DESIGN AND EVALUATION OF A WINDOW-BASED WIRELESS MEDIUM ACCESS CONTROL PROTOCOL

BY

XIAO SU

B.E., Zhejiang University, 1994

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1997

Urbana, Illinois

Abstract

The proliferation of wireless devices calls for a suitable network infrastructure for supporting communication among them. Medium Access Control(MAC), the most crucial part of the protocol stack, is discussed in the thesis. The design of wireless MAC protocols is significantly different from wired MAC protocols because of distinctive features of wireless medium, such as high error rates, limited radio propagation range and low bandwidth.

An efficient, scalable and fair protocol – Wireless Window Protocol(WWP) – is proposed. WWP is a limited contention protocol in which the set of contending mobiles is chosen based on a global contention window maintained by every mobile station. The contention window is adjusted based on three possible channel states: no transmission, success and collision. Window bounds are functions of current channel load. Simulations have shown that

- WWP can resolve contention in an average of 2.4 contention slots, independent of the number of contending stations.

- The access delays for each station under heavy load is exponentially distributed, both in a single-cell and multi-cell scenarios.

We show that the performance of WWP is superior as compared to the distributed coordination function of IEEE 802.11 draft standard.

To my parents, my sister and Tao

Acknowledgments

I would like to express my sincere gratitude to my advisor, Prof. Benjamin W. Wah, for his guidance, encouragement and valuable insights in this work. He introduced me to the area and inspired me along the way.

I would like to thank all the members in our research group for providing crucial comments on the work and for providing a congenial environment for me to work in.

Special thanks go to Tao for his love and care which makes me feel at home in the States. I am also indebted to my parents for all their support, motivation and encouragement during my academic career.

Finally, I would like to acknowledge the support of National Science Foundation Grant MIP 96-32316 without which this work would not have been possible.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivations

Advances in VLSI technology have enabled the proliferation of wireless devices, such as palmtops, personal digital assistants(PDAs) and portable computers. While nowadays these are self-contained devices, they will eventually be part of a larger network infrastructure. Although the combination of wired and wireless networks gives greater flexibility to computer users, it also introduces a lot of technological challenges.

First, wireless networks are less reliable than wired networks because channels are "exposed" to the surrounding environment that is susceptible to interference and noise. Further, radio communication introduces additional echoes and interferences. Methods to provide reliable transmission services over inherent noisy channels and suitable error control mechanisms must be developed before such networks can be used in wide practice.

Second, wireless devices may require the capability of constant network access while in motion. How to provide seamless roaming so that users can still be granted the same level of quality of service, regardless of location changes?

Third, wireless devices operate on finite energy sources. How to reduce power consumption so that devices can function for longer time with small batteries?

Last, wireless networks have lower bandwidth as compared to wired networks, e.g., current products can achieve only 1Mbps for infrared communication, 2Mbps for radio communication and 9-14kbps for cellular telephony [13] while Ethernet supports from 10Mbps to 100 Mbps, and ATM provides over 100Mbps. How to efficiently make use of the limited bandwidth is an important issue.

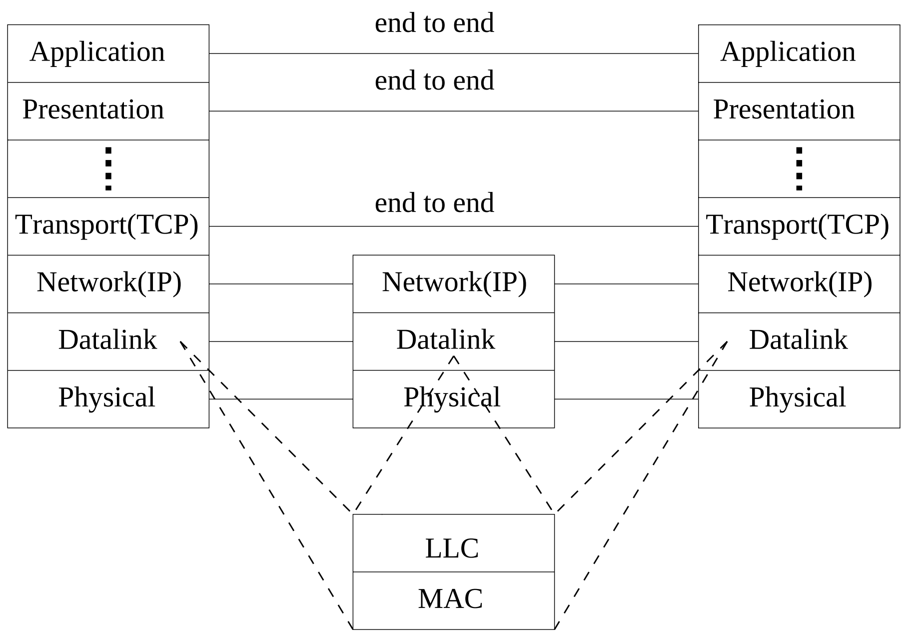

Figure 1.1: Network Protocol Layers

The current TCP/IP network protocol stack shown in Figure 1.1 works well on wired networks but needs extensive modifications in order to adapt to a wireless environment and address the issues raised above. At the application level, efficient computation and reduction in periodic monitoring operations can help reduce power consumption, while advanced compression algorithms can be applied at the presentation layer to efficiently utilize the relatively small bandwidth. TCP is a reliable transport protocol tuned to perform well in wired networks for which congestion is the major cause of packet losses. However, in wireless networks, frequent handoffs, occasional disconnections and high bit error rates may cause significantly more packet losses. Revisions of current TCP protocols are necessary in face of new problems over wireless links. Some of the modifications have been proposed in [1, 2, 10]. Current IP is also not suitable to be used in wireless networks since IP addresses reflect physical network connections, which implies that if a station moves to another place it must acquire a new IP address and reconfigure its network device. Research on mobile-IP has been done extensively [6, 12, 14] with the aim of providing seamless mobility to users. *Medium Access Control*(MAC) regulates the access to a shared channel by granting each node a chance to transmit its packets. Current *carrier sense multiaccess with collision detection*(CSMA/CD) protocols on Ethernets rely on the ease of carrier sensing by measuring current from the network. Carrier sense is not ideal for a wireless environment because of the hidden terminal scenario [7, 11]. Improvements over the CSMA approach have been proposed in the literature [17, 21, 23].

Therefore, it is a real undertaking to redesign a wireless network infrastructure. In this thesis we concentrate on the design of the wireless MAC layer since the channel is a precious resource for the wireless network, and efficient and fair allocation of the channel is considered to be critical in its overall design.

1.2 Problem Statement

To design a good MAC protocol, the first question that should be asked is what expected features it should provide. The IEEE 802.11 study group has already addressed this problem by stating twenty requirements of a decent MAC protocol.

- **Throughput** – The channel is a scarce resource, so throughput is the foremost design consideration. Reducing contention time is one of the possible ways to increase throughput. Wireless Window Protocol(WWP) that this thesis proposes applies a dynamic programming formulation to minimize contention time.

- **Delay** – Short response time and small variances in response times are important for synchronous services such as real time audio and video transmission.

- **Transparency to Physical Layers** – IEEE 802.11 draft standard has proposed three different physical layers, namely, direct sequence spread spectrum, frequency-hopped spread spectrum and diffuse infrared. One single MAC protocol should be able to run on top of these three physical carriers.

- **Ability to Serve Data, Voice and Video** – Wired networks are approaching the age of integrated asynchronous and synchronous services. Protocols such as *Integrated Services Digital Networks*(ISDN) and *Broadband Integrated Services Digital Networks*(B-ISDN) have been proposed. These services should be designed in future wireless networks.

- **Fairness of Access** – The fading characteristics of radio channels may cause unequal receiving power at a base station even when power control is applied. This results in unfair access to a channel. A MAC protocol should not favor a paricular station because of its location.

- **Battery Power Consumption** – Wireless devices rely on limited battery power, so power conservation should be an important consideration for a MAC protocol. Those requiring constant monitoring of a channel are not efficient since more power will be consumed.

- **Scalability** – Wireless LANs may need to support various number of nodes in different settings, ranging from a few to hundreds. A MAC protocol should scale well to deliver satisfactory performance for a large number of stations.

- **Robustness with Cell Overlays** – Co-located networks impose two challenges for MAC design: users from an adjacent LAN may break into the system and cause security problems; and interference from a nearby LAN can result in unfair channel allocation since it is difficult for stations on the boundary to cleanly receive contention signals, reducing their chances to contend for the channel.

- **Handoff Support** – To support high level mobility, a MAC protocol has to provide a handoff function for nodes migrating from one service area to another.

- **Support of Ad Hoc Networking** – To support ad hoc networking, a MAC protocol should not be based on *a priori* information on network topology.

- **Security** – Unauthorized accesses may have negative impact on system throughput, delay and security. A decent MAC protocol should reject such kind of accesses and minimize their effects.
- **Broadcast(Multicast) Support** – While broadcasts are natural in wireless communication, multicasts are not. A good MAC protocol should provide this functionality in order to support applications like video conferencing.
- **Critical Delay** – A MAC protocol requiring synchronization among stations can only function well in a relatively small coverage area, since large differences in propagation delay may degrade the performance of the protocol.
- **Minimization of Capture Effects** – Capture refers to the phenomenon that while two stations are transmitting at the same time, the receiver is still able to correctly receive from one station due to a large difference in received power from simultaneous transmissions. Although capture increases throughput it causes unfairness. In general, capture effects should be minimized.
- **Priority Support** – Stations contending for the channel have different priorities in some applications; hence, priority traffic should be supported.
- **Support for Asymmetric Uplink and Downlink** – Downlink traffic is usually larger than uplink traffic, and a good MAC should support this feature.
- **Packet Ordering** – Correct ordering of incoming packets is crucial for synchronous applications. It can be supported by alternating bit protocol, go-back-n or selective repeat window protocol.
- **Compatibility** – Different types of wireless LANs may need to function together, and compatibility is an important concern.
- **Simpilicity of Physical Layer** – MAC design that relegates difficult tasks to the physical layer is not desirable.

- **Simplicity of Protocol** – Simplicity of a MAC protocol helps produce high quality products in time for the market.

Some of the characteristics may not be compatible with others, and certain properties are achieved at the expense of others in some designs. Take CSMA/CD as an example. Peak throughput brings about intolerable delay which means that the theoretical upper bound for throughput is hard to achieve in practice. Therefore, we need to focus on a subset of features that are considered to be the most important. Our goal is to design a MAC protocol that renders high throughput, good delay characterisctics, fairness in access and scalability.

1.3 Thesis Contributions

The thesis proposes a new wireless MAC protocol – WWP. It is distinct from other protocols in that it minimizes contention time by a dynamic programming formulation. Simulations have shown that it has a load-independent behavior and needs an average of 2.4 cycles to resolve contention regardless of network load. Similar work has been done by Wah and Juang [3, 4, 5] for wired local area networks. The thesis presents an efficient generation of lookup tables to be used in WWP and modifies its wired counterpart in order to adapt to noisy channels, the overlapping of cells and longer carrier state detection time as compared to that of wired local area networks. The thesis also compares the performance of WWP and the IEEE 802.11 draft standard protocol.

1.4 Organization of the Thesis

This thesis is organized in five chapters. This chapter motivates the design of an efficient, fair and scalable wireless LAN MAC protocol and lists twenty desirable features of an ideal MAC protocol. Since almost every wireless LAN MAC protocol proposed in the literature has its origin in wired MAC protocols, Chapter 2 descibes popular MAC protocols for conventional networks. Chapter 3 addresses some issues in wireless MAC design and analyzes some proposed protocols. Chapter 4 studies several design alternatives of WWP in different scenarios. We evaluate performance of WWP by simulations in Chapter 5 and also compare WWP with the *distribution coordination function*(DCF) of the 802.11 draft standard.

Chapter 2

Wired Medium Access Overview

2.1 Aloha/Slotted Aloha

Aloha is the first type of *multiple access protocols* that allow stations to transmit at random times. It is more flexible and robust than fixed allocation schemes like *time division multiple access*(TDMA) and *frequency division multiple access*(FDMA). Ethernet's CSMA/CD was derived from a variant of Aloha protocol.

There are two types of Aloha protocol: pure and slotted. In pure Aloha, stations can transmit at arbitrary times. In slotted Aloha, time is divided into slots that are equal to the packet transmission time. Ready stations can only transmit at the beginning of each slot. In both versions, stations retransmit packets that collide with others after a random delay.

The maximum efficiency of pure Aloha is 18% and that of slotted Aloha is 36%, where efficiency is defined as proportion of time that is used for successful packet transmission in heavy load conditions. The efficiency of the two Aloha protocols is rather poor. One way to improve it is to use *reservations*. R.Aloha adopts this scheme and can achieve 88% efficiency if the reservation slot is 5% of a packet transmission [15].

2.2 CSMA/CD

Carrier sense multi-access with collision detection(CSMA/CD) is a polite version of Aloha protocol in that it senses the channel before transmission and stops transmission once collision is detected [15]. To transmit a packet, a station executes the following algorithm:

1. Wait until the channel is idle;
2. Transmit and listen to the channel at the same time;
3. If collision is detected, stop transmission, jam the channel with short signals and set the backoff counter.

The backoff counter is set by a rule called *exponential backoff*: after a packet has been collided for n times, the station waits for $K \times 512$ bit times, where K is chosen from the set $\{0, 1, 2, \ldots, 2^{p-1}\}$ and $p = min\{10, n\}$. Since by specification, propagation times are less than 512 bit times, the station choosing the smaller K will succeed if no other new stations join the contention. A station throws away the packet after it has been collided for 16 times.

