus, the probability that the privileged node chooses a random backoff value i is uniform :

$P1 = \frac{1}{N}$

Now, the probability that all other privileged nodes (x-1) choose a random backoff that is greater than i is :

$$P2 = (\frac{N-i}{N})^{x-1}$$

Also the constraint that needs to be satisfied is that no other non-privileged node of the given y, does not select a random backoff less than or equal to i. Thus, the probabilty of a non-privileged node selecting a backoff greater than i becomes :

$$P3 = (\frac{N-i}{N})^{y}$$

If EDCF is assumed,

$$P3 = (\frac{N-i}{N+k})^{y}$$

where, k is a positive integer that ensures that the non-privileged nodes have a larger interval to select a backoff ([0,(N+k)]).

The total probability becomes,

$P = P1 * P2 * P3$

Summing over all possible values of i in the given interval,

$$P = \sum_{i=1}^{N} \frac{1}{N} (\frac{N-i}{N})^{x-1} (\frac{N-i}{N+k})^{y}$$

Now, the probability that a privileged node is able to transmit successfully in the kth attempt is :

$$P_k = (1-P)^{k-1} * P$$

Now the probability is calculated such that this kth attempt happens before the end of the superframe and in the *CP*.

$$P(k <= T) = \sum_{k=t1+1}^{t1+t2} (1-P)^{k-1} * P$$

In the given scenario, the worst case would be when all the nodes, both privileged as well as the non-privileged have packets outstanding in the queue. The expectation that this privileged node transmits in the *CP* after k attempts is:

$$E(k) = \sum_{k=t1+1}^{t1+t2} P_k * k$$

The average delay for a privileged node is:

$$D = E(k) * m$$

D is the average delay in the worst case that a high priority privileged node suffers in the *CP*. The closed form expression for the total average delay over the entire superframe is now calculated.The privileged node gets polled in the *CFP* for data transmission followed by an acknowledgement for the same by the AP.

The probability of transmission in the *CFP* is :

$$P_{CFP} = P_{Poll} * P_{Data} * P_{Ack}$$

where, P_{Poll} and P_{Ack} are the probabilities of successful receipts of poll and acknowledgement frames respectively. P_{Data} is the probability of successful transmission of data packet.

We assume that once a poll is received the data probability of data transmission is 1. So, $P_{Data} = 1$. Hence,

$$P_{CFP} = P_{Poll} * P_{Ack}$$

The delay encountered due to the overhead of polls and acknowledgements is :

$$P_{poll} * P_{ack} * (T_{overhead})$$

where,

$T_{overhead} = T_{Poll} + T_{Ack} + 2 * T_{SIFS}$

T_{Poll} = Time for Poll

T_{Ack} = Time for Acknowledgment

T_{SIFS} = Short Interframe Space

Let $D1 = P_{poll} * P_{ack} * (T_{overhead})$

As mentioned above, the average delay for the privileged node over the entire superframe is total number of data transmissions per superframe time.

$$AvgDelay = \frac{T}{\frac{t2*m}{D} + \frac{t1*m}{D1}} \tag{6.1}$$

Thus the average delay depends on the number of nodes of the privileged and unprivileged type, the value of backoff interval and the delays due to the control information.

6.2 Delay Analysis of DTMA

There is a bounded delay requirement for the high priority nodes, that generate real time, delay sensitive traffic. We derive a closed form expression for this bounded delay.

Using our model DTMA, we prove the following claims :

1. The maximum delay that any high priority, privileged node suffers is less than or equal to *superframe time (T).*
2. The delay for real time traffic is strictly bounded and we give the expression for this bounded delay.
3. The bounded delay given by DTMA is stricter and lesser than that given by 802.11e.

In HCF, all the high priority privileged nodes transmit their data in the assigned time slot in the *CFP*. Now as the CFP is repeated after every superframe, it is trivially proved

that the *maximum delay* that is suffered by a privileged node is T. The first claim, (1) is proved here.

Now let us calculate the average delay for a high priority node. As said above, the privileged nodes have a fixed time slot to transmit in the *CFP*. As with the analysis of 802.11e, we calculate the delay suffered in the *CP* and then calculate the average along with the *CFP*. Following is the delay calculated for the privileged node in the *CP*.

For ease of comparison, the parameters used for analysis of DTMA are the same as those for 802.11e.

On lines of initial steps of derivation of (1), the probability of a privileged node selecting a random backoff i, that is the lowest among all the nodes is,

$$P = \sum_{i=1}^{N} \frac{1}{N} (\frac{N-i}{N})^{x-1} (\frac{N-i}{N})^{y}$$

Assuming EDCF,

$$P = \sum_{i=1}^{N} \frac{1}{N} (\frac{N-i}{N+k})^{x-1} (\frac{N-i}{N+k})^{y}$$

Now, the probability that a privileged node is able to transmit successfully in the kth attempt is :

$$P_k = (1-P)^{k-1} P$$

So, the probability of the privileged node transmitting in the *CP* is:

$$P(k <= T) = \sum_{k=t1+1}^{t1+t2} (1-P)^{k-1} P$$

For the privileged node, average number of transmissions it waits before it transmits would therefore be the expectation, which is:

$$E(k) = \sum_{k=t1+1}^{t1+t2} P_k * k$$

The worst case average delay for the privileged node is:

$$D = E(k) * m$$

The closed form expression for the total average delay over the entire superframe for DTMA is:

$$AvgDelay = \frac{T}{\frac{t2*m}{D} + 1} \tag{6.2}$$

We see, when compared with equation 6.1, the delay bound given by DTMA is stricter as well as lesser than that given by 802.11e HCF and EDCF. We have removed the overhead of the polls and acknowledgements and this has helped us to grant more time for the actual data transmissions and thereby reduce the delay and improve performance and throughput. Equation 6.2 proves the claims (2) and (3).

In the next chapter, we discuss about the implementation of DTMA in Network Simulator.

Chapter 7

Implementation Details

In this chapter we discuss the important code points and also the implementation issues that we faced. We also present the various decisions that we took and the justifications for the same.
We used Network Simulator 2 [10] for the computer simulations. We added the functionality for our proposed scheme DTMA and then carried out the experiments for the various scenarios. In the following sections we describe the important changes made by us during the course of this work.

7.1 Important Changes

Following are the important changes that we had to make to implement DTMA.

1. Timers :

   We needed to add two new timers to the MAC. They are:

   - QoS Timer : As per our scheme, we needed to wake up the particular privileged node to transmit in its given slot. So, there was requirement for a timer that would notify the privileged node the transmission time upon expiry.
   - CFEnd Timer : Our scheme requires that once the last node completes transmission in the given slot, all the other nodes should be notified. This is required as the transmissions might get finished well before the time for which the NAVs of

the nodes are set. So there should be a mechanism by which these nodes can start contending for channel access immediately after the receipt of the cummulative acknowledgement.

2. Privileged Bit :

   A new bit to recognise the node being a privileged or non-privileged, based on the association request parameter sent by the node was added. The AP checks this bit and then based on the value adds that node to the privileged list if need be.

3. AckList :

   Our scheme needs to have a cummulative acknowledgement for intimating the success of transmission. A new packet type is added called AckList that does this function. The AckList has the mapping of the node identifiers to the transmission bit set or unset depending on successful or unsuccessful transmission respectively. After the receipt of this, the node checks its corresponding bit and takes the decision whether to retransmit or not.

4. Class Ordered_List :

   We needed to maintain a list of the privileged nodes that could be given the chance to transmit in their respective time slots. We defined a class for this purpose. This class contains elements of type PCFPollable. There are member functions for the calculation of transmit time and slot size, inserting a node, deleting a node etc. The instantiation is done at the start before the transmission of the beacon.

   The following important member functions are appended:

   - insertNode : Upon the receipt of an association request from a privileged node, the privileged_bit is checked and if it is set, the node is added to the ordered list.
   - markDontSchedule : This function has a counter that keeps track of the nodes that do not send data frames in their scheduled transmit time periods. The counter is incremented after every successive non-transmitting time-slot. After the threshold number of 'marks', the deleteNode function is called.
   - clearDontSchedule : This function is complementary to the above *markDontSchedule* function. If a node is marked and in some subsequent time-slot it transmits a data frame, the 'mark' on the node is cleared and the counter is set to zero.

- deleteNode : As per the scheme, if the node is marked for a particular number of superframes, the protocol infers that the node is either sleeping or does not have data to send. The node is then removed from the list and not scheduled for the next superframe. So, no time slot is given to this node, hence avoiding the wastage of a time slot.
- get_time_slot_value : The DTMA scheme requires the calculation of the slot size value that would be optimal for the given network size and load. The *get_time_slot_value* function does that by taking the number of nodes operating in the CP and CFP and then calculating the optimal slot size. Currently, we follow a simplistic scheme. But a more powerful and effective function can be used for the same.

5. calculate_transmit_time :

   DTMA conveys the order in which the privileged nodes need to transfer during the CFP through the beacon. The elements call the function *calculate_transmit_time* to find the exact time at which it is scheduled to transmit. Based on the time-slot value and the position in the list, the calculation is done and the time is found.

6. Beacon Modifications :

   As discussed before, DTMA provides the ordered privileged list to the nodes using the beacon. Also, the *AckList* is transmitted for redundancy to overcome the possibility of the list getting garbled in the earlier attempt and hence the acknowledgements getting lost.

   We added these two list to the beacon. Also the time slot value is required to be sent to the privileged nodes, based on which the transmit times will be calculated by them.

7.2 Handling Issues

This section deals with the issues that we foresaw and how we handled these. We also briefly describe some of the interesting problems that we faced while working with NS and how we dealt with them.

The possible issues with the implementation of the scheme are:

Issue : The cummulative acknowledgement gets lost or garbled.
Solution : We again send the cummulative ack along with the beacon at the start of CFP to take care of this issue. This is redundant, but it provides some reliability.

Issue : The beacon gets lost.
Solution : The beacon is sent with more transmission power and hence the likelihood of it getting lost is very less. Even if does get lost, the first part of the superframe will be skipped for that round.

Issue : The nodes have multiple real time flows.
Justification : Our approach is station based, so these multiple flows would be taken care of internally at the node and finally the queue would be prepared based on the priority of each flow.

Issue : The node that is off the list wants to get back on the list.
Solution : The node contends for a slot in the CP and sends an *Association Request* to the AP, which schedules the node for the next superframe.

Issue : Synchronisation of clocks of the nodes and the AP.
Justification : As discussed earlier, the scenario we consider is a small LAN on an airport etc. The area considered is small and hence the problem of clock synchronisation would not arise.

Issue : The extra cost of newer cards.
Justification : The proposed 802.11e standard would also require new cards. So these cards could be incorporated with DTMA functionality.

7.2.1 Implementation Problems

We also faced some interesting problems while implementing the scheme in NS. Here are some of these problems.

1. **Packet Parameters Retrieval** :

   Our scheme required the packet to include some parameters to support QoS. We added the required parameter to the packet structure. But after compiling and then running, many of the parameters were not getting the values that were earlier being assigned.

   After a lot of debugging, we finally found the problem. We had overlooked the fact that the values of the packet were being retrieved using the offsets at which these parameters were in the packet structure. Due to the addition of new parameters. the offsets were being changed and hence the problem.

2. **Scheduler Exceptions** :

   As discussed earlier, our scheme needed the use of timers for various purposes. For some values of the timer expiry, the scheduler would complain and would not find some of the events valid. The scheduler would throw an exception and the program would halt.

   On detailed analysis, we could track the problem. The problem was, there are many timers running simultaneously. So, at times, the timers would get overlapped and before the expiry of one the timers the other would be invoked. The expiry of timers causes events to be scheduled at the Scheduler and so there were problems. So, before starting any timer, we needed to check if its state was busy. If it was, we stopped it and then started the required timer.

In the next chapter we present the computer simulations done by us and the results obtained.

Chapter 8

DTMA Evaluation

In this chapter we evaluate our proposed scheme DTMA. We discuss the various computer simulation scenarios that we considered for our experiments. We also compare our work against the existing schemes. We try to support the theoretical analysis with our simulation results in this chapter.

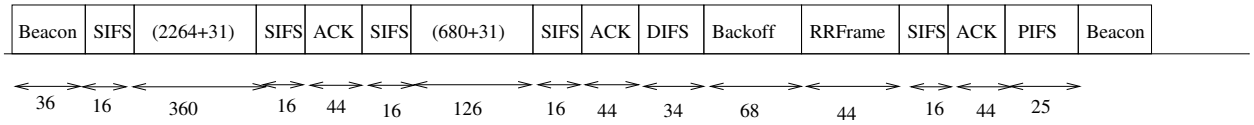


Figure 8.1: Channel Occupancy for 802.11e

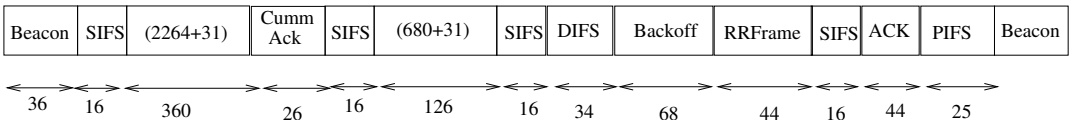


Figure 8.2: Channel Occupancy for DTMA

8.1 Throughput Analysis

Throughput Analysis of the 802.11e was done by James Crawford et al [11]. The experiments were conducted for two real time flows, HDTV and SDTV and a http data stream. The paper mentions the calculations done based on the sequence of data transmissions in *CFP* and *CP*.

We replaced Crawford's model by DTMA in the analysis. We removed the delay overheads of polling and acknowledgements, and subsequently added the delay induced by the cummulative acknowledgement of DTMA, we recalculated the throughput for the HDTV stream. Using DTMA, we had an enhancement of approximately 19% for the HDTV stream in throughput.

The improvement would be more prominent for more number of real time flows. For more number of flows, the control information transfer becomes very high and so the subsequent reduction in that improves the overall performance.

Figure 8.1 shows the channel occupancy for 802.11e. The times for PIFS, SIFS and DIFS are considered as the conventional values and the unit is μsecs. The beacon size is also considered as per the standard values and takes 36μsec. The example shows that because of the MAC and PHY overheads in the HCF, it can barely support only one HDTV connection and one SDTV connection simultaneously [11].

Figure 8.2 shows the corresponding channel occupancy for the DTMA scheme. As shown in the figure, the reduction of MAC overheads of poll and explicit acknowledgements for each of the nodes, there is an improvement of 18.67% in the throughput of the HDTV flow and approximately 16% for SDTV. Also, due to the reduction of these overheads, the total delay suffered by these nodes before they transmit in the next superframe also becomes lesser.

8.2 Simulation Results

8.2.1 Comparison with 802.11X

In their paper titled *Performance Evaluation of IEEE 802.11e*, Antonio Grilo and Mario Nunes [4] use simulations for evaluating the efficiency of 802.11e. We compare the results of the simulation runs for DTMA against the results of this paper.

The simulation setup for the experiments is as follows. Three types of traffic sources are considered : bursty data (HTTP sessions), VoIP and video. The bursty data source was replicated using Poisson Distribution and each data generated at the rate of 200Kbps. The audio source generated messages of size 60 bytes at the bitrate of 24 Kbps. The video source model used a H.263 video stream encoded to provide an output rate of 256 Kbps. This

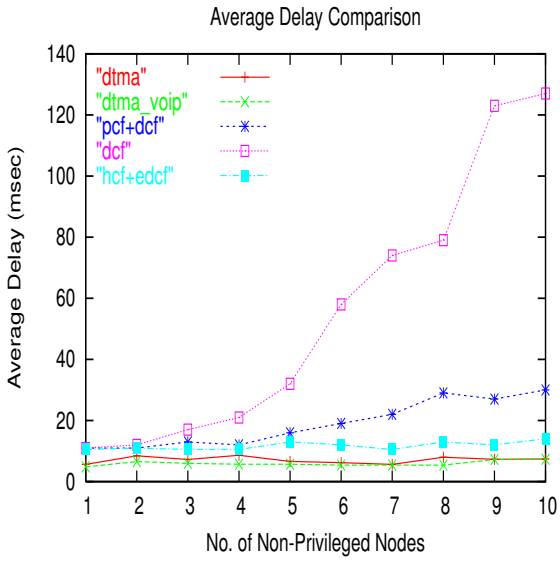


Figure 8.3: Average Delay Comparison

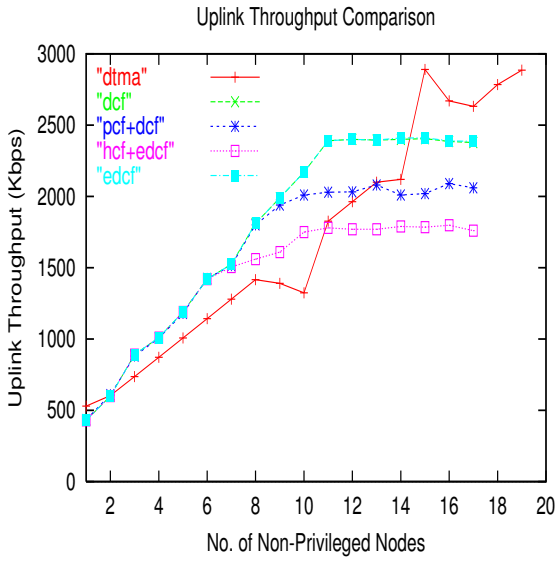


Figure 8.4: Throughput Comparison

stream is illustrative of the H.323 videoconference traffic. The VoIP and video sessions are high-privileged sessions.

We conducted our experiments in a similar environment and traffic flows were replicated. Figures 8.3 and 8.4 show the graphs comparing the average delay suffered and the uplink throughput of DTMA, compared with the other schemes. The various schemes considered for comparison are DCF, PCF and DCF, EDCF and HCF, and DTMA.

As shown in 8.3, the average delay suffered by a priority node running in an EDCF and HCF environment stays just below the 10msec mark. DTMA shows better performance for both video as well as the VoIP flows. As proved by our theortical analysis, DTMA performs better than HCF and EDCF as it reduced the considerable MAC overhead of polling and explicit individual acknowledgements. As we had predicted, the number of non-privileged nodes does not have an effect on the performance of DTMA. The average delay remains less than 7msec consistently for any number of flows. We can see that the VoIP traffic suffers still lesser delay due to the less data rate and also the considerable smaller packet size. Thus the average delay suffered using DTMA is lesser.

The throughput comparison shown in 8.4 also conforms to our prediction regarding the performance of DTMA. The performance enhancement is not much for lesser number of nodes,

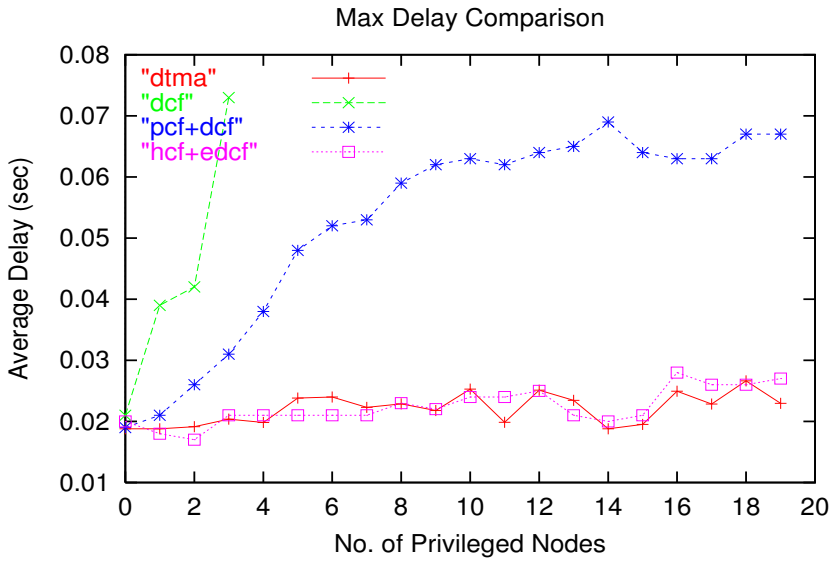


Figure 8.5: Maximum Delay Comparison

though the slope followed is similar compared to the other schemes. For the other schemes, the throughput stabilizes after the number of non-privileged nodes starts increasing over 8. But we can see that the throughput is increasing for DTMA till node 15. The stabilization occurs for DTMA when the number of nodes reach around 22. For more number of privileged nodes the performance improvement will be more significant as the amount of control traffic reduced will be more.

The figure 8.5 shows the comparison of the maximum delays suffered by a packet for various schemes. As seen, the maximum delay for DTMA is almost the same when compared to 802.11e. Again, the graph adheres to the claim that the number of other nodes do not affect the performance or the service received by a particular privileged, high priority node.