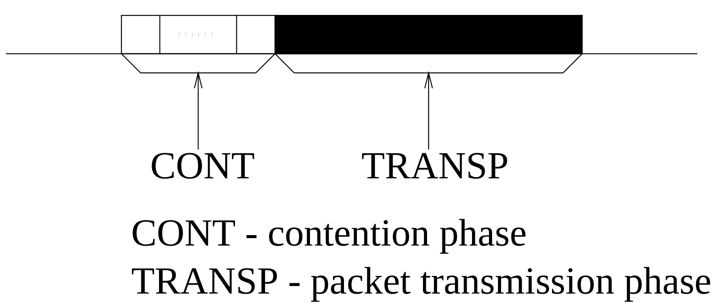

Figure 2.1: Two Phases of a Packet Transmission

The successful transmission of a packet consists of two phases: contention phase and transmission phase as shown in Figure 2.1. Thus the efficiency of the CSMA/CD protocol is given by [15]

$$\eta_{CSMA/CD} = \frac{TRANSP}{CONT + TRANSP}$$

$$= \frac{1}{1 + \frac{CONT}{TRANSP}} \tag{2.1}$$

where $CONT$ = (number of contention slots) $\times$ (contention slot time). In Ethernets, owing to their collision detection capability, the contention slot time is equal to the round-trip propagation time, namely, $2PROP$. This is very small compared to the packet transmission time. The average number of contention slots is measured to be 2.5 by simulations. The efficiency is, therefore, given by

$$\eta = \frac{1}{1 + \frac{5PROP}{TRANSP}} \tag{2.2}$$

Since one Ethernet frame cannot exceed 1518 bytes, the maximum efficiency CSMA/CD protocols can achieve is 82.9%, which is significantly higher than that of Aloha protocol. The success of CSMA/CD is due to the fact that collision detection can be done in very short time, resulting in very small amount of time wasted for contention.

2.3 Token Ring

Token ring is another class of popular MAC protocols for wired local area networks. Stations are attached to a unidirectional ring, around which a *token* circulates. The station possessing the token can transmit packets for a period called the *token holding time(THT)*.

There are two types of token-ring protocols as distinguished by when the token is released: *release after transmission*(RAT) and *release after reception*(RAR) [15]. In RAT protocols, a station releases the token when it has held it for longer than THT and finishes transmission of a packet. In RAR protocols, a station releases the token after it has been kept longer than THT, and the last tranmitted packet has been received. Given that there are N stations connected to the ring and every station has packets to transmit once the token arrives, the efficiencies of RAT and RAR protocols are, respectively [15],

$$\eta_{RAT} = \frac{1}{1 + \frac{PROP}{N \times THT}} \tag{2.3}$$

$$\eta_{RAR} = \frac{1}{1 + \frac{PROP}{THT}} \tag{2.4}$$

where $PROP$ is the propagation time around the ring. Clearly, the efficiency of RAT is larger than that of RAR. However, when the ratio $PROP/THT$ is small, using RAR does not cause much loss in efficiency. Besides, it saves acknowledgement transmission time by making the receiver attach the acknowledgement to the original packet.

2.4 Window Protocol

In *window protocols*(WP) [3, 4, 5], stations that have packets to transmit generate random numbers called *contention parameters* at the beginning of a *contention period.* The contention period consists of a sequence of contention slots followed by the successful transmission of the winning station. It is assumed that each station is equally likely to transmit in the contention period, so the contention parameters are uniformly distributed in a predefined range, e.g. $(0, 1]$. Note that the contention parameters are fixed during one contention period. A global window is maintained by every contending station for determining the set of stations that can transmit in one slot. Stations with contention parameters in the window transmit short signals to contend for the channel. There are three possible channel states: *collision, idle* and *success.* If collision occurs, which means at least two contention parameters are in the same window, the window interval is reduced in order to isolate one station. If the channel is idle, which means that no contention paramters reside in the window, stations switch to the other half of the window. The process is repeated until one station successfully captures the channel.

An example illustrating how WP works is shown in Figure 2.2. The lower bound and upper bound of the current contending window are denoted by l and w, respectively, and the upper bound of the last collision window is recorded in u. It illustrates the steps involved in one contention period. In Contention Slot 1, each of the four contending stations generates a random contention parameter in $(l, u]$, and sets the initial window to $(l, w]$. Stations 1 and 2 have contention parameters in window $(l, w]$ and proceed with transmission, while Stations 3 and 4 keep quiet since their contention parameters are out of the window. As two stations transmit simultaneously, collision is detected. In

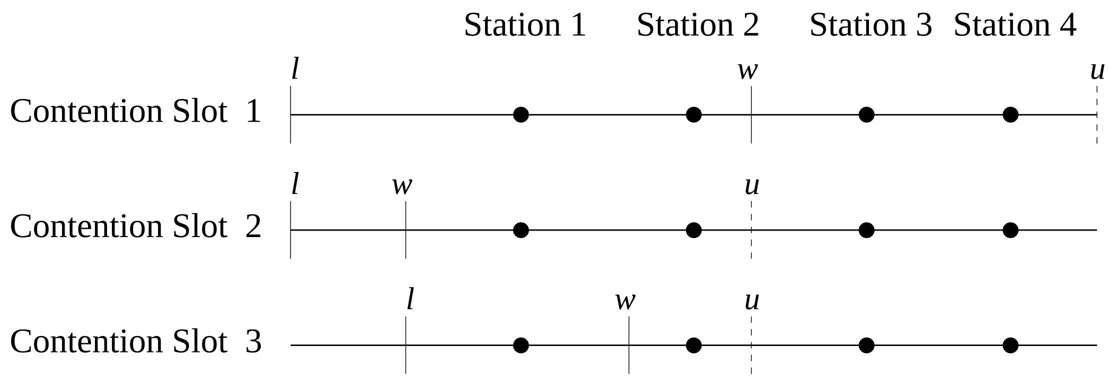

Figure 2.2: One Contention Period in Window Protocol

Contention Slot 2, u is adjusted to w to record the upper bound of the collision window . The contending window is reduced by setting the upper bound w to some value in (l, u). Stations 1 and 2 are out of the window, so the channel is sensed to be idle in this step. In Contention Slot 3, the contending window is changed to the other half by setting lower bound l to w and upper bound to $w \in (l, u]$. Only Station 1 transmits in this slot, so successful transmission is detected.

The window is adjusted according to the objective that the number of future contention slots is minimized. Simulations have shown that the number of slots to resolve collision is around 2.4, independent of the number of contending stations. The efficiency of the window protocol follows

$$\eta_{WP} = \frac{1}{1 + \frac{4.8PROP}{TRANSP}} \tag{2.5}$$

where $PROP$ and $TRANSP$ are defined in Section 2.2. Clearly, it is an improvement over CSMA/CD.

2.5 Influences on Wireless MAC Protocols

In summary, MAC protocols for local area networks can be classified into three categories: contention based, contention free and contention limited. In a contention based scheme, every ready station can transmit once it senses the channel to be free; thus, collisions occur because of simultaneous transmissions by multiple stations. A collision-resolution strategy is required in this class of protocols. CSMA/CD is an example of contention based protocols that adopts exponentail backoff in its collision resolution algorithm. With a contention free scheme, at most one station can access the channel at any time so that no collision will occur, but a channel arbitration algorithm must be employed to choose which station has access to the channel. Token ring is an example in this class of protocols, and only stations in possession of a token are allowed to transmit. A cotention limited sheme is a hybrid of the first two protocols. At any time, only a subset of ready stations are allowed to contend for the channel based on load estimation. The window protocol belongs to this class. The subset of contending stations is chosen based on channel load, which is reflected in the window size.

Nearly every proposed wireless MAC protocol has its origin in wired MAC protocols. *Distributed foundation wireless medium access control*(DFWMAC), the MAC part of the upcoming IEEE 802.11 standard, is based on a CSMA scheme with a rotating backoff window and an optional *request to send*(RTS) and *clear to send*(CTS) message exchange. It is a contention based protocol. *Packet reservation multiple access*(PRMA) is derived from R.Aloha protocol. *Basic access protocol solutions for wireless* (BAPU) is a multi-token multi-access collision avoidance protocol that has its root in both CSMA/CD and token ring. *Random addressed polling*(RAP) is a polling-based protocol that incorporates multiple access. Both BAPU and RAP are contention limited protocols. *Wireless window protocol*(WWP) discussed in this thesis is derived from WP and is a contention limited protocol.

Chapter 3

Wireless Medium Access Overview

3.1 Issues in Wireless MAC Design

Some issues arising from the special features of wireless media have greatly influenced MAC design. These factors are addressed next:

1. Long Collision Detection Time – Collision detection consumes more time than in wired LANs as wireless devices are unable to listen while sending with only one antenna, so that collided transmissions cannot be stopped immediately as in Ethernets. A collision is detected when the acknowledgement from the receiver does not arrive in prescribed time. This means that the detection time cannot be smaller than the round-trip time of a packet transmission.

2. Hidden Terminal Problem – A hidden station that is in the range of the receiver but out of the range of the sender can potentially corrupt an ongoing transmission. It is illustrated in Figure 3.1.

   Assuming that A wants to initiate transmission to B, since no other station in A's range is transmitting, A begins sending packets. At the same time, C is sending to B. Simultaneous transmissions by A and C collide at B. This is due to the fact that A is unable to detect transmission by its hidden station C. This scenario can cause performance degradation in CSMA type protocols.

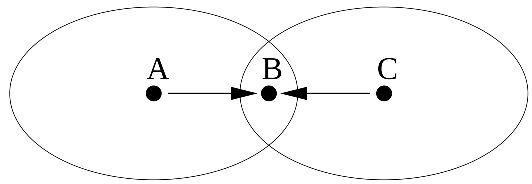

Figure 3.1: Hidden Station Scenario

3. Exposed Terminal Problem – An exposed station that is in the range of the sender but out of the range of the receiver may give up transmission unnecessarily. It is illustrated in Figure 3.2.

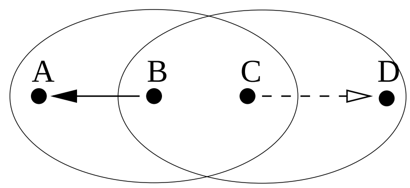

Figure 3.2: Exposed Station Scenario

While B is transmitting to A, since C is in the range of B it cannot initiate another transmission to D although packets from C to D will not collide with packets from B to A as A is out of the range of C. In this scenario, exposed station C is deferring unnecessarily, leading to lower throughput.

4. Interference among Cells Sharing the Same Channel – If two cells are using the same channel, stations along the boundary in one cell can potentially interfere with stations in the other cell.

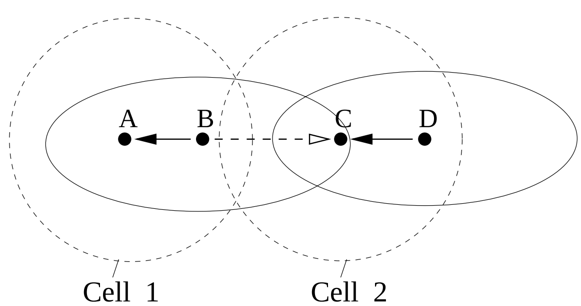

Figure 3.3: Interference

The effect of interference is illustrated in Figure 3.3. In Cell 1, B is the sender and A is the receiver. In Cell 2, D is the sender and C is the receiver. B is near the boundary of Cell 1, and its transmission can interfere with transmissions from D to C so that D's packets cannot be correctly received by C.

3.2 DFWMAC

IEEE 802.11 working group has selected DFWMAC as the draft standard for wireless MAC. DFWMAC combines two coordination funtions – a *point coordination function*(PCF) and a *distribution coordination function*(DCF) [16]. The former is intended for synchronous services with bounded delay and periodic timing constraints. The latter is aimed at bursty traffic such as data. These two modes share the medium in a time multiplexed fashion realized by a superframe structure. Access mechanisms proposed in DCF are studied in this section.

DCF integrates two access methods : basic access and RTS/CTS exchange. In basic access, a station senses the channel before initiating a transmission. If the medium is determined to be idle for an interval that exceeds the *distributed interframe space*(DIFS), the station transmits. An *immediate positive acknowledgement* is sent by the receiver to confirm that the transmission is successful after the channel is idle for longer than a time interval called the *short interframe space*(SIFS), which is less than DIFS, immediately following the reception of the packet. The process is shown in Figure 3.4. If the acknowledgement does not arrive in a timeout period, a retransmission is scheduled according to a backoff window. Every time DIFS is detected, the backoff counter is decremented by one. The station cannot initiate another transmission before the backoff counter reaches zero.

A basic access depends on reliable carrier sense. However, the hidden terminal scenario inherent in carrier sense in wireless media degrades the performance of this mechanism. An optional RTS/CTS exchange is introduced to tackle the hidden terminal problem. The transmission proceeds as in Figure 3.5. The station, after the DIFS in-

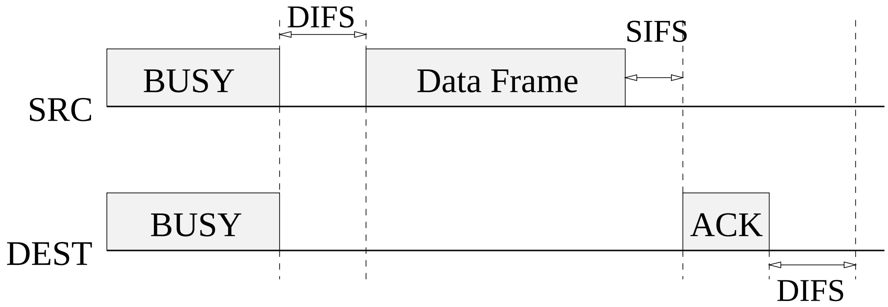

Figure 3.4: Basic Channel Access

terval and before transmitting the data frame, sends a short RTS frame containing the duration of transmission. The receiver, if not deferring to prior transmissions, waits until the medium is idle for SIFS, and sends a CTS frame together with the expected transmission duration. After another SIFS, the sender proceeds with the data frame. Then the receiver responds with an acknowledgement to confirm that the transmission is successful. Stations in the range of either the sender or the receiver read the duration from RTS or CTS and keep silent during the transmission.

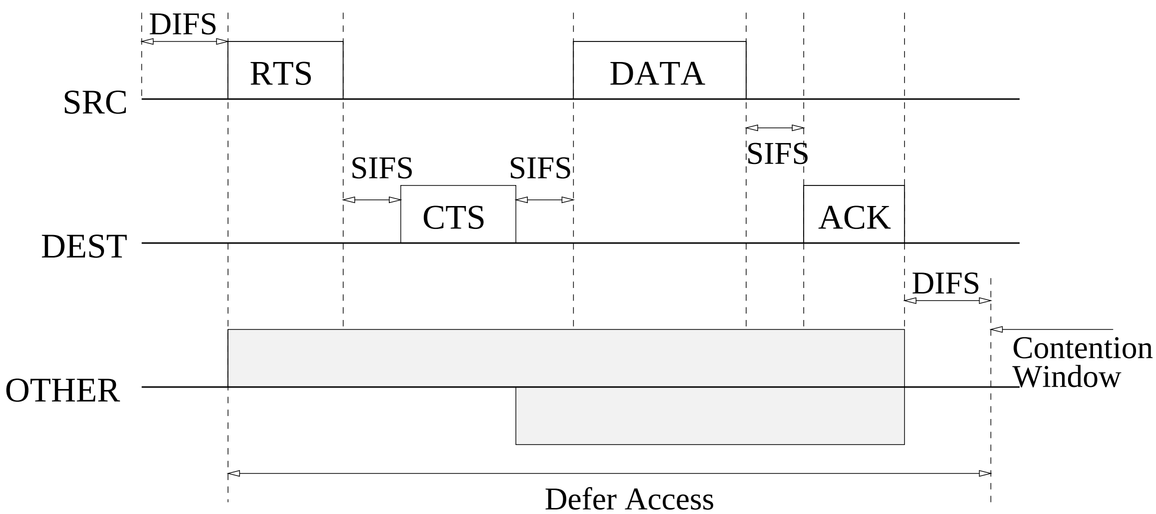

Figure 3.5: RTS/CTS Exchange

The effects of exchanging RTS and CTS prior to data transmission can be summaried as follows.

- The exchange increases throughput because data packets are protected by RTS and CTS, leading to reduced chances of collision.

- It decreases throughput since it involves two more control packets that do not contribute to the effective bandwidth.

- It decreases throughput since it reserves unnecessary time for ongoing transmissions. Suppose in Figure 3.6, B is the sender and C is the receiver. After B sends RTS to C, A knows it should wait until the entire RTS-CTS-DATA-ACK exchange has been completed before transmission. It is not efficient in that A should be able to proceed while B is transmitting data because A's transmission is not colliding with B's transmission to C.

Because of the above tradeoffs of RTS/CTS exchange, DFWMAC allows its usage but does not require it. It is turned on only when packet size exceeds some threshold.

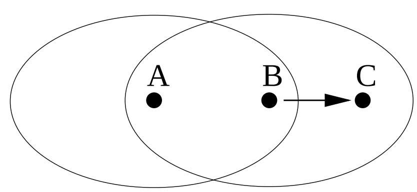

Figure 3.6: Example Illustrating the Negative Effect of RTS/CTS Exchange

3.3 PRMA

PRMA [9] is a protocol designed for transmitting a mixture of voice packets and data packets over short range radio channels. It is assumed that all mobiles use a single channel to send packets to a central base station. The base station transmits a short acknowledgement in the contention free downlink channel. Uplink and downlink share the channel by time multiplexing or frequency multiplexing. The upstream channel is organized into time slots, commensurating with the periodic rates of voice packets. Every slot is marked as *reserved* or *available* according to the acknowledgement. At the beginning of the transmission, the station uses slotted Aloha to contend in an *available* slot. If it succeeds in acquiring the channel, it can specify the slot as *reserved* and no subsequent collisions will occur in this slot. After the transmission is over, the station releases the slot.

PRMA has the advantage of accomodating periodic and random information. However, when the number of active terminals exceeds the number of slots in a frame, stations activated after all the slots have been reserved suffer from starvation. The frame size is difficult to choose especially in dynamic loading conditions. The *reservation based multiple access with variable frame length*(RMAV) [8] addresses this problem by using variable length frames.

3.4 RAP

RAP [18, 19] integrates polling and multiple-access. The nice feature of the protocol is that a base station only polls the active mobiles under its coverage. It has two parameters: p denoting the upper-bound of the randomly generated numbers by mobile stations and L representing the number of times that mobile stations submit their random numbers in each polling cycle. These parameters must be chosen before the RAP protocol is applied. One polling cycle involves the following steps:

1. The base station transmits a READY signal to all the mobiles in its cell.

2. Each active mobile randomly generates a number from the set $I_R = \{0, 1, \ldots, p-1\}$ and sends the number to the base station. All random numbers are sent simultaneously using *code division multiple access*(CDMA) or FDMA. This step is repeated for L times.

3. The base station selects stage l with the largest number of distinctive random numbers – n_l, and informs mobiles that they will be polled by the random numbers from stage l. If n_l is equal to the number of active mobile stations, no collision occurs. If n_l is smaller, the collided stations join future polling.

Simulations have shown that this protocol offers good throughput, low delay, fair access and seamless services in multi-cell infrastructures. Its limitation is that it relies on fixed channel allocation schemes, like CDMA and FDMA, to transmit random numbers simultaneously.

3.5 BAPU

BAPU [22] is a hybrid of multiple-access and token-based protocol evolved from MACAW [23]. It combines collision avoidance, collision resolution, multiple tokens, dynamic addressing and security schemes into one framework.

Collision avoidance is accomplished by the RTS-CTS-DS-Data-ACK packet sequence, where DS denotes "data sending." It adds a DS packet to alleviate the problem of unnecessary deferral as discussed in Section 3.2. Once a station overhears a RTS packet, it only defers for a period long enough for the sender to receive the CTS packet as opposed to the deferral of the entire transmission as shown in Figure 3.5. Only when the station overhears the DS packet, it waits for subsequent data transmissions. RTS and DS solve the exposed sender problem, and CTS solves the exposed receiver and hidden sender problems. Hidden receiver and interference are dealt with by introducing two channels: *control channel* and *data channel*.

In collision resolution, contention measurement and backoff propagation techniques are combined to reflect the local congestion level around a station, which ensures statistically fair access to the channel.

Collision avoidance and collision resolution algorithms constitute the multiple-access part. Multiple tokens are built on top of the multiple-access method in the following fashion. Several tokens are distributed in one cell, and those stations with tokens are permitted to contend for the channel with multiple-access algorithms. Tokens are introduced to achieve scalability and priority support.

The dynamic addressing scheme in BAPU provides locally unique addressing and location privacy. Short MAC addresses reduce control overhead as well. Security protocols are optionally supported in BAPU.

Simulation results have demonstrated that BAPU yields higher throughput and lower delay variance than RTS/CTS exchange of DFWMAC. Performance improvement is achieved at the expense of simplicity though. Also for small size packets, the exchange of RTS, CTS and DS constitutes large overhead.

3.6 Summary

Wireless MAC protocols can be categorized into token-based and multiple-access. Token-based schemes rely on the knowledge about the network configuration in that the base station needs to know which mobiles are under its coverage; in some cases each mobile needs to know who are in its neighbourhood. This configuration information is hard to obtain in a dynamically changing environment like wireless networks, which makes the pure token-based scheme unattractive. Multiple access protocols offer the flexibility of allowing a random number of mobiles to share the channel without the knowledge of network topology. However, it lacks the support of QoS for time bounded services. Therefore, IEEE 802.11 working group integrated a token-based PCF function with DFWMAC to meet the demands of both synchronous and asynchronous services. PRMA aims to provide services for mixed traffic of data and voice packets, but the frame size for the protocol is difficult to choose. In addition, the contention phase for an *available* slot is inefficient. RAP is a polling based protocol that doesn't depend on prior knowledge about network topologies. However, its dependence on fixed allocation schemes, like CDMA and FDMA, limits its use to networks with such physical layer support. BAPU is a unified framework of token-based and multiple-access schemes that solves all the hidden sender, hidden receiver, exposed sender and exposed receiver problems. However, its complexity may make it hard to implement in practice.

Chapter 4

Wireless Window Protocol

4.1 Motivation

Due to the popularity of Ethernets, CSMA/CA is a class of protocols being actively investigated. But carrier sense is difficult in a wireless setting because of the limited range of radio wave propagation. On Ethernets, channel states can be correctly and globally detected by measuring the current from the cable. In contrast, in a wireless environment, channel states detected by different stations can be quite different; e.g., the sender may sense the channel to be free while collision actually happens at the receiver in the presence of a hidden terminal. In order to solve the hidden sender and receiver, and exposed sender and receiver problems, some control packets, such as RTS, CTS and DS [23], must be sent prior to data transmission.

Hidden and exposed station problems are inherent in CSMA/CA type of protocols. Our proposed wireless window protocol(WWP) takes a different approach in which mobile stations don't sense the carrier in order to determine whether to transmit or not; instead, channel state is broadcast by the base station so that every mobile has the correct information about the carrier. Traffic is classified into *uplink* and *downlink*. The uplink is shared by mobiles so that there is contention for the channel while downlink is contention free. The base station is in the range of every mobile station in the cell, and has correct information about the channel state, which is broadcast in the downlink channel. The uplink and downlink can share the channel in a time multiplexed or frequency multiplexed fashion.

4.2 WWP for One Cell

WWP is similar to WP for the one cell case. Figure 4.1 illustrates the steps taken in a contention period. First, every station randomly generates a contention parameter and initializes its window to a common global window. A station transmits a short control packet(CP) if its contention parameter lies in the window and otherwise keeps quiet. It updates the window based on the three possible channel states broadcast by the base station. If the channel state is either idle or collision, the window needs to be updated. The contention period is concluded with a SUC packet, in which the duration of the subsequent data transmission is included so that other stations know how long to wait before contending in a new contention period.

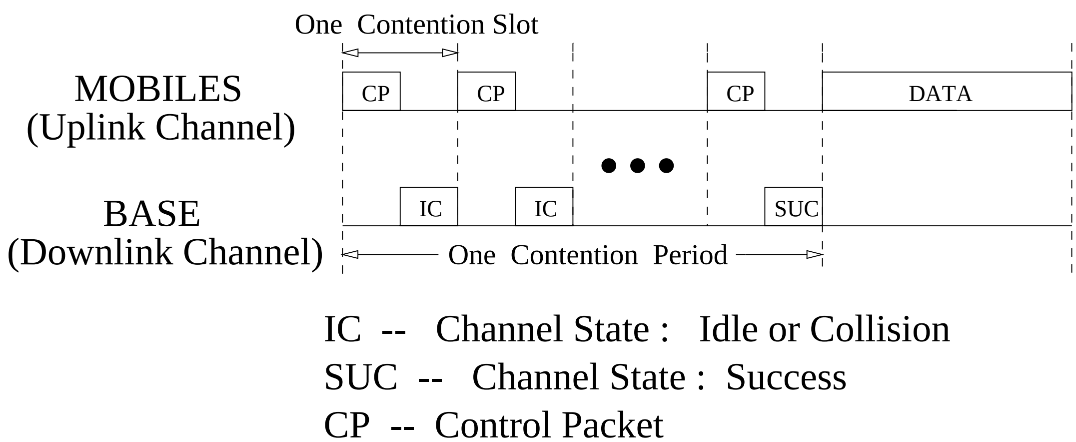

Figure 4.1: Illustration of Wireless Window Protocol

WWP has a major difference from WP on Ethernets. On Ethernets, because stations can listen while sending, and stop transmission immediately after detecting simultaneous transmissions. Hence, contention and collision detection can be completed in one contention slot. In contrast, in wireless LANs, mobile stations rely on the base station to broadcast the state of contention one slot later. This affects the performance and control of the window protocol. We utilize this property and discuss lookahead WWP in Section 4.4.

Section 4.2.1 gives the state diagram of the protocol. Since the key aspect of the protocol is the adjustment of windows based on the channel state, Section 4.2.2 discusses the dynamic programming formulation of window adjustment. Window updates are based on the current channel load, and Section 4.2.3 presents the load estimation algorithm. Because of the high computational overhead of dynamic programming, Section 4.2.4 describes the algorithm to compute lookup tables ahead of time and compares the number of contention slots under different *probability distribution functions*(PDFs) of contention parameters.