8.2.2 Optimal Slot-Size

As discussed earlier, we need to find the optimal time-slot size for the given network condition. We tried simulations with number of values. Finally the value of 1.5μ was found to be optimal for the network with upto 8 privileged nodes and 15 non-privileged nodes. The figures 8.6 and 8.7 show the average delay and throughput suffered when the number of privileged nodes is 7. As shown in 8.6, the average delay suffered for a node using slot sizes of 55μsecs is the lowest. Though it follows the similar curve as the graph of 1.5msec it does show lower delay

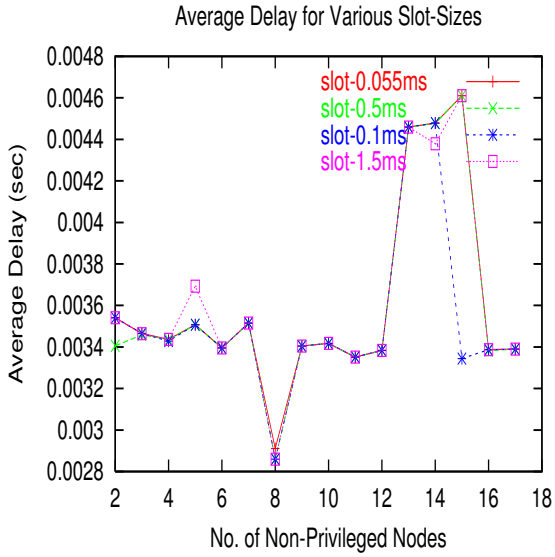


Figure 8.6: Average Delay for various Slot Sizes for 7 Priority Nodes

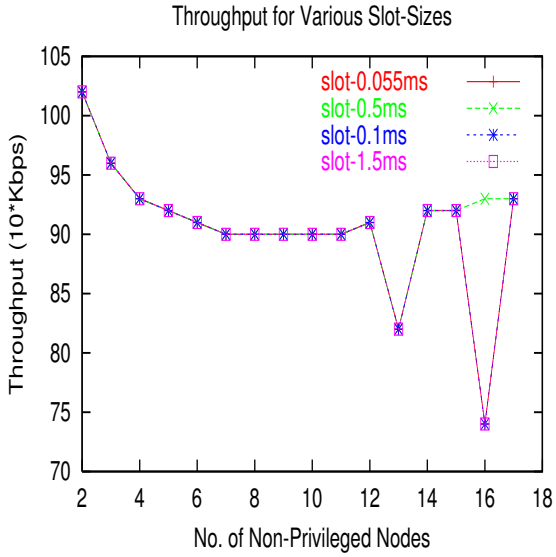


Figure 8.7: Throughput for Various Slot Sizes for 7 Priority Nodes

at quite few places. Looking at the throughput, we can see that because more time is used up by the bigger slots, the number of transmissions that can occur within a given time frame

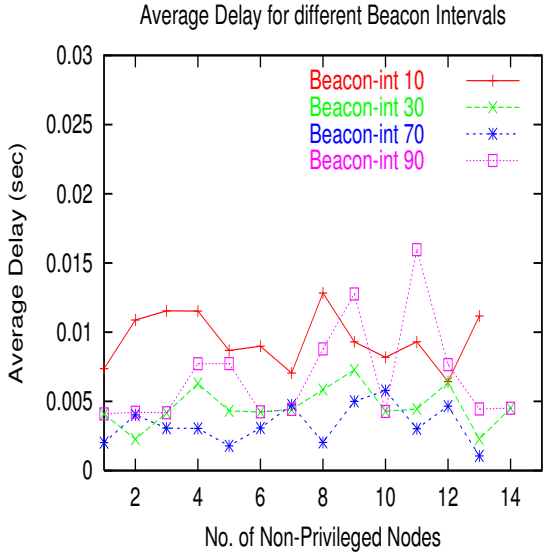


Figure 8.8: Average Delay for Altering Beacon Intervals

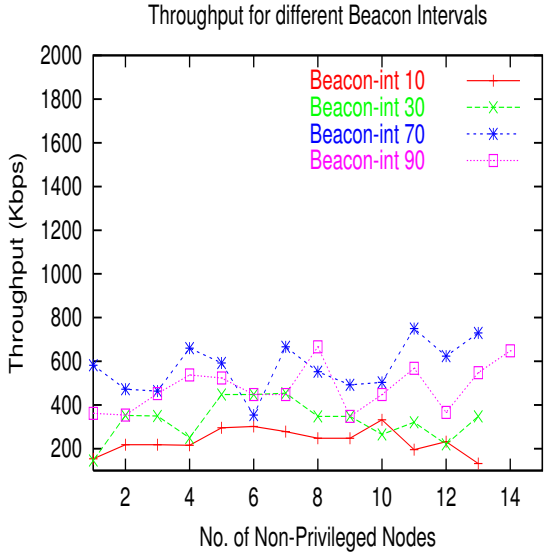


Figure 8.9: Throughput for Altering Beacon Intervals

become less. As it can be inferred that as the number of nodes increase, the contention for the channel access increases and hence the performance is not maintained equally. Thus as an optimum, we can claim that for the given scenario, the optimal slot-size is 55μsecs.

For more graphs showing various slot sizes and their corresponding delays and throughputs, please refer to the Appendix.

8.2.3 Optimal Beacon Interval

As for the slot-size, we also need to find the optimal beacon interval for the given scenario. We did rigorous simulations for the same and we now claim that the ideal beacon interval for the simulation scenario we considered is 70ms. As we can see in the figures 8.8 and 8.9, the lowest beacon interval in the graph has a bit lower average delay than the rest, but the throughput suffers a lot. This is so as, because the interval is less, the transmission time in the CP becomes very less. So because of DTMA, the nodes transmit guaranteedly only in the CFP. Due to very less time in the CP, many nodes can starve and hence the lower throughput.

We tested our protocol DTMA, using some more parameters. Refer the Appendix for more detailed graphs.

Chapter 9

Conclusion and Future Work

9.1 Conclusion

With the advent of real time applications like video conferencing, voice applications on the wireless networks, there arises the need for guarantees of finite delay and dedicated bandwidth. The user wants this QoS to be transparent, even if he / she is mobile. The current schemes do provide some quality of service, but still better utilization of the channel can be done and better QoS guarantees can be given to the end user. Our scheme DTMA claims to avoid the overheads of explicit control messages and thus ensures more time for the actual data transfer and hence less delays for the previleged nodes that have been assured some QoS guarantees.

Also we lay foundation for a scheme, wherein the APs in the WLAN can communicate and share their data and control information and based on this information, the QoS guarantees will be provided even when the nodes move between APs. The APs could also 'suggest' the nodes to associate with an AP that is less loaded, hence increasing the quality of service the node gets.

Using theoretical models and thorough simulations, we intend to show that our scheme does perform better than the ones that are currently existent and thus provides user with finite delay guarantees and better QoS, while achieving the optimal throughput for the given scenario.

9.2 Future Research

DTMA tries to provide bounded delays to nodes associated with a particular Access Point. But it could happen that an AP might run out of resources and might not be able to serve any more nodes. The situation is even worse if an adjacent AP is very lightly loaded and can serve the node. This situation could be exploited if the APs communicate with each other using hello messages. So a lightly loaded neighbouring AP could balance the load of its 'loaded neighbour'. Based on the information communicated between APs, if the other AP is in the range of the node, the current AP could explicitly inform the node to change to the frequency to the operating frequency of the lightly loaded AP.

If the coverage areas are not overlapping, the 'loaded' AP could inform the node trying to get associated to move in the vicinity of an AP that is lightly loaded and hence the node has a chance of getting better service. The Inter Access Point Protocol (IAPP) is not much exploited. The APs can communicate the service requirements and hence provide transparent QoS based services even when the nodes are mobile. These possibilties of using the collective information of the APs in a domain to provide overall better QoS guarantees are open for research.

The proposed scheme **DTMA** requires the slot size to be optimal. It might change depending on the given network conditions. We manually changed the values and conducted experiments for our work. Finding this slot-size dynamically using the information available at the AP is still an open problem.

Literature specifies and we too observed that EDCF and HCF do not perform equally well in all conditions. Some parameters like the network load, maximum delay suffered, throughput could be used to determine a threshold to switch from one scheme to another. Determining these exact parameters and finding an aggregate function to determine the switching point can also be an interesting problem to solve.

Appendix A

Additional Results

This appendix provides the graphs depicting the results of some of the additional experiments we conducted to observe the behaviour of DTMA in various interesting scenarios. The scenarios considered are similar to the ones described in chapter 8. The graphs comparing the performance of DTMA against some of the earlier works have been presented in chapter 8 itself.

The graphs presented here are divided into three parts :

1. Slot Size Determination
2. Varying Packet Size
3. Varying Data Rates

The graphs in the slot-size determination section are the ones that we used for determining the optimal slot-size for the kind of network scenario we were considering. As quoted before, the optimal slot-size was found to be 55μsecs. The performance in terms of the delay suffered and the throughput for different slot sizes that were candidates for our experiments are shown in this section. These graphs justify the selection of 55μsecs as the best slot-size as it outperforms the other candidates.

The second set of graphs depict the effect on the performance, that is, the throughput and average delay due the variation in the size of data packets transmitted. The graphs show the usual trend that as the packet size increases beyond 1500 bytes, there is an increase in the

delay suffered due to the fragmentation. For the same reason, the throughput also degrades. These are the results that are expected and proved to be the starting results that showed that our scheme was implemented properly and conforms to the conventional trends of data transmission.

The last set of graphs show the performance got when the input data rate was varied for different number of privileged nodes.

Thus, the results in this appendix are for the more curious reader who might want to study the behaviour of DTMA under the scenarios of varying packet-size, data rate, number of privileged nodes and slot-size for the TDMA portion.

A.1 Packet Size Varying

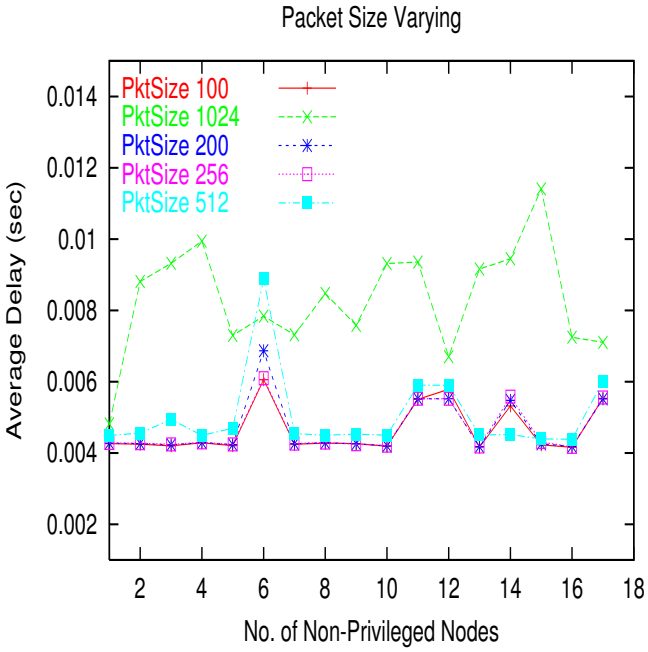


Figure A.1: Varying Packet Size 1

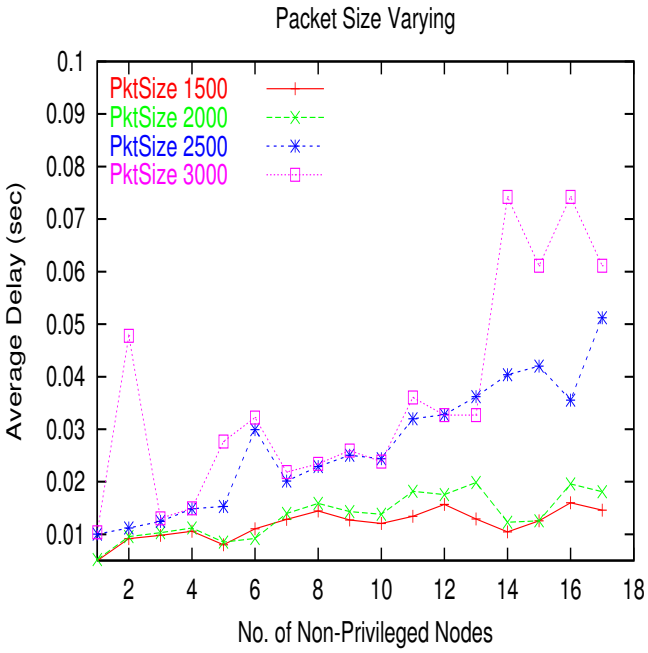


Figure A.2: Varying Packet Size 2

A.2 Slot Size Determination

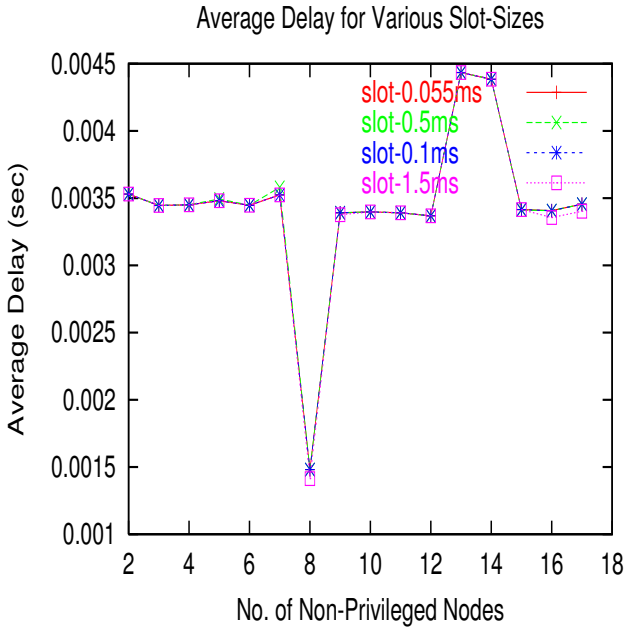


Figure A.3: Slot Size Determination : Average Delay for 4 Priority Nodes

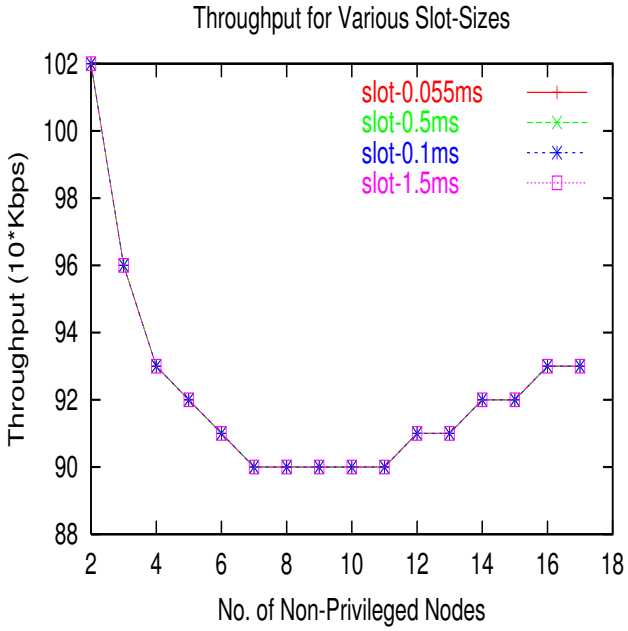


Figure A.4: Slot Size Determination : Throughput for 4 Priority Nodes

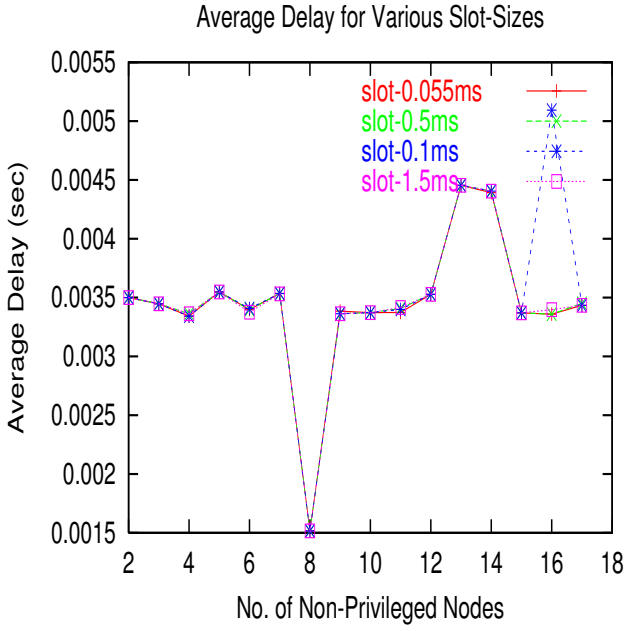


Figure A.5: Slot Size Determination : Average Delay for 5 Priority Nodes

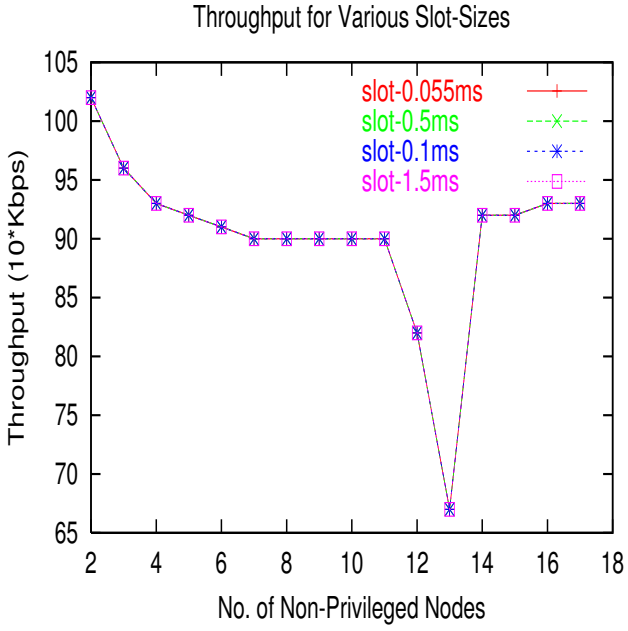


Figure A.6: Slot Size Determination : Throughput for 5 Priority Nodes

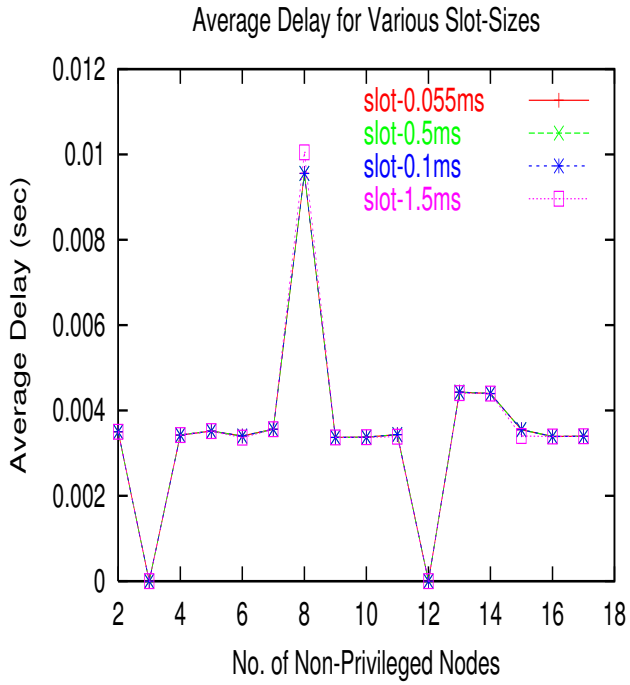


Figure A.7: Slot Size Determination : Average Delay for 6 Priority Nodes

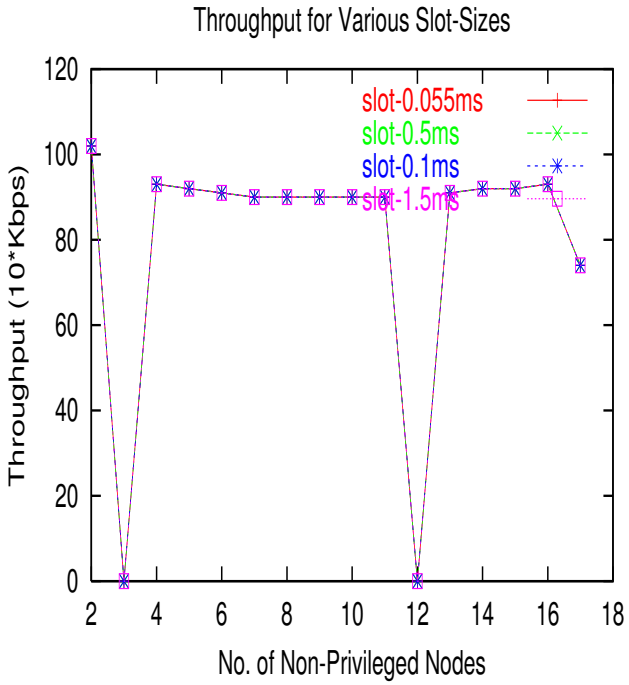


Figure A.8: Slot Size Determination : Throughput for 6 Priority Nodes

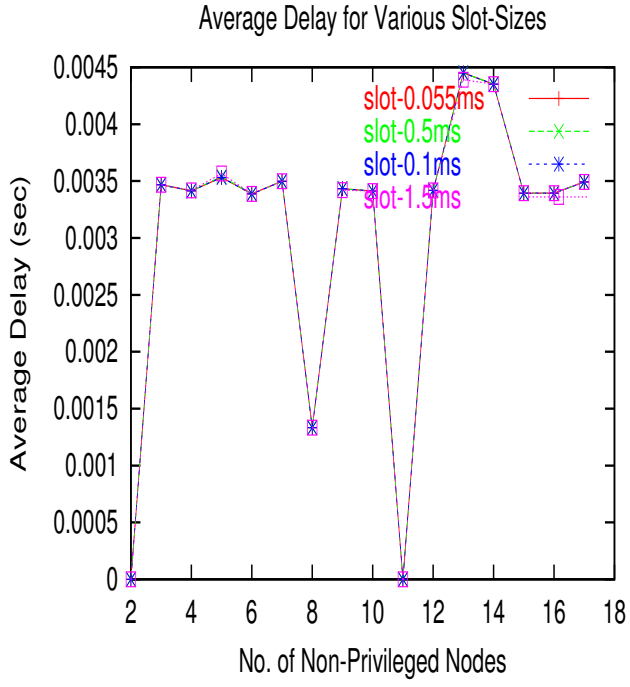


Figure A.9: Slot Size Determination : Average Delay for 8 Priority Nodes

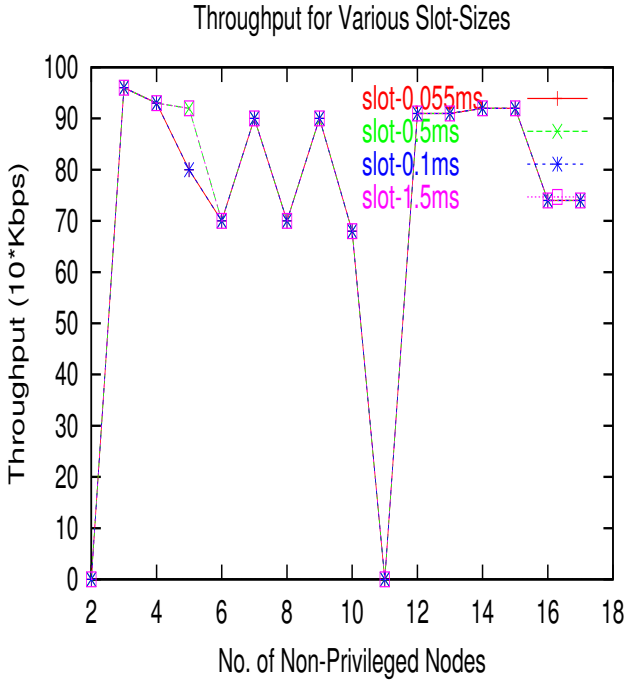


Figure A.10: Slot Size Determination : Throughput for 8 Priority Nodes

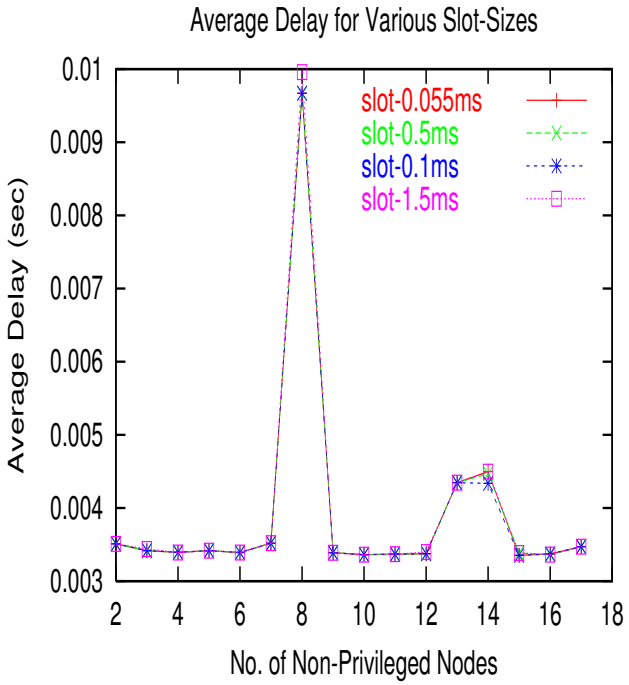


Figure A.11: Slot Size Determination : Average Delay for 9 Priority Nodes

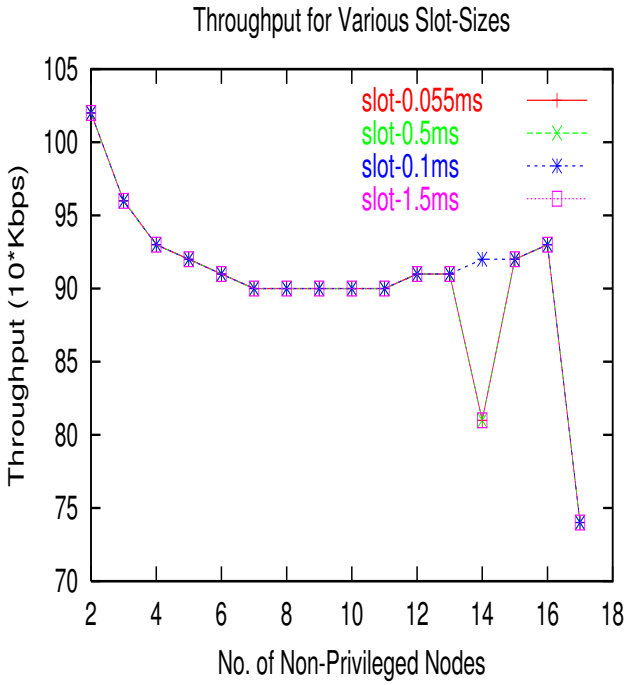


Figure A.12: Slot Size Determination : Throughput for 9 Priority Nodes

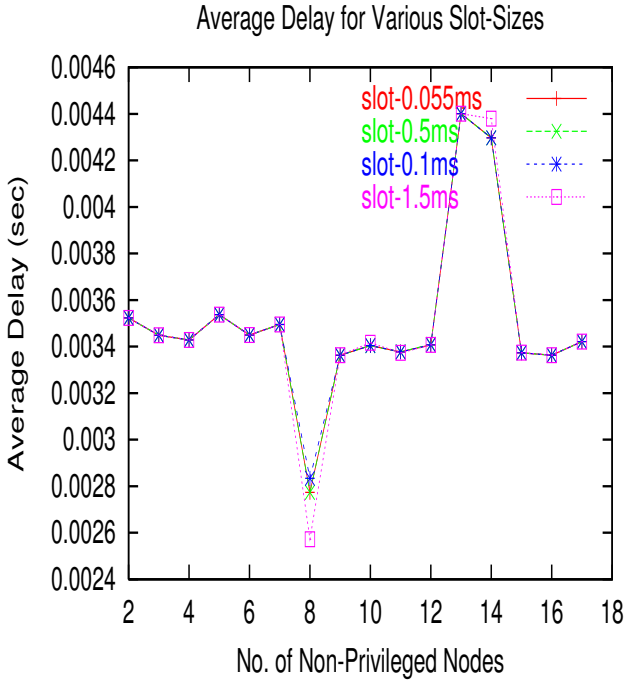


Figure A.13: Slot Size Determination : Average Delay for 2 Priority Nodes

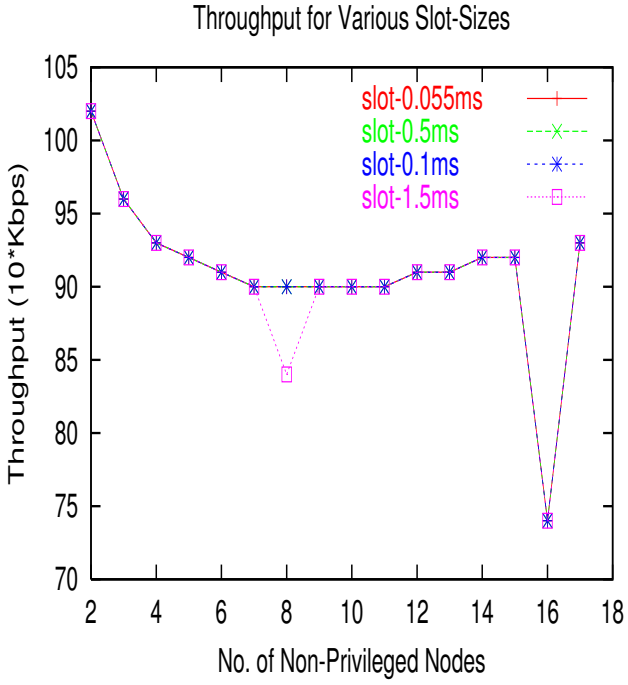


Figure A.14: Slot Size Determination : Throughput for 2 Priority Nodes

A.3 Varying Data Rates

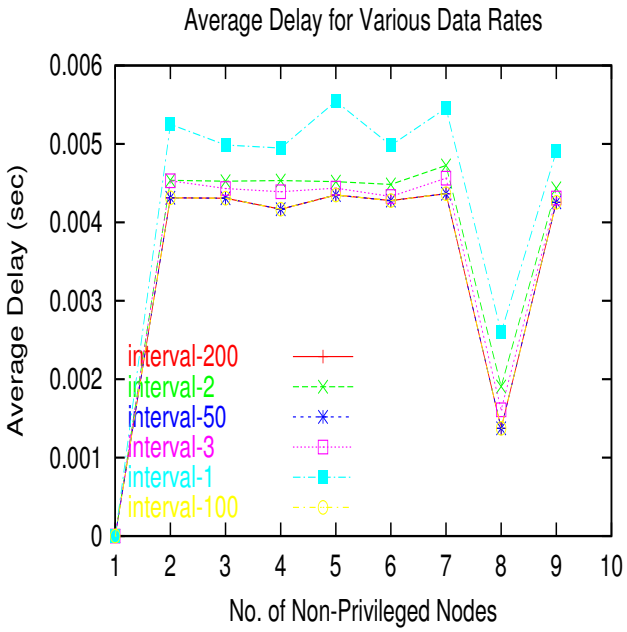