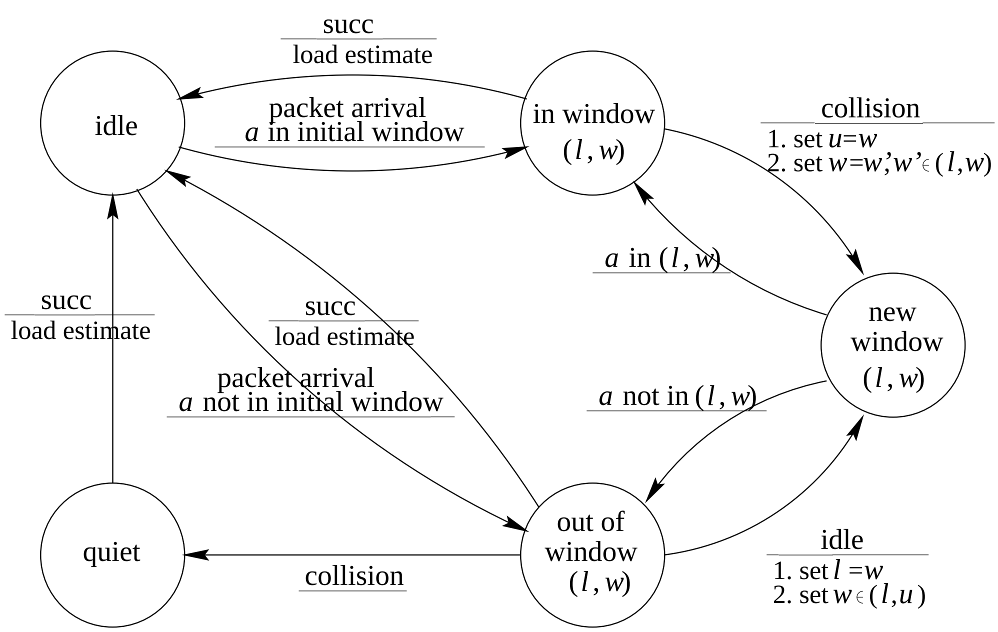

Figure 4.2: State Diagram of Wireless Window Protocol

4.2.1 Protocol Description

Figure 4.2 shows the state diagram of the wireless window protocol. A station can be in one of the following five states: *idle, quiet, in-window, out-of-window* and *new-window.* All stations are in the *idle* state between contention periods. In each contention period, stations that have been ruled out in previous slots are in the *quiet* state and wait for the end of the current contention period. Stations are in the *new-window* state after the window update in each slot. The *in-window* and *out-of-window* states are self-explanatory. Let (l, w) denote the window that is maintained by each station and u denote the upper bound of the window that has resulted in collision. A station sits in the *idle* state before the contention period begins. At the beginning of the contention period, each station randomly generates a contention parameter a and goes to either the *in-window* state or the *out-of-window* state, depending on whether a lies in the window or not. In the *in-window* state, on hearing collision from the channel, a station tightens the window by moving the upper bound towards the lower bound and switches to the *new-window* state. On hearing success from the channel, it performs load estimation and returns to the *idle* state. In the *out-of-window* state, it switches to the *new-window* state, *quiet* state or *idle* state when it senses idle, collision or success, respectively. In the *new-window* state, a station goes to either the *in-window* or the *out-of-window* state, depending on whether a resides in the new window or not. In the *quiet* state, the station is not allowed to participate in contention. Only when it senses a success signal passing by, it returns to the *idle* state and waits for the beginning of another contention period.

Several remarks about the WWP are as follows. First, since the channel state is known to all the stations and they execute the same algorithm driven by the state diagram shown in Figure 4.2, the same window can be maintained in all the stations if they start from the same initial state. Second, the algorithm shown above basically finds the minimum number among a set of contention parameters. It can be modified to find the maximum number as well. Third, the performance of the protocol depends on how the window is adjusted. In the next section, we study the window adjustment scheme by a dynamic programming formulation.

4.2.2 Window Adjustment

When the channel is sensed to be either *collision* or *idle*, the currently maintained window needs to be updated. However, the channel being idle implies that there is collision in the other half of the window as shown in Contention Slot 3 of Figure 2.2. Thus the window adjustment algorithm is reduced to finding the next window $(a, w) \subseteq (a, b)$ if collision occured in the current window $(a, b]$

Suppose n denotes the number of initial contending stations for the contention period. Let's define the following notations:

$N(a, b)$:	the minimum expected number of future slots to resolve contention, given that collision occurs in the current window $(a, b]$,
$P_{succ}(a, w, b, n)$:	probability of success in the next slot if window $(a, w]$, $a < w < b$, is used,
$P_{col}(a, w, b, n)$:	probability of collision in the next slot if window $(a, w]$, $a < w < b$, is used,
$P_{idle}(a, w, b, n)$:	probability of channel being idle in the next slot if window $(a, w]$, $a < w < b$, is used.

$N(a, b)$ can be derived recursively from the following formula:

$$\begin{aligned} N[a, b] \;=\; \min_{a<w<b} \; & \{1 + 0 \times P_{succ}(a, w, b, n) + N[a, w] \times P_{col}(a, w, b, n) \\ & + N[w, b] \times P_{idle}(a, w, b, n) \; \} \end{aligned} \tag{4.1}$$

$P_{succ}(a, w, b, n)$ is not needed for computing $N[a, b]$. $P_{col}(a, w, b, n)$ and $P_{idle}(a, w, b, n)$ can be derived from conditional probabilites as follows. Assuming $F(.)$ is the PDF of contention parameters:

$$P_{col}(a, w, b, n) = \frac{(1 - F(a))^n - (1 - F(w))^n - n(F(w) - F(a))(1 - F(w))^{n-1}}{(1 - F(a))^n - (1 - F(b))^n - n(F(b) - F(a))(1 - F(b))^{n-1}} \tag{4.2}$$

$$P_{idle}(a, w, b, n) = \frac{(1 - F(w))^n - (1 - F(b))^n - n(F(b) - F(w))(1 - F(b))^{n-1}}{(1 - F(a))^n - (1 - F(b))^n - n(F(b) - F(a))(1 - F(b))^{n-1}} \tag{4.3}$$

Details on how to arrive at the above equations can be found in [5]. It follows that once the channel state and contention parameter PDF are known, an optimal window can be calculated in each slot. However, the recursive algorithm has high computational overhead and is difficult to be carried out in real time. We study how to precompute $N(a, b)$ in Section 4.2.4 .

4.2.3 Load Estimation

In the evaluation in the previous subsection, it is assumed that the number of initial contending stations, n, is known. Therefore in WWP, load estimation needs to be done before every contention period. The estimate is based on the contention parameter PDF and the previous window that has successfully isolated one station.

At the end of the t'th contention period, the window $(l, w(t))$ that has successfully isolated one station is known to all the stations. The probability of minimum number Y lying in $(l, w(t))$ can be derived as follows, assuming that contention parameters are independently and uniformly distributed over interval (l, u),

$$\begin{aligned} Q(\hat{n}, w(t)) &= Prob(l < Y < w(t)) \\ &= \hat{n}(w(t) - l)(u - w(t))^{\hat{n}-1}/(u - l)^{\hat{n}} \end{aligned} \tag{4.4}$$

$Q(\hat{n}, w(t))$ is maximized when

$$\hat{n} = \frac{-1}{\log{(u-l)} - \log{(u - w(t))}} \tag{4.5}$$

The number of stations that will participate in the *(t+1)*'th contention period can be estimated by adding $(\hat{n} - 1)$ to the number of arrivals during the t'th contention period.

4.2.4 Generation of Lookup Table

Although the dynamic programming formulation is optimal in terms of minimizing the number of future contention slots, it incurs high computational complexity, which makes it infeasible for real time evaluation. In wireless systems, the roundtrip propagation time between terminals and the base station is less than one microsecond indoors, and packet duration is on the order of a few milliseconds for 1Mbps infrared radio channels. This gives every contention slot a duration of a few milliseconds. A lookup table has to be generated in advance for each given n in order to meet the timing requirement. In this section, we discuss how to efficiently compute the lookup table and the impacts of different contention parameter PDFs on the lookup table.

4.2.4.1 Efficient Implementation

$N(a,b)$ can be computed as in Eq. 4.1. However, there are two implementation details that need to be considered:

1. The dynamic programming formulation is continuous, and boundary conditions must be set to terminate the evaluations after a certain number of iterations. It is assumed that when the window size is less than $\frac{1}{10n}$, where n is the number of initial contending stations, the probability of more than two stations having contention parameters in this window is so small that contention can always be resolved in one slot, i.e.,

$$N(a,b) = 1 \quad \text{when } (b-a) < \tfrac{1}{10n} \tag{4.6}$$

2. Instead of calculating $N(a,b)$ in a forward fashion, we choose to compute it in a backward way. The advantage of this method is that it avoids repetitive evaluations of the same subproblem as would be required by the first approach.

From the above discussion, it is clear that the original dynamic programming formulation is reduced to finding the upper right triangle of the matrix shown in Figure 4.3 with the entries along the diagonal being filled with one by assumption. We calculate $N(i,j)$ for $|j-i| = 1, 2, \ldots, 10n$ successively, proceeding from entries near the diagonal to the upper right corner.

$N(0,10n)$(the shaded area in Figure 4.3) gives the expected minimum number of contention slots needed to isolate one station, given that n stations participate in the contention initially. Table 4.1 shows $N(0,10n)$ for different n. The results indicate that statistically WWP has load independent behavior, which is a desirable feature as compared to the CSMA/CA protocol that only behaves satisfactorily when the channel load is low.

It is straightforward that the lookup table for a given n can be generated in $O(n^2)$ using the above algorithm.

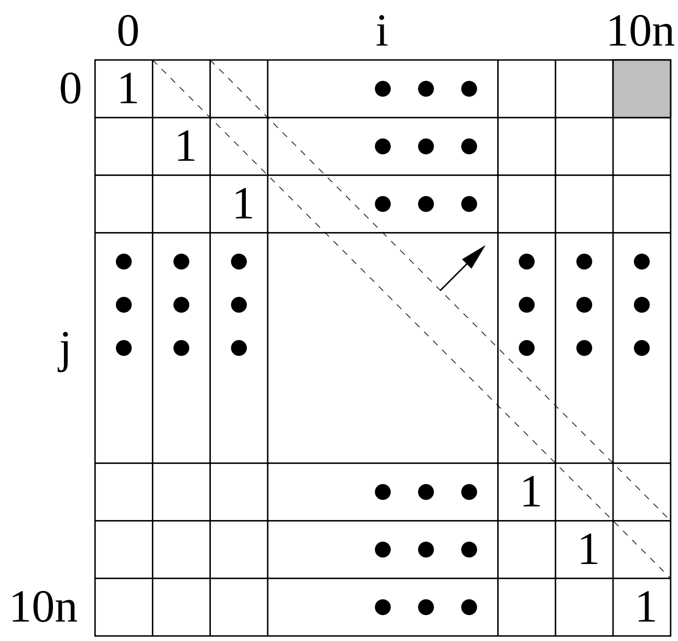

Figure 4.3: Computing Lookup Table

4.2.4.2 Choices of Contention Parameter PDF

It is assumed that contention parameters are uniformly distributed over interval $(0, 1]$ in previous subsections. We now investigate how different contention parameter PDFs can affect the average number of slots to resolve collision.

Increasing PDF Let's define *probability density function*(pdf) $f_1(x)$ as in Figure 4.4 and Eq. 4.7,

$$f_1(x) = \begin{cases} \frac{2}{M^2}x & 0 \leq x \leq M \\ 0 & x > M \end{cases} \tag{4.7}$$

where $M = 10n$.

$f_1(x)$ is a valid pdf since it satisfies

$$\int_0^M f_1(x)dx = 1 \tag{4.8}$$

Table 4.1: Number of Contention Slots for Different Number of Contending Stations

n	contention slots: $N(0, 10n)$
5	2.257
10	2.340
20	2.380
25	2.388
50	2.404
100	2.411

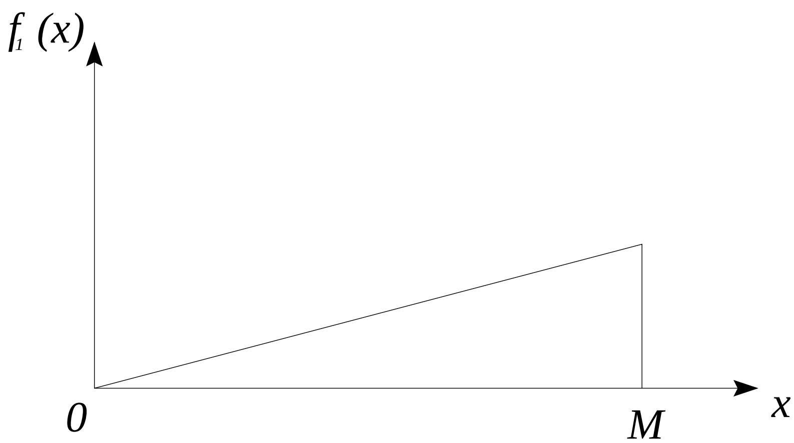

Figure 4.4: Probability Density Function $f_1(x)$

The corresponding *probability distribution function* $F_1(x)$ follows,

$$F_1(x) = \begin{cases} 0 & x < 0 \\ \frac{1}{M^2}x^2 & 0 \leq x \leq M \\ 1 & x > M \end{cases} \tag{4.9}$$

Substituting $F_1(x)$ into Eqs. 4.2 and 4.3, a lookup table can be reconstructed. We are interested in comparing $N(0, 10n)$s for two different PDFs, since $N(0, 10n)$ gives the average number of contention slots for a given n. Table 4.2 illustrates how results differ for two PDFs. It appears that less number of contention slots are needed if a uniform distribution is used.