Figure A.15: Interval Varying : Average Delay for 1 Priority Node

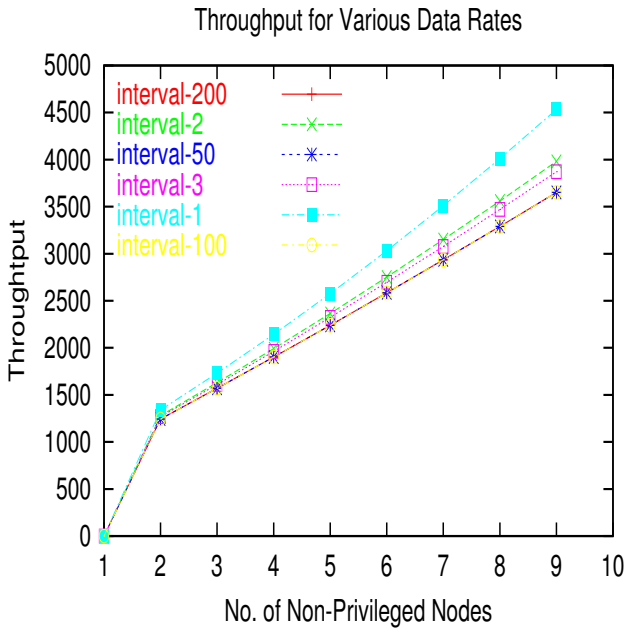


Figure A.16: Interval Varying : Throughput for 1 Priority Nodes

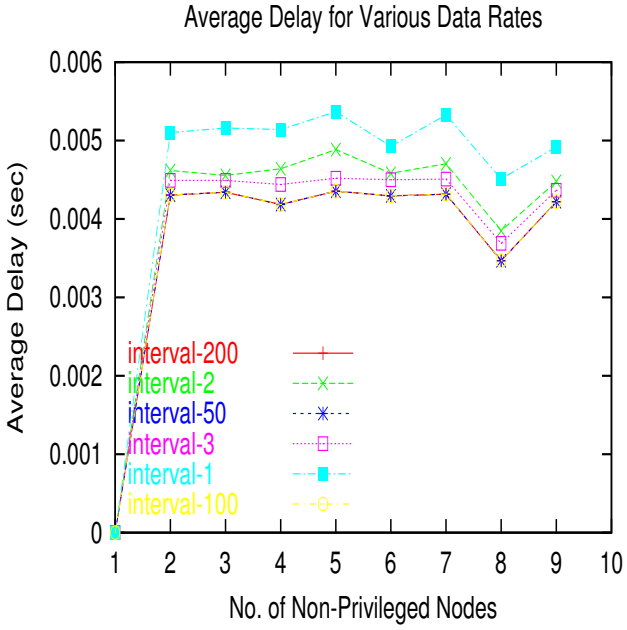


Figure A.17: Interval Varying : Average Delay for 2 Priority Nodes

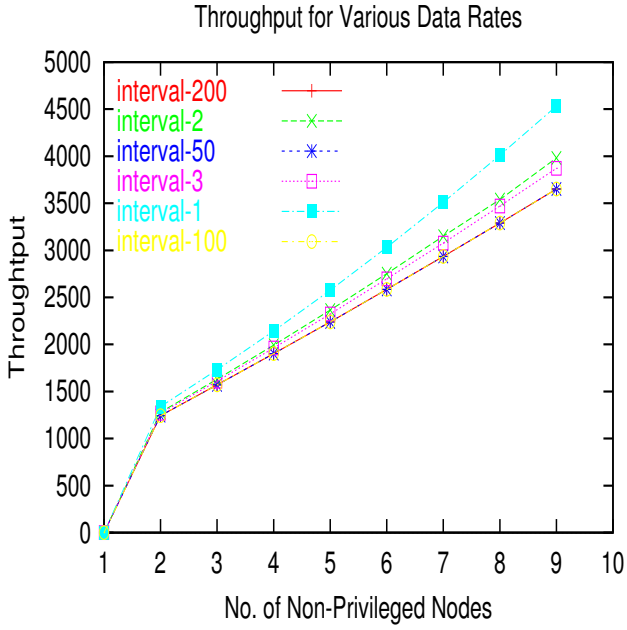


Figure A.18: Interval Varying : Throughput for 2 Privileged Nodes

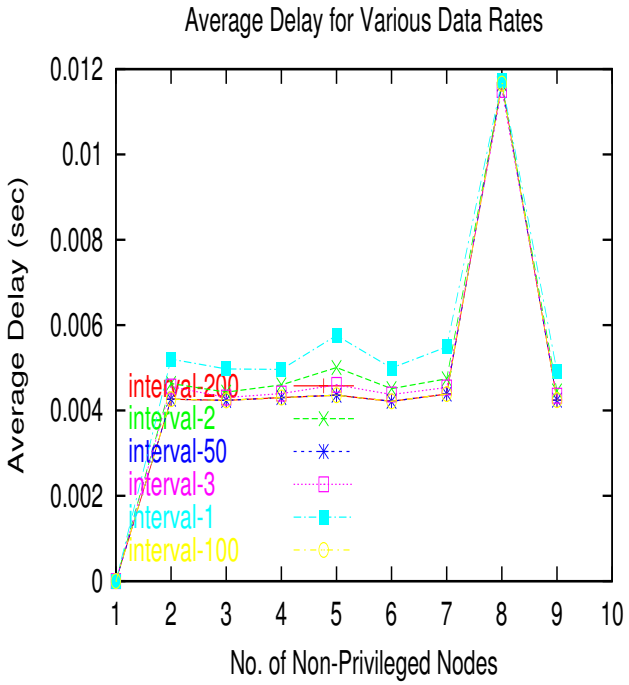


Figure A.19: Interval Varying Graph : Average Delay for 3 Priority Nodes

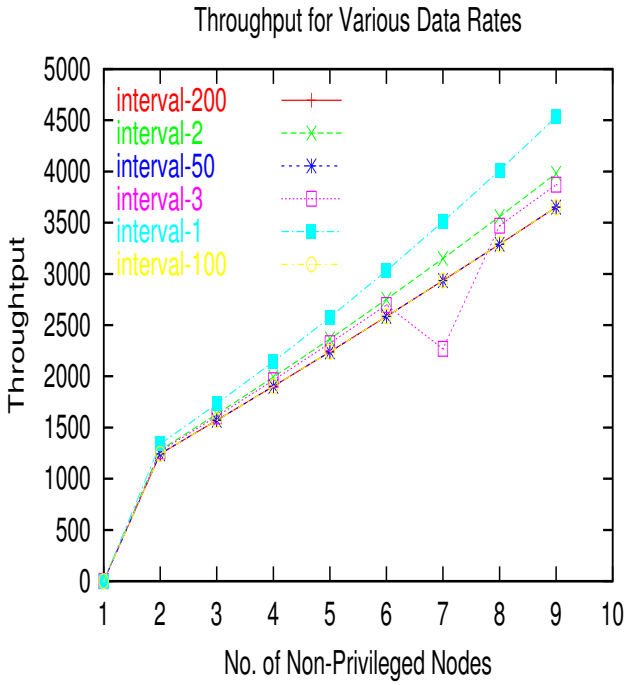


Figure A.20: Interval Varying Graphs : Throughput for 3 Privileged Nodes

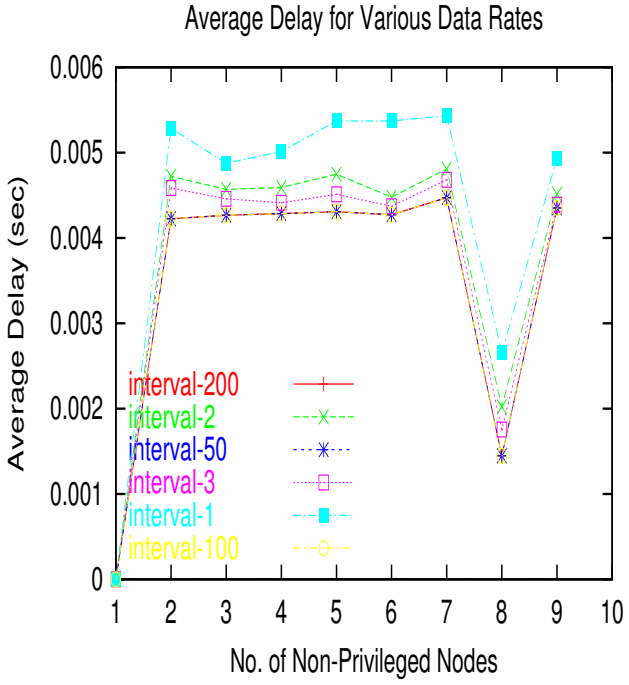


Figure A.21: Interval Varying Graph : Average Delay for 4 Priority Node

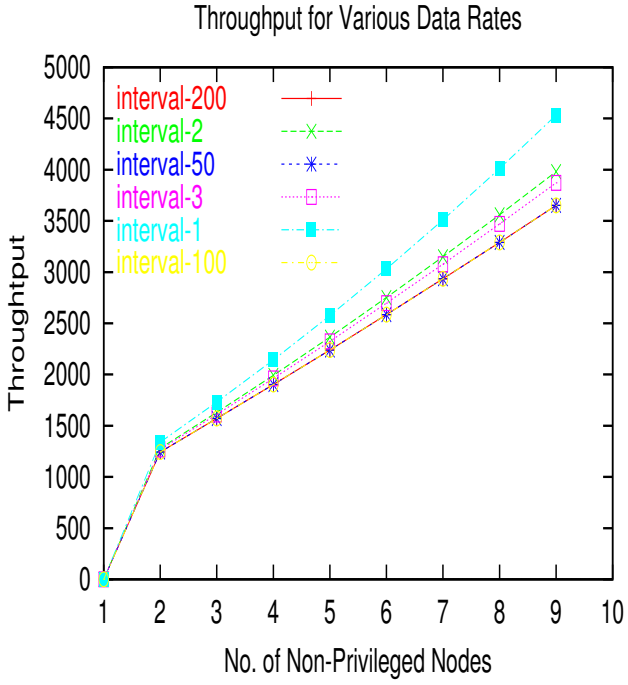


Figure A.22: Interval Varying Graphs : Throughput for 4 Privileged Nodes

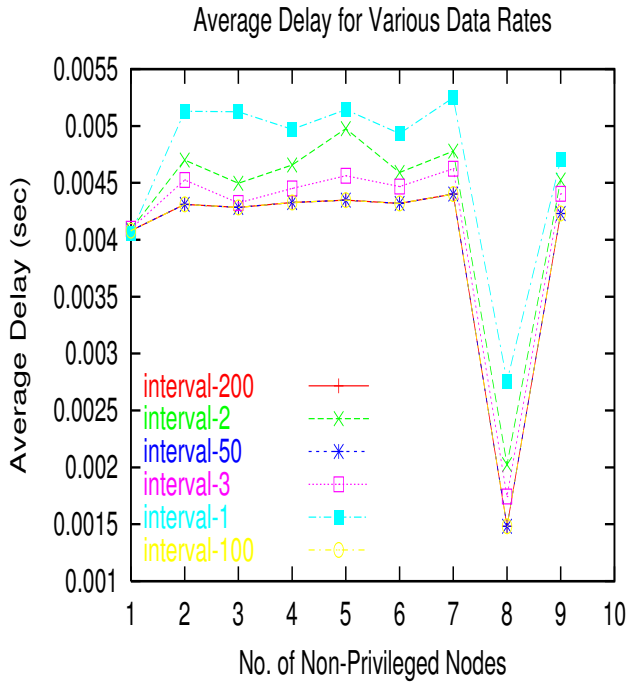


Figure A.23: Interval Varying Graph : Average Delay for 5 Priority Nodes

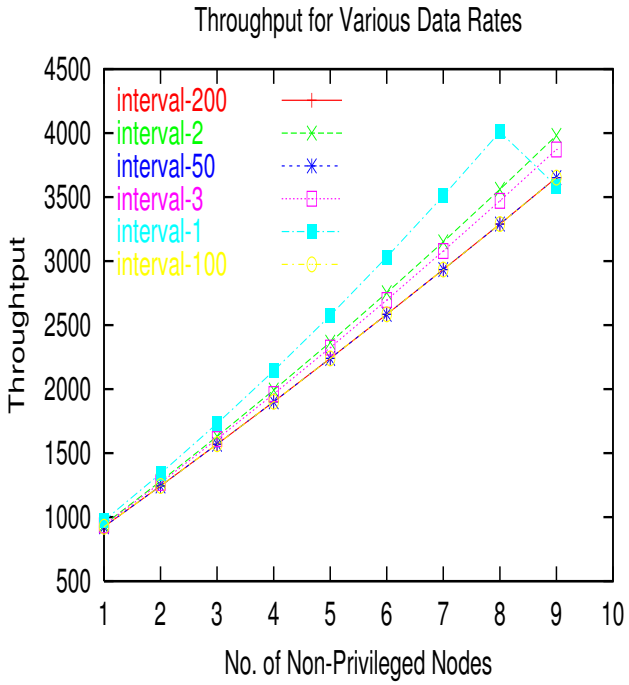


Figure A.24: Interval Varying Graphs : Throughput for 5 Privileged Nodes

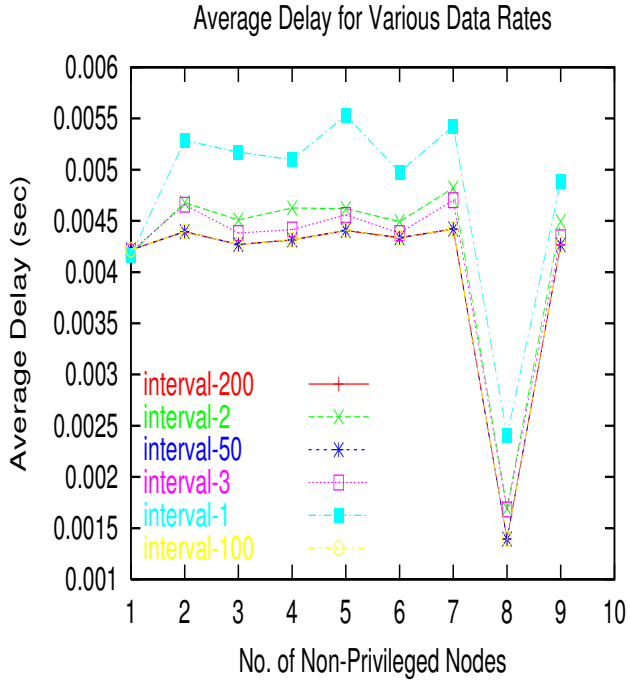


Figure A.25: Interval Varying Graph : Average Delay for 6 Priority Nodes

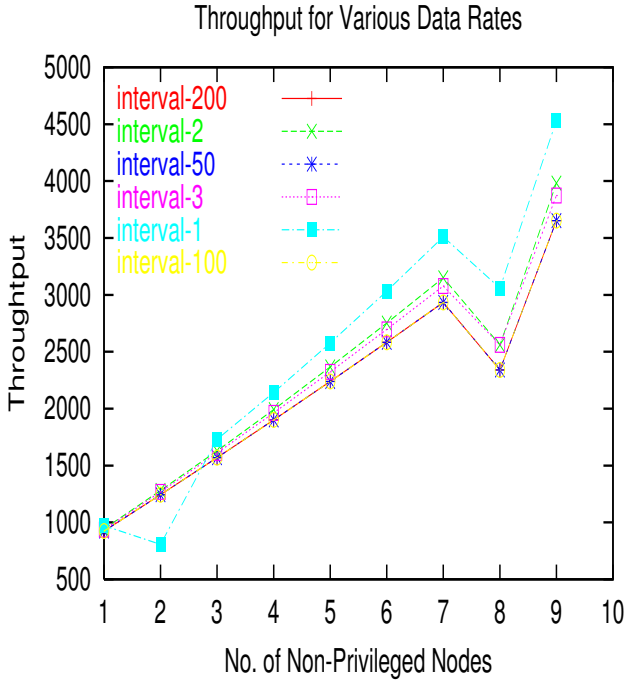


Figure A.26: Interval Varying Graphs : Throughput for 6 Privileged Nodes

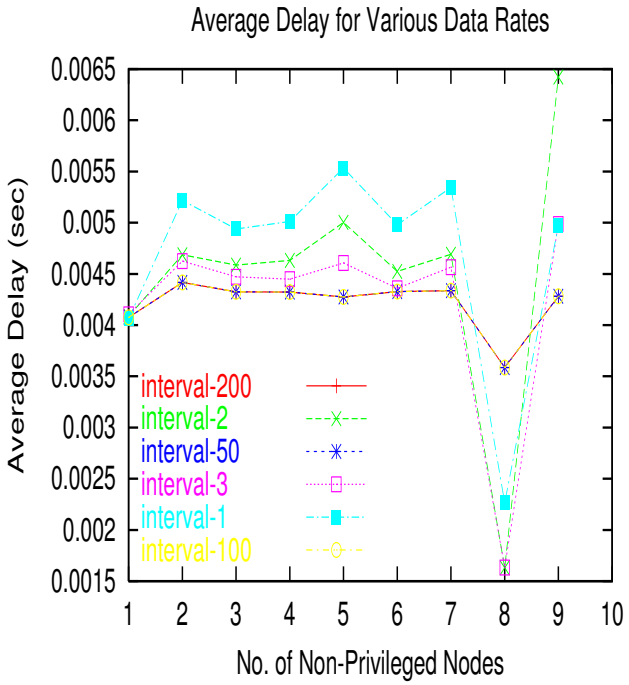


Figure A.27: Interval Varying Graph : Average Delay for 7 Priority Nodes

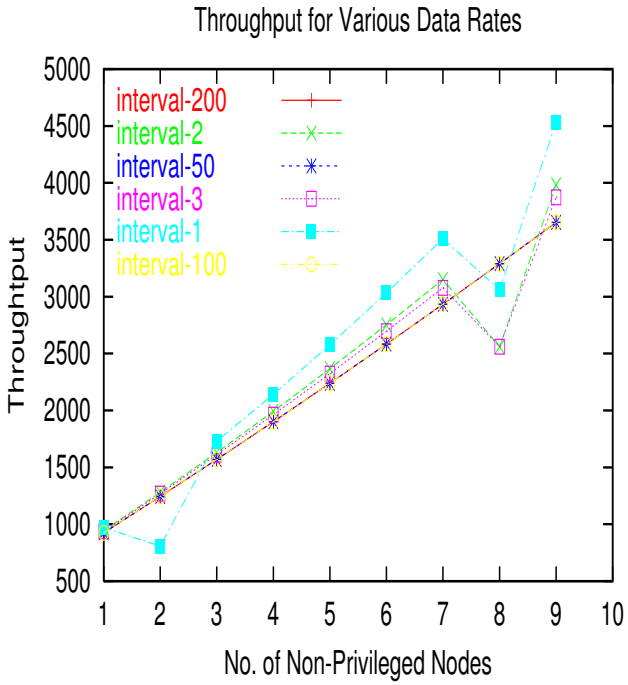


Figure A.28: Interval Varying Graphs : Throughput for 7 Privileged Nodes

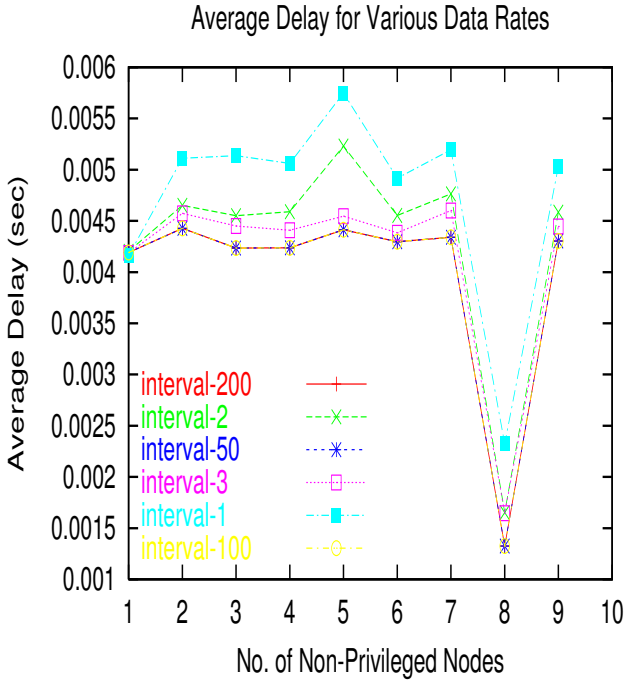


Figure A.29: Interval Varying Graph : Average Delay for 8 Priority Nodes