Table 4.2: Comparison of Number of Contention Slots for $F_1(x)$ and Uniform Distribution

n	$N(0, 10n)$(uniform)	$N(0, 10n)(F_1(x))$
5	2.257	2.260
10	2.340	2.358
20	2.380	2.401
25	2.388	2.412
50	2.404	2.431
100	2.411	2.442

Decreasing PDF Let's define pdf $f_2(x)$ as in Figure 4.5 and Eq. 4.10,

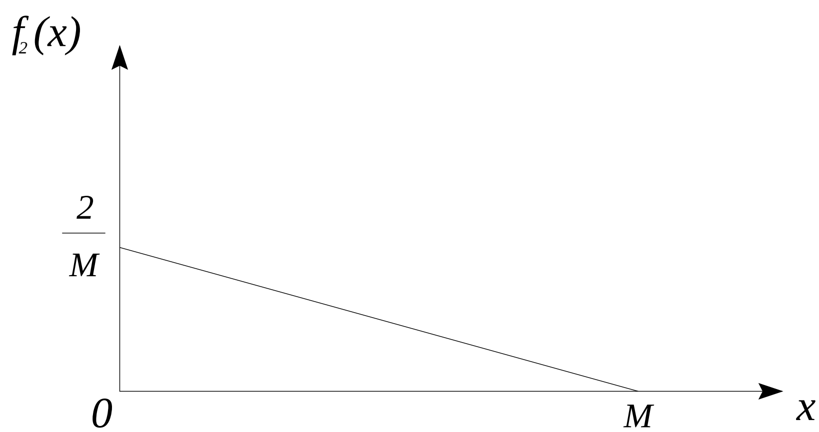

Figure 4.5: Probability Density Function $f_2(x)$

$$f_2(x) = \begin{cases} \frac{2}{M} - \frac{2}{M^2}x & 0 \le x \le M \\ 0 & x > M \end{cases} \tag{4.10}$$

where $M = 10n$.

It can be verified that $f_2(x)$ is also a valid pdf by,

$$\int_0^M f_2(x)dx = 1 \tag{4.11}$$

The corresponding *probability distribution function* $F_2(x)$ is,

$$F_2(x) = \begin{cases} 0 & x < 0 \\ \frac{2}{M}x - \frac{1}{M^2}x^2 & 0 \le x \le M \\ 1 & x > M \end{cases} \tag{4.12}$$

Substituting $F_2(x)$ into Eqs. 4.2 and 4.3, a new lookup table can be generated. Table 4.3 gives the comparison of $N(0, 10n)$ for two contention parameter PDFs: $F_2(x)$ and uniform distribution. It seems that $F_2(x)$ can help reduce the number of contention slots.

Table 4.3: Comparison of Number of Contention Slots for $F_2(x)$ and Uniform Distribution

n	$N(0, 10n)$(uniform)	$N(0, 10n)(F_2(x))$
5	2.257	2.226
10	2.340	2.302
20	2.380	2.340
25	2.388	2.347
50	2.404	2.361
100	2.411	2.370

Discussion Tables 4.2 and 4.3 show that different contention parameter PDFs can result in different contention slots $N(0, 10n)$. In this subsection, let's explore the reasons why contention parameter PDFs should have influence on contention slots.

As discussed in Section 4.2.4.1, when generating the lookup table, boundary condition Eq. 4.6 has been set in order to terminate the computation, namely, the initial window $(0, 1]$ is divided into $10n$ sub-intervals with equal lengths. Lookup tables can only be used to resolve collision in the window which is larger than a sub-interval. Although the probability of two or more stations falling in the same sub-interval is very small, it cannot be ignored when evaluating protocol performance. For the purpose of illustration, the following notations are first defined.

$W(n)$: the expected number of contention slots to resolve collision, when there are n contending stations,

$N(n)$: the expected minimum number of contention slots obtained from the lookup table,

$P(n)$: the probability of two or more smallest contention parameters lying inside the same sub-interval of length $\frac{1}{10n}$,

$L(n)$: the expected number of slots to resolve collision in a sub-interval using other techniques, such as binary exponential backoff or binary window division.

It follows from the above definitions that

$$W(n) = N(n) + P(n) \times L(n) \tag{4.13}$$

$W(n)$ is the performance metric that needs to be compared, while Tables 4.2 and 4.3 have merely compared $N(n)$s for different PDFs. Assume that $L(n)$s are identical for different PDFs if the same technique is employed to resolve contention in sub-intervals. $P(n)$s need to be computed for fair comparison of different PDFs. It can be evaluated as follows.

$$\begin{aligned} P(n) \;=\; & \sum_{i=0}^{10n-1} \left(1 - F\left(\frac{i}{10n}\right)\right)^n - n\left(F\left(\frac{i+1}{10n}\right) - F\left(\frac{i}{10n}\right)\right)\left(1 - F\left(\frac{i+1}{10n}\right)\right)^{n-1} \\ & - \left(1 - F\left(\frac{i+1}{10n}\right)\right)^n \end{aligned} \tag{4.14}$$

Let's define

$P_0(n)$: the probability of two or more smallest contention parameters lying inside the same sub-interval, if uniform distribution is used,

$P_1(n)$: the probability of two or more smallest contention parameters lying inside the same sub-interval, if PDF $F_1(x)$ is used,

$P_2(n)$: the probability of two or more smallest contention parameters lying inside the same sub-interval, if PDF $F_2(x)$ is used.

$P_0(n)$, $P_1(n)$ and $P_2(n)$ can be calculated by substituting the uniform PDF, $F_1(x)$ and $F_2(x)$ into Eq. 4.14, respectively. The results are listed in Table 4.4. For PDF $F_1(x)$,

Table 4.4: Comparison of $P_0(n)$, $P_1(n)$ and $P_2(n)$

n	$P_0(x)$	$P_1(x)$	$P_2(x)$
5	0.04933	0.03997	0.08686
10	0.04925	0.02804	0.09206
20	0.04921	0.01977	0.09443
25	0.04920	0.01768	0.09489
50	0.04918	0.01250	0.09579
100	0.04918	0.00884	0.09623

$P(n)$s are smaller than those of the uniform distribution while $N(n)$s are larger as shown in Table 4.2. Decreases in $P(n)$s have offset increases in $N(n)$s in Eq. 4.13. In the case of PDF $F_2(x)$, increases in $P(n)$s have offset decreases in $N(n)$s. It is reasonable to conclude that varying PDF for contention parameters has little influence on the overall protocol performance. For simplicity, a uniform distribution is assumed in the rest of the thesis.

4.2.5 Binary Window Division Phase

Lookup tables are generated based on the assumption stated in Eq. 4.6. When two or more contention parameters reside in an interval less than $\frac{1}{10n}$, lookup tables cannot be used anymore. In WWP, binary window division is employed to further resolve collision in an interval less than $\frac{1}{10n}$; namely, if collision occurs in window $(a, b]$ with $|b - a| < \frac{1}{10n}$, w is set to be $(a + b)/2$ instead of applying Eq. 4.1. The window is reduced by half in each subsequent step.

The overhead of the binary window-division phase can be approximated by calculating the number of slots needed to resolve collision if two numbers are in interval $(a, b]$, since the probability of three or more numbers in a short interval $(a, b]$ is so small that it can be negligible. Let's define $q(j)$ as the probability of j slots to resolve collision. The number of slots N_{binary} is given by

$$N_{binary} = \sum_{j=1}^{\infty} j \times q(j) \tag{4.15}$$

To compute $q(j)$, we begin from the simplest case: $j = 1$

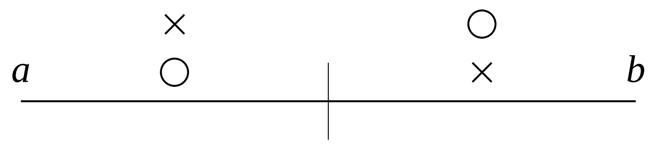

Figure 4.6: Resolving Contentions in One Slot: $j = 1$

As shown in Figure 4.6, two numbers must be in different halves of the interval, otherwise one slot is not enough to resolve contention. The probability of one contention parameter lying in the first or the second half is equal to 1/2 for a uniform distribution. Thus we have

$$q(1) = 2 \times \frac{1}{2} \times \frac{1}{2} = \frac{1}{2} \tag{4.16}$$

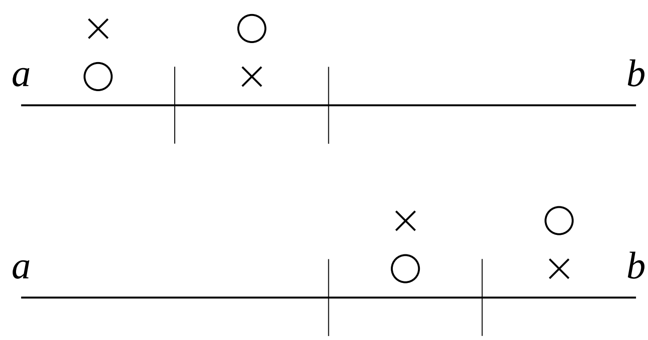

Figure 4.7: Resolving Contention in Two Slots: $j = 2$

Similarly, in case of $j = 2$, both parameters must lie in either the first half or the second half so that one slot is not sufficient for resolving contention. However, they must be in different quarters as shown in Figure 4.7 so that two slots are enough to resolve contention. $q(2)$ is, therefore, given by

$$q(2) = 2 \times \frac{1}{4} \times \frac{1}{4} \times 2 = \frac{1}{4} \tag{4.17}$$

By deduction, $q(n)$ can be calculated as

$$q(n) = 2 \times \frac{1}{2^n} \times \frac{1}{2^n} \times 2^{n-1} = \frac{1}{2^n} \tag{4.18}$$

N_{binary} can, therefore, be evaluated as follows

$$N_{binary} = \sum_{j=1}^{\infty} j \times \frac{1}{2^j} = 2 \tag{4.19}$$

To compute the number of slots needed in the binary division phase, we need to evaluate $P(n)$, the probability that the two smallest numbers reside in the window with the interval length less than $\frac{1}{10n}$, if n stations join the contention period initially.

$$P(n) = \sum_{i=0}^{10n-2} \frac{n(n-1)}{2} \times \left(\frac{1}{10n}\right)^2 \times \left(1 - \frac{i}{10n} - \frac{1}{10n}\right)^{n-2} \tag{4.20}$$

For example, $P(100)$ is equal to 0.048, and approximately $0.096(2 \times 0.048)$ slot is needed in the binary window division phase on the average. This example illustrates that the binary window division does not cause significant increase in the total number of contention slots.

4.3 WWP for Multiple Cells

Wireless LAN systems are constantly changing when mobile users migrate from one cell to another from time to time. Those mobile stations in the intersection of cells can potentially interfere with contentions in adjacent cells. WWP needs to be modified to accomodate the situation. Consider the scenario shown in Figure 4.8.

Cells 1 and 2 are adjacent to each other. Stations 1 and 2 are in Cell 1 initially and walk into the intersection area. Stations 3 and 4 are in the coverage of Cell 2. Suppose four stations begin a contention period simultaneously and their contention parameters are shown in parentheses in the figure. The window of stations in Cell 1 is $(0, 0.2]$, and that in Cell 2 is $(0, 0.3]$. Note that the windows can be different due to different load estimations in the two cells. Stations 1 and 2 have parameters in the window of Cell 1, and they transmit. Stations 3 and 4 refrain from transmission as their parameters are out of the window of Cell 2. However, the base station in Cell 2 hears collision in

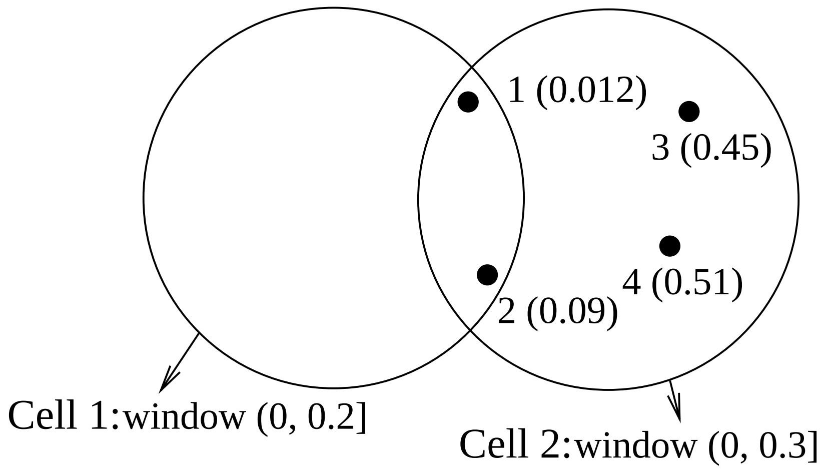

Figure 4.8: Impact of Mobile Stations in the Intersection Area

the uplink because of the transmissions by Stations 1 and 2. It cannot tell whether the collision is caused by stations in its own cell or by stations on the boundary of an adjacent cell. It broadcasts the collision state, so Stations 3 and 4 reduce their window further in subsequent intervals and never get a chance to transmit.

Two mechanisms are proposed to address this problem. The first mechanism, *relaxed upper-bound*, is realized by not reducing the upper-bound u to b after collision in window $(a, b]$ is detected. The basic idea is that future contention windows need to be extended to $(b, 1]$ after a collision is detected in $(a, b]$, if this collision is a wrong channel state caused by interference from an adjacent cell. Let's see how the window of Cell 2 is adjusted using this technique as illustrated in Figure 4.9.

In step (1), Stations 3 and 4 refrain from transmission but hear collision in window $(0, 0.3]$ due to the transmissions by Stations 1 and 2 in Cell 1. In step (2), u is kept at 1 instead of being set to 0.3 as in the original WWP. A new upper-bound $w_2 \in (0, 0.3]$ is chosen according to the lookup table. Stations 3 and 4 do not transmit because their parameters are still out of the window. If Stations 1 and 2 still collide, a collision signal is broadcast at the end of this step. This process continues until one of the stations in Cell 1 has succeeded at step (n). Assume that the base station can tell the success packet is from an adjacent cell, therefore it ignores it and broadcasts an idle state in this step.