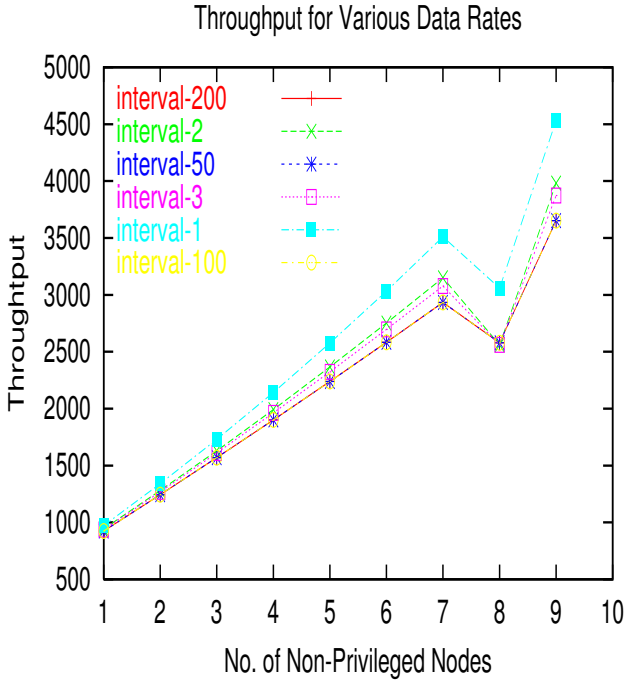


Figure A.30: Interval Varying Graphs : Throughput for 8 Privileged Nodes

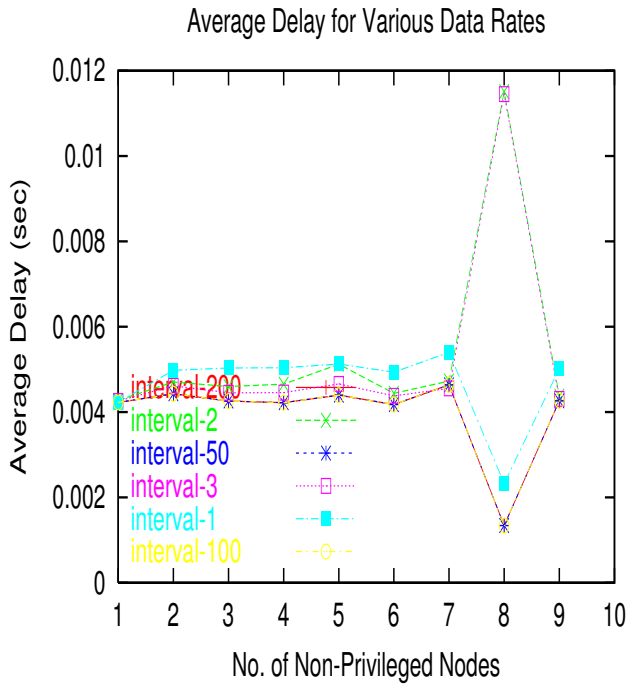


Figure A.31: Interval Varying Graph : Average Delay for 9 Priority Nodes

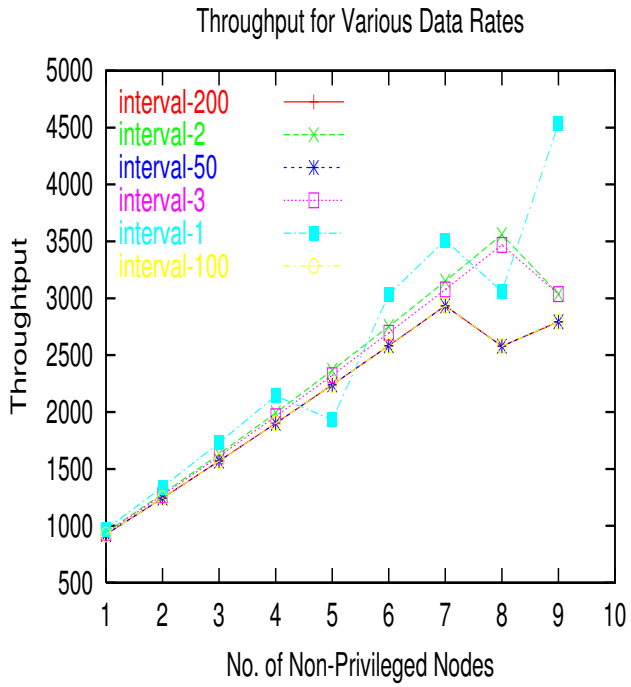


Figure A.32: Interval Varying Graphs : Throughput for 9 Privileged Nodes

Bibliography

[1] Abhishek Goliya, "Dynamic Adaptation of PCF and DCF mode of 802.11 WLAN" , M.Tech Dissertation, 2003.

[2] Anand Balachandran, Paramvir Bahl, Geoffrey M. Voelker, "Hot-Spot Congestion Relief in Public-area Wireless Networks", *Proceedings of teh Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '02)*, February 2002.

[3] Andereas Kopsel, Jean-Pierre Ebert, Adam Wolisz, "A Performance Comparison of Point and Distributed Coordination Function of an IEEE 802.11 WLAN in the presence of Real-Time Requirements", *Proceedings of 7th Intl. Workshop on Mobile Communications*, October 2000.

[4] Antonio Grilo, Mario Nunes, "Performance Evaluation of IEEE 802.11E" *Proceedings of PIMRC02*, Lisbon 2002.

[5] Aravind Velayutham, J. Morris Chang "An Enhanced Alternative to IEEE 802.11e MAC Scheme", IOWA State University, Ames-50011, IA. 2003.

[6] Duke Lee, Roberto Attias, Anuj Puri, "A Wireless Token Ring Protocol for Intelligent Transportation Systems", *IEEE Intelligent Transportation Systems Proceedings*, Oakland Marriot City Center Hotel, Oakland, California, USA. August 25-29, 2001.

[7] Hua Zhu and Imrich Chlamtac, "An Analytical Model for IEEE 802.11e EDCF Differential Services", *Proceedings of ICC 2003* , 2003.

[8] IEEE 802.11 WG. *IEEE Std 802.11, 1999 Edition International Standard [for] Information Technology-Telecommunications and Information Exchange between systems-Local and metropolitan networks.* IEEE,1999 .

[9] IEEE 802.11 WG.*Draft Supplement to International Standard [for] Information Technology-Telecommunications and Information Exchange between systems LAN/MAN Specific Requirements.* IEEE 802.11e/D2.0, Nov 2001.

[10] Information and Source Code,*Network Simulator 2.* http://www.isi.edu/nsnam/ns. 2002.

[11] James Crawford, "IEEE 802.11E/A Throughput Analysis", *Magis Document No. E10282*, 2003 .

[12] Li Zheng, Arek Dadej, Steven Gordon, "Hybrid Quality of Service Architectur for Wireless / Mobile Environment" *IFIP Interworking 2002 Convereged Network : Real Time Over IP Conference*, 2002 .

[13] Moustafa Youseff, Arunchander Vasan, Raymond Miller, "Specification and Analysis of the DCF and PCF Protocols in the 802.11 Standard",Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP 2002), Paris, France, November 2002.

[14] Priyank Garg, Rushabh Doshi, Russell Greene, Mary Baker, Majid Malek, Xiaoyan Cheng "Using IEEE 802.11e MAC for QoS over Wireless", *The Proceedings of the 22nd International Performance Computing and Communications Conference (IPCCC 2003)*, Phoenix, Arizona, April 2003.

[15] Qiang Ni, Lamia Romdhani and Thierry Turletti, "AEDCF : Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks", *RINRIA*, IEEE 2003.

[16] Qiang Ni, Lamia Romdhani, Thierry Turletti and Imad Aad, "QoS Issues and Enhancements for IEEE 802.11 Wireless LAN", *RINRIA*, November 2002.