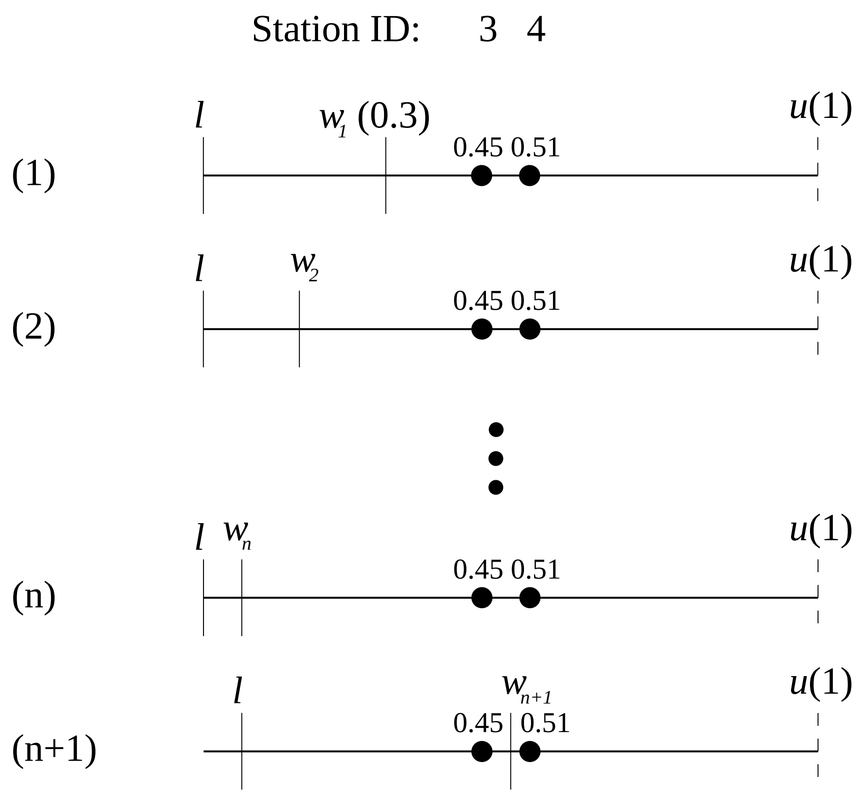

Figure 4.9: New Window-Adjustment Scheme Based on Relaxed Upper-bound for Cell Overlay

In step (n+1), w_{n+1} is selected from the other half, $(w_n, 1]$, and only Station 3 falls in the window. The successful transmission by Station 3 concludes the contention period.

Simulations show that the starvation problem as explained in the second paragraph in this section can still occur to the WWP with upper-bound relaxation, although the chance of its occurance is very small. For example, when the window of Cell 2 is $(0, 0.015]$, since its interval length is less than $\frac{1}{10n}$, $n = 2$, it will be updated by binary window division and reduced by half in subsequent steps. Stations 3 and 4 are kept out of the window, and thus "starved."

The second technique to be investigated is called *bounded contention*. It requires that a base station keeps a count of how many contention cycles have elapsed in the current contention period. If this count exceeds a certain threshold, the base station assumes that the stations in its own cell are suffering from the starvation problem, and it broadcasts a success signal to start a new contention period. Ethernets have employed the same technique to bound the length of the contention phase of a packet transmission. Simulations have verified that this technique completely solves the starvation problem.

4.4 Lookahead WWP

In this section, we consider a design alternative if a single channel is shared by uplink and downlink by frequency division multiplex. In this case, the base and mobile stations can transmit simultaneously by using different frequency bands. The lookahead technique can be used to take advantage of this fact to reduce the number of contention slots.

In WWP, after a mobile transmits a packet, it waits for an acknowledgement packet from the downlink channel. In lookahead WWP, instead of waiting for the result, the mobile station sets the next window based on the estimated channel states. By the time that the channel state corresponding to the first packet transmission comes back, the second packet transmission has been finished with the estimated window. The last two windows need to be kept by each station since the estimated channel state may be wrong. Let $(a_1, b_1]$ denote the previous window and $(a_2, b_2]$, the current one. The detailed lookahead WWP consists of the following steps:

1. Generate random number a and estimate the number of contending stations $\hat{n}$ based on the previous channel load as shown in Section 4.2.3;

2. Initialize $(a_1, b_1]$ by the lookup table for $\hat{n}$ and tranmsit depending on if a lies in $(a_1, b_1]$ or not.

3. Calculate P_{col} and P_{idle} by Eqs. 4.2 and 4.3.

    - If $P_{idle} \geq P_{col}$, set estimated-state = idle, $a_2 = b_1$ and $b_2 = w$ where $w \in (b_1, u)$ and u is the upper bound of the window that has resulted in collision;
    - If $P_{idle} > P_{col}$, set estimated-state = collision, $a_2 = a_1$ and $b_2 = w$ where $w \in (a_1, b_1)$.

Transmit according to window $(a_2, b_2]$.

4. Read the channel state from the downlink channel. If it is success, exit; if the feedback corresponds to the window derived from the incorrectly estimated channel state, then discard the feedback and perform lookahead as in Step 3; otherwise,

   - If channel state = estimated-state, lookahead as described in Step 3;
   - If channel state $\neq$ estimated-state, roll back to window $(a_1, b_1]$ and adjust window to $(a_2, b_2]$ based on the correct channel state.

Intuitively, the lookahead technique reduces the number of contention slots by making use of the time waiting for acknowledgements. The best case happens when every estimation is correct: only half of the slots needed by the original WWP are sufficient to resolve collision. The worst case happens when every estimation is wrong: the same number of contention slots are needed as in conventional WWP. Performance improvement is shown in next chapter.

Chapter 5

Performance Evaluation

5.1 Performance Metrics and Simulation Model

Computer simulations have been done to evaluate WWP's performance and compare WWP with the DCF part of DFWMAC, the draft IEEE 802.11 standard. In our simulations, we are interested in the number of contention slots and the delay characteristics, such as delay time averages, standard deviations and probability distributions. The reasons why we investigate these measures are as follows.

- Channel efficiency is directly influenced by the number of contention slots. It can be expressed as

$$\eta = \frac{1}{1 + cslots \times \frac{slottime}{ptransmission}} \tag{5.1}$$

  where *cslots* represents the number of contention slots, *slottime* represents the duration of a contention slot and *ptransmission* denotes the packet transmission time. Given that *slottime* and *ptransmission* are fixed, η decreases as *cslots* increases.

- Fairness is reflected in the delay characteristics. Long-term fairness requires that the average delay time of a station is proportional to the fraction of the load that the station places on the channel. Assuming equal load conditions, large variations in average delay times across stations always imply the lack of long-term fairness. Short-term fairness requires that the delay variances of all the stations to be small.

Simulations are done with CSIM [20], a discrete event process oriented simulation library. Both base and mobile stations are modeled as *processes*. Each process has a *mailbox* to communicate with others. Uplink traffic is simulated by mobiles sending messages to the base station's mailbox, and downlink traffic is emulated in a similar way. Our similations normally run until the 0.95 confidence interval is reached for every station. It is assumed that each station has equal load, namely, every station has probability p to transmit at the beginning of a contention period. In our simulations, p is set to 1.0.

5.2 WWP in One Cell

This section evaluates the performance of WWP in the one-cell scenario. First we compare the average number of contention slots stored in the lookup table and obtained from our simulations with the binary division phase included. Figure 5.1 shows that binary division phase doesn't introduce significant overhead in the protocol as explained in Section 4.2.5.

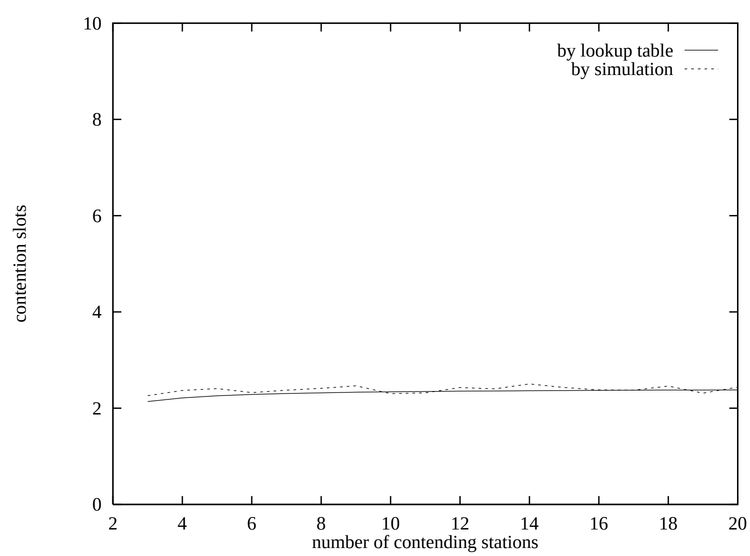

Figure 5.1: Number of Contention Slots by Simulation and by Lookup Table

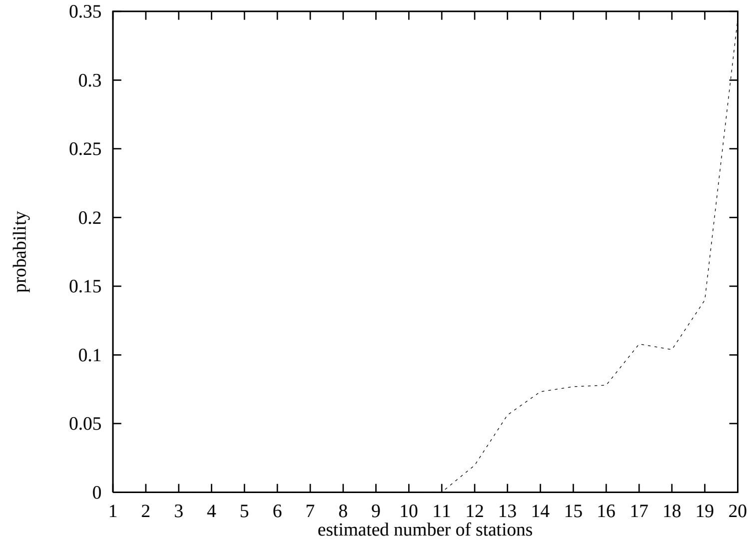

Figure 5.2: Probability Density Function of Channel Estimate

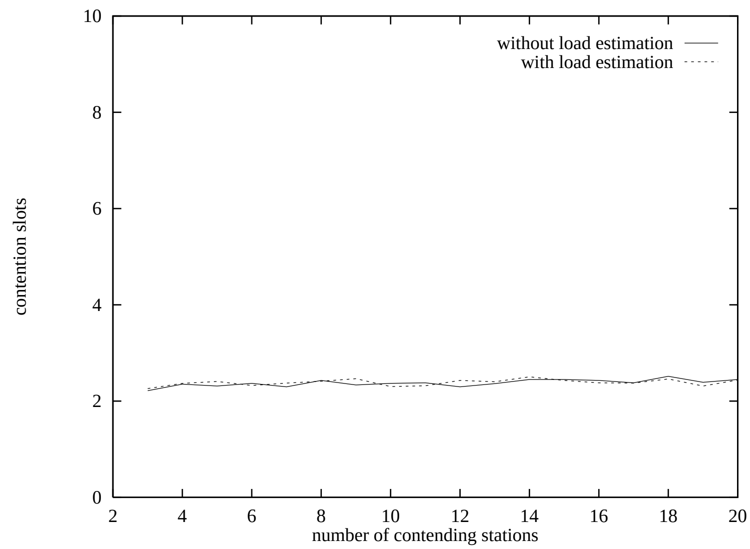

Figure 5.3: Number of Contention Slots with and without Load Estimation

In WWP, channel load $\hat{n}$ is estimated at each contention period and used to determine which lookup table to consult. Figure 5.2 shows the probability density function of the estimated number of stations when the load is fixed at twenty stations. Almost 70% of the estimates are accurate enough (above 17). Figure 5.3 compares the number of contention slots with and without load estimation(by assuming the number of initial contending station to be known a *priori*). These two figures illustrate two points. First,

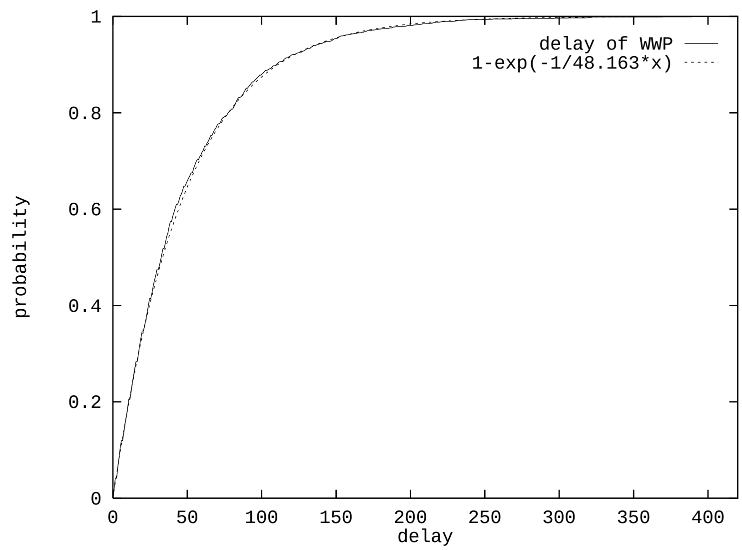

Figure 5.4: Delay Distribution of a Single Station in One Cell

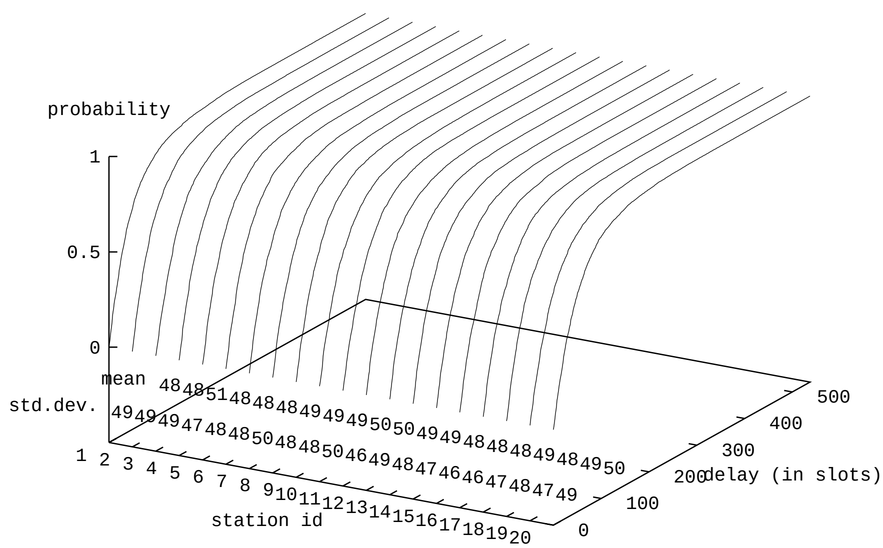

Figure 5.5: Delay Distributions in a 20-Station Cell

the channel load estimation algorithm described in Section 4.2.3 provides good estimates. Second, the protocol is not sensitive to the load parameter, namely, it performs fairly well even with some discrepancies between the estimated load and the actual load.