[17] Q. Qiang, L. Jacob, R. R. Pillai and B. Prabhakaran, "MAC Protocol Enhancements for QoS Guarantees and Fairness over IEEE 802.11 Wireless LANs ", *Proceedings of International Conference on Computer Communications and Networks*, October 2002.

[18] Ravindra S. Ranasinghe, Lachlan L.H. Andrew, David Everitt, "Distributed Contention Free Traffic Scheduling in IEEE 802.11 Multimedia Networks", *10th IEEE Workshop on Local and Metropolitan Area Networks*, pp 18 -28, 2001.

[19] RFC 1889, *RTP : A Transport Protocol for Real-Time Applications,* S. Casner, V. Jacobson, H. Schulzrinne, R. Frederick. January 1996.

[20] RFC 2543, *SIP : Session Initiation Protocol,* M.Handley, E.Schooler, H. Schulzrinne, J. Rosenberg. March 1999.

[21] Sobrinho JL and Krishnakumar AS, "Real Time Traffic Over IEEE 802.11 MAC Layer", *Bell Lab Technical Journal*, pages 172-187, Autumn 1996.

[22] Stefan Mangold, Sunghyan Choi, Peter May, Guido Hiertz, "IEEE 802.11E Fair Resource Sharing betwen Overlapping Basic Service Sets", *Proceeding of PIMRC02* Lisbon, 2002 .

[23] S. Mangold, S. Chio, P. May, O. Klien, G. Hiertz and L. Stibor, "IEEE 802.11e Wireless LAN for Quality of Service", *Proceedings of European Wireless*, February 2002.

[24] Y. Chen, Qing-An Zeng and Dharma P. Agarwal,"Performance analysis and enhancement for IEEE 802.11 MAC protocol", *Proceedings of International Conference on Telecommunications*, February 2003.

[25] Yunli Chen, Qing-An Zeng and Dharma P. Agarwal, "Performance analysis of IEEE 802.11e Enhanced Distributed Co-ordination Function", *The 11th IEEE International Conference on Networks*, October 2003.

[26] Zheng Wang, "Internet QoS : Architectures and Mechanisms for Quality of Service" *San Francisco:Morgan Kauffmann Publishers*, 2001.