Last, we consider a cell consisting of twenty mobile stations with identical load. Delay distribution of Station 1 is plotted in Figure 5.4. When tested against an analytical exponential distribution, it gives a perfect fit. This is due to the fact that contentions start anew in every contention period when stations generate a new set of contention parameters, and the protocol doesn't remember the stations which have succeeded before(memoryless property of exponential distribution). Figure 5.5 further shows that the twenty stations have similar delay distributions, which demonstrates both long-term and short-term fairness of WWP.

5.3 WWP in Two Cells

In this section, we evaluate the performance of our modified WWP in a multiple-cell scenario. The experiments were done in the configuration shown in Figure 5.6. There are 10 mobiles in each cell, and they are named $1, 2, \ldots, 10$ in Cell 1 and $11, 12, \ldots, 20$ in Cell 2. Mobiles 1 and 2 are in the overlap area of the two cells.

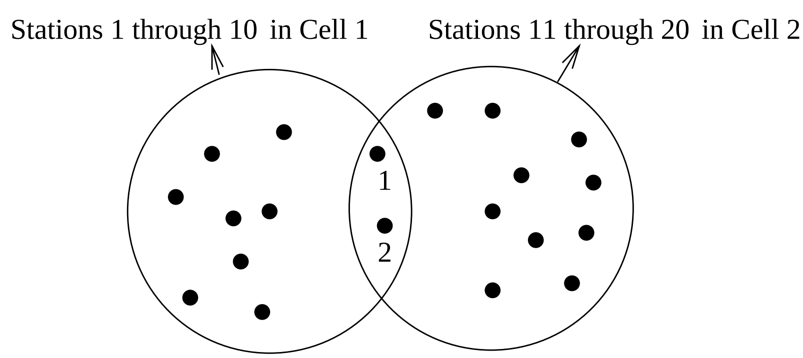

Figure 5.6: Configuration of Two Cells

In Section 4.3, we proposed the technique – bounded contention – to solve the starvation problem. It requires setting a threshold t in the base station. t must be chosen properly. Too many slots will be wasted if t is too large when interference happens. On the other hand, if t is too small, a contention period may be terminated prematurely, wasting the contention slots already incurred. To determine t, N_w, the number of wasted slots in the bounded contention phase, is evaluated as follows:

$$N_w = t \times freq \tag{5.2}$$

where $freq$ records the frequency that bounded contention is applied. Figure 5.7 plots N_w versus t in the two-cell configuration as shown in Figure 5.6. It shows that when t is around 16, N_w is the smallest for both cells. Thus, t is set to 16 in our simulations. Simulations using other multi-cell configurations also show that $t = 16$ is a good choice.

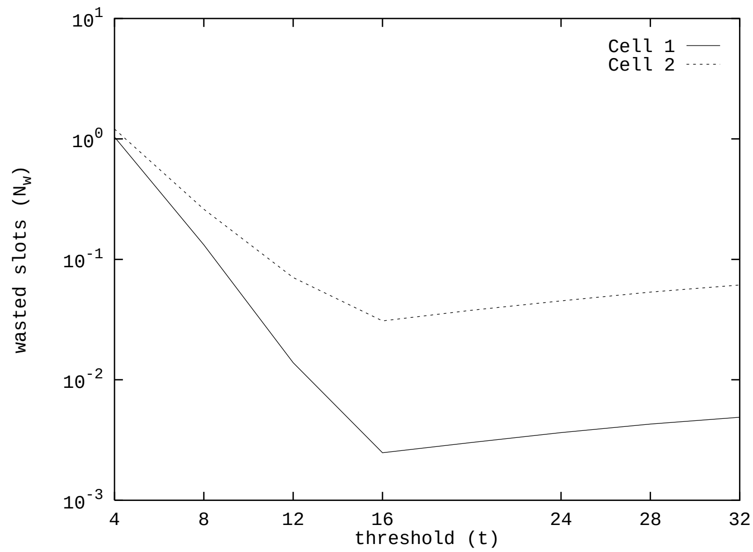

Figure 5.7: Wasted Number of Contention Slots in Bounded Contention Phase

Table 5.1 shows the number of contention slots of the WWP with bounded contention. In fact, Two cells yield similar throughput, which proves that the bounded contention technique has completely solved the starvation problem. We can write the number of contention slots, N, as composed of two parts:

$$N = N_{org} \times (1 - freq) + t \times freq \tag{5.3}$$

where N_{org} is the number of contention slots needed without the applications of bounded contention. In our simulations, $freq$ is around 0.0025 for Cell 1 and 0.031 for Cell 2, hence, only 0.04 and 0.5 slots are paid to solve the starvation problem.

Figure 5.8 plots the delay distributions of the twenty stations in the two cells using the WWP with bounded contention. Stations 3 through 10 in Cell 1 have similar delay

Table 5.1: Number of Contention Slots for Two Cells

	WWP with bounded contention phase
cell 1	2.50
cell 2	2.62

distributions. Likewise, Stations 11 through 20 in Cell 2 also have similar delay distributions. The average delays and delay deviations are slightly larger for stations in Cell 2 because of interference from Stations 1 and 2 of Cell 1. For Stations 1 and 2, delay times are larger than other stations in Cell 1 since they are in the range of both bases. However, after they completely migrate into Cell 2, they can share the channel equally with other stations in Cell 2.

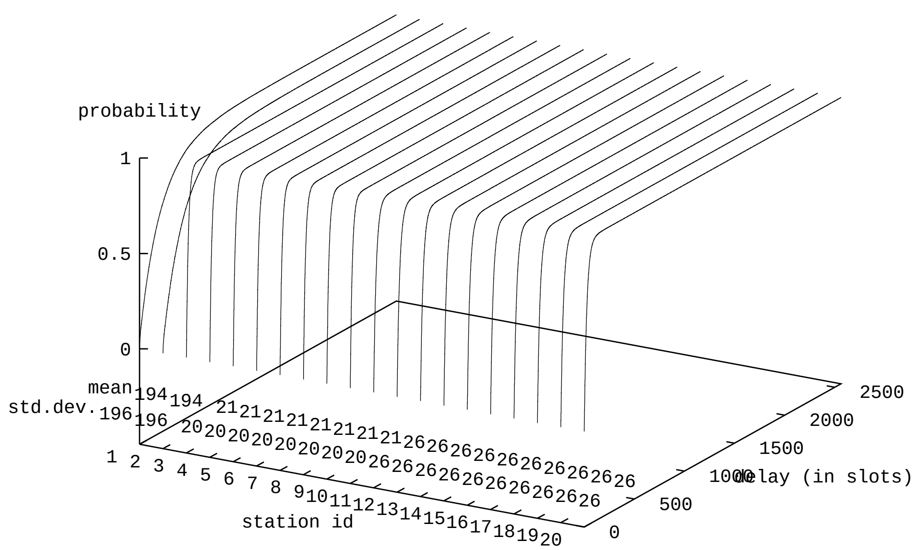

Figure 5.8: Delay Distributions in Two Cells

5.4 Lookahead WWP

Lookahead WWP aims at reducing the number of contention slots by making use of the time waiting for acknowledgements. Figure 5.9 compares the number of contention slots with and without lookahead. As expected, the number of contention slots required by lookahead WWP is between 1.2, i.e. half of that required by the original WWP, and 2.4. Therefore, it has an obvious performance gain over the original WWP implemented by frquency division multiplex.

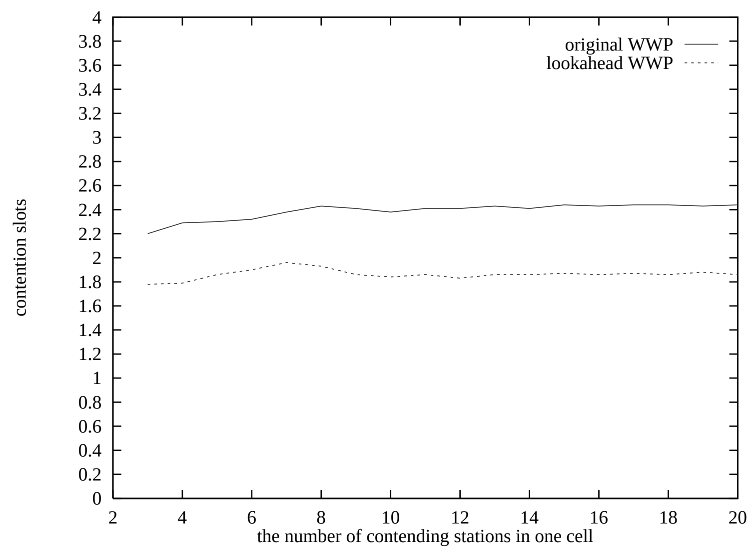

Figure 5.9: Number of Contention Slots for WWP and Lookahead WWP

Figure 5.10 demonstrates that fairness is maintained even with introduction of the lookahead technique.

5.5 Comparsion with DFWMAC in One Cell

In this section, we compare WWP with the RTS/CTS exchanges in DFWMAC. In our simulations of the DFWMAC, we set the parameters as follows: $DIFS = 32\mu s$, $SIFS = 4\mu s$, $CW_{min} = 2$ and $CW_{max} = 256$, where CW_{min} is the minimum contention window that stations start off with when they join the contention, and CW_{max} is the maximum contention window that stations can have. RTS and CTS last $120\mu s$(30 bytes in a 2Mb/s

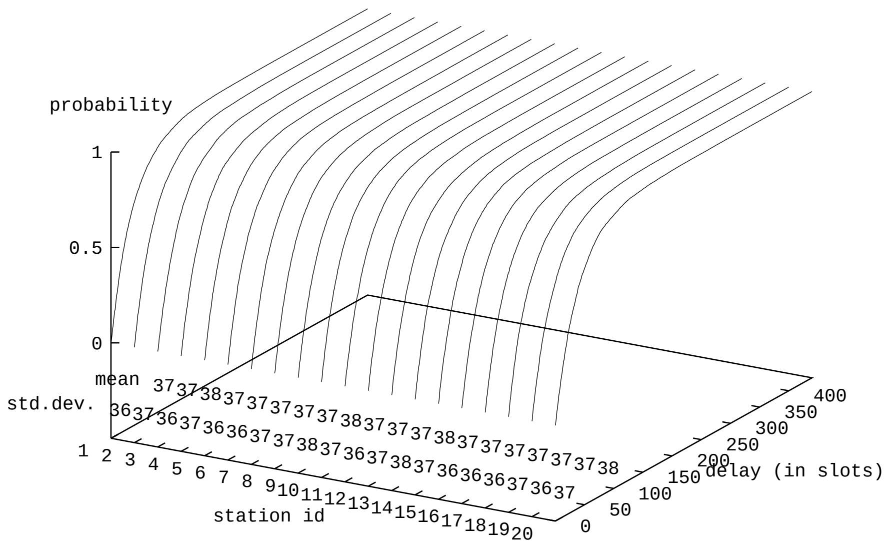

Figure 5.10: Delay Distributions in a 20-Station Cell with Lookahead Technique

network). We define one slot as the time needed for one successful RTS/CTS exchange without contention, i.e., $DIFS + RTS + SIFS + CTS = 276\mu s$. The simulations run until the 0.95 confidence interval is reached for each station. This guarantees that stable states are reached before statistics are taken.

Figure 5.11 compares the number of contention slots required by WWP and DFW-MAC. Although DFWMAC has a smaller number of contention slots when the number of contending stations is small, the increase in contention slots is faster than that of WWP so that when the number of contending stations is larger than 100, DFWMAC has larger numbers of contention slots, i.e, lower throughput. It shows that WWP is a scalable protocol and it has almost constant throughput regardless of the channel load.

The number of contention slots is an important metric but fairness is another important metric to consider since one can have zero contention slot, i.e. 100% throughput, by exclusively allocating the channel to one single station and turning down all the other stations. We observe that the lower throughput of WWP as compared to DFWMAC for small number of contending stations is compensated by a fairer allocation of the channel.

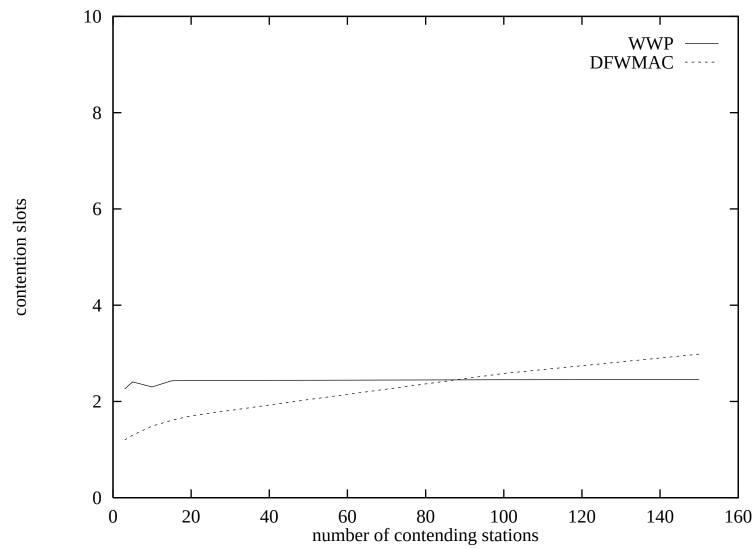

Figure 5.11: Number of Contention Slots of WWP and DFWMAC in One Cell

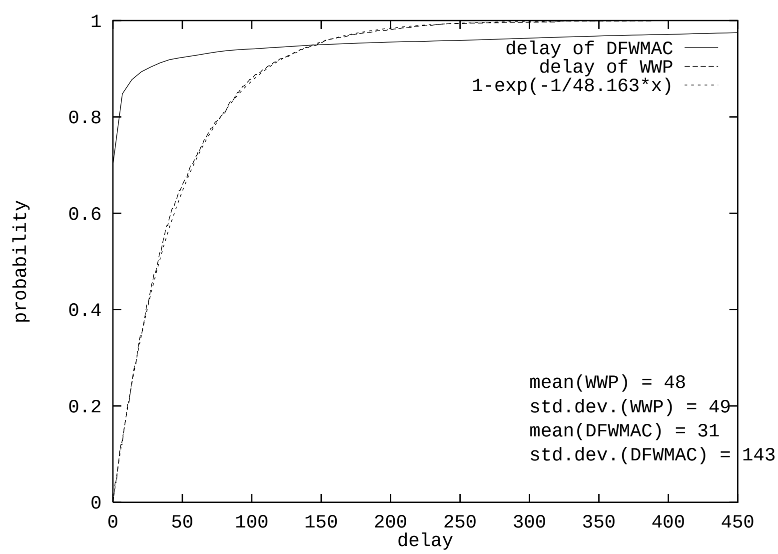

Figure 5.12: Delay Distributions of WWP and DFWMAC of a Single Station

Figure 5.12 compares the delay distributions of DFWMAC and WWP. It can be seen that DFWMAC has a skewed access pattern: over 85% of the accesses can be made within a few contention slots, while 5% of the accesses incur exceedingly long delays. As a result, it has far larger delay deviations than WWP(3 times). DFWMAC has large variances in access delays because of the backoff scheme. It can be illustrated by the following scenario. Suppose there are two stations in a cell sending packets to each other.

The one who has just succeeded in capturing the channel has a smaller backoff value, therefore, it has a larger chance to succeed in the next contention period. It is likely that one station dominates the channel for a number of consecutive contention periods before the other takes over, resulting in high variances in delays. In constrast, in WWP, stations have equal probabilities to succeed in every contention period. The fact that a station succeeded for the previous contention doesn't favor it for the current contention period. This behavior leads to an exponential distribution of access delays as shown in Figure 5.12.

To summarize, WWP provides both long-term and short-term fairness while DFW-MAC only renders long-term fairness. WWP is a fairer, more scalable protocol than DFWMAC, yet yielding comparable throughput.

5.6 Comparison with DFWMAC in Two Cells

In this section, we evaluate the performance of WWP and DFWMAC in the two-cell configuration shown in Figure 5.13, where Station 1 is in the range of five other stations, and the others can only hear stations in their own cell. In this scenario, Stations 4, 5 and 6 are hidden to 2 and 3. Moreover, Station 1, 4, 5 and 6 may become an exposed station if any of them is transmitting. As an example, when Station 4 is the sender and 5 is the receiver, 1 can't initiate transmission to 3 while 4 is transmitting. Hence, 1 is the exposed station in this case.

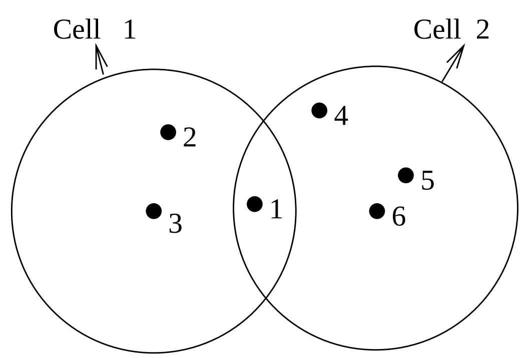

Figure 5.13: Example Illustrating a Two-Cell Configuration

Table 5.2 gives the number of contention slots of both cells in WWP and DFWMAC. In DFWMAC, Cell 1 has larger number of contention slots(lower throughput) than Cell 2 because it suffers more from the hidden terminal and exposed terminal problems. Suppose there are packet exchanges in Cell 2, 2/3 of the packet transmissions in Cell 1 are either corrupted or delayed by ongoing transmissions in Cell 2, namely, packets from 2 or 3 to 1 are corrupted and those from 1 to 2 or 3 are delayed. On the contrary, if there are packet exchanges in Cell 1, only 1/3 of the packets in Cell 2 are delayed, namely, it happens only when 1 is sending to either 2 or 3, and other cases have no effects on data transmissions in Cell 2. The above analysis is based on the assumption that packet destinations are randomly chosen. However, WWP is not affected by this configuration asymmetry, and two cells have comparable throughput.

Table 5.2: Number of Contention Slots of WWP and DFWMAC in Two Cells

	WWP	DFWMAC
cell 1	2.331	7.267
cell 2	2.577	1.523

Comparisons of average delays and delay deviations are shown in Figure 5.14 and Figure 5.15 , respectively. They are plotted in log-scale to amplify the differences among stations. In DFWMAC, Stations 2 and 3 have much larger average delays and delay deviations than Station 1. This can be explained by the following analysis: if there are transmissions in Cell 2, 1/2 of the packets generated by Station 2 or 3 are collided while none of the packets by Station 1 are collided. The collisions result in larger backoff windows in Stations 2 and 3. Therefore, Station 1 has more chances to succeed than Stations 2 and 3. The fact that Station 1 is in the overlap area causes large performance degradation of other stations in Cell 1. On the contrary, in WWP, only Station 1 has larger average delay and delay deviation than other stations, and the rest stations have similar delay characteristics.

In summary, WWP is less sensitive to hidden and exposed terminal problems than DFWMAC.

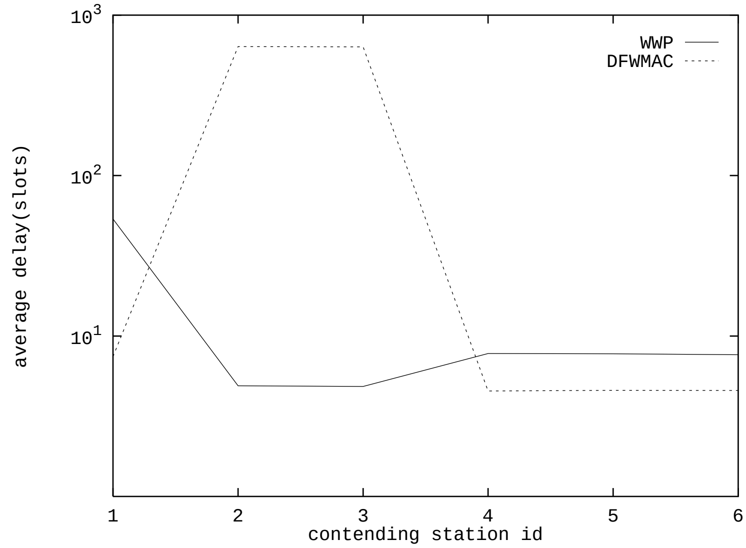

Figure 5.14: Delay Averages of WWP and DFWMAC in Two Cells

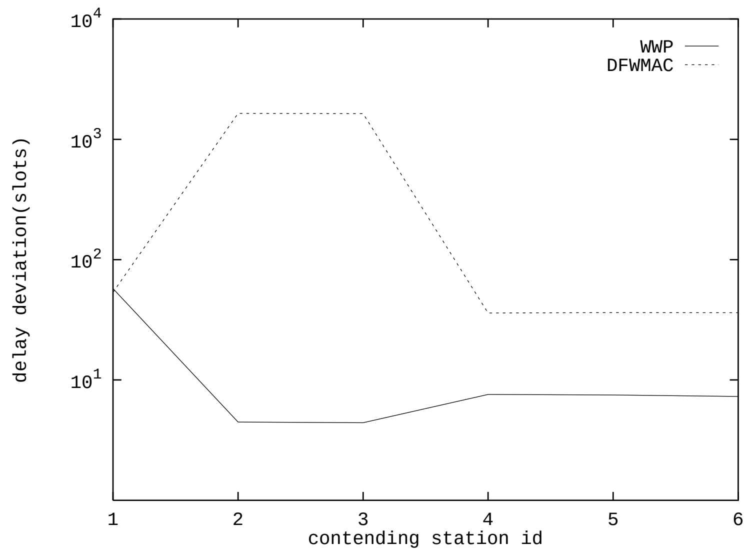

Figure 5.15: Delay Deviations of WWP and DFWMAC in Two Cells

Chapter 6

Conclusion

In previous two chapters, we described the design and performance evaluation of WWP in the one-cell and multiple-cell scenarios. WWP is a limited contention protocol, and the choice of contending stations is based on channel load. Specifically, contending stations are chosen based on a global common window whose bounds are functions of channel load estimate.

Some protocols sacrifice fairness for high throughput. However, WWP provides both long-term and short-term fairness while maximizing the throughput. Simulations show that it achieves the four goals stated in Section 1.2:

- **throughput** – It is closely related to the number of contention slots. In WWP, windows are chosen based on the objective to minimize the future number of contention slots. It turns out WWP has comparable throughput as DFWMAC, the IEEE 802.11 draft standard.

- **delay** – Delay variances for WWP are very small as compared to those of DFW-MAC. This is very desirable for applications with QoS requirements.

- **fairness** – Long-term and short-term fairness are reflected in average delays and delay variances of stations. WWP provides both long-term and short-term fairness.

- **scalability** – WWP has load-independent characteristics. The throughput remains constant regardless of the number of contending stations. This feature makes WWP a highly scalable protocol.

Future work includes building a prototype of WWP in which stations communicate with wireless modems that can detect three possible states of a channel: no transmission, success and collision. WWP is to be implemented in kernel determining who is transmitting according to the feedback by the modem. The output module of the IP layer needs to be modified in order to direct outgoing packets to the WWP output queue.

We also plan to evaluate higher level protocol performance using WWP as the MAC protocol. It is possible to reduce handoff and transport layer jitters because of the small delay variances introduced by WWP.

Bibliography

[1] A.Bakre and B.R.Badrinath. I-TCP: Indirect TCP for mobile hosts. Technical Report DCS-TR-314, Rutgers Univ., October 1994.

[2] A.Bakre and B.R.Badrinath. Handoff and system support for indirect TCP/IP. In *Second Usenix Symp. on Mobile and Location-Independent Computing*, April 1995.

[3] B.W.Wah and J.Y.Yuang. An efficient contention resolution protocol for local multiaccess networks. U.S. Patent 4630264, Filed Sept. 21, 1984, Granted Dec. 16, 1986.

[4] B.W.Wah and J.Y.Yuang. An efficient protocol for load balancing on CSMA/CD networks. In *Proc. 8th Conf. Local Comput. Networks*, pages 55–61, October 1983.

[5] B.W.Wah and J.Y.Yuang. Resource scheduling for local computer systems with a multiaccess network. *IEEE Trans. on Computers*, C-34(12):1144–1157, December 1985.

[6] C.Ma. On building very large naming systems. In *5th SIGOPS Workshop on Models and Paradigms for Distributed Systems Structuring*, September 1992.

[7] D.Allen. *Hidden Terminal Problems in wireless LANs.* IEEE 802.11 Working Group paper 802.11/93.

[8] D.G.Jeong, C.H.Choi, and W.S.Jeon. Design and performance evaluation of a new medium access control protocol for local wireless data communications. *IEEE/ACM Trans. on Networking*, (6), December 1995.

[9] D.J.Goodman, R.A.Valenzuela, K.T.Gayliard, and B.Ramamurthi. Packet reservation multiple access for local wireless communication. (8), August 1989.

[10] E.Amir, H.Balakrishnan, S.Seshan, and R.H.Katz. Efficient TCP over networks with wireless links. In *Fifth IEEE Workshops of Hot Topics in Operating Systems*, May 1995.

[11] F.A.Tobagi and L.Kleinrock. Packet switching in radio channels: Part ii - the hidden terminal problem in carrier sense multiple-access and the busy-tone solution. *IEEE Trans. on Communications*, com-23(12):1417–1433, December 1995.

[12] F.Teraoka and M.Tokoro. Host migration transparency in IP networks: the VIP approach. *Computer Communication Review*, (1):45–65, January 1993.

[13] G.H.Forman and J.Zahorjan. The challenges of mobile computing. Technical Report 93-11-03, Univ. of Washington, 11 1993.

[14] J. Ioannisdis, D.Duchamp, and G.Q.Maguire. IP-based protocols for mobile internetworking. In *SIGCOMM'91 Symposium*, pages 235–245, September 1991.

[15] J.Warand. *Communication Networks: a First Course*. Richard D. Irwin and Aksen Associates, 1991.

[16] J.Weinmiller, H.Woesner, J.P.Ebert, and A.Wolisz. Analyzing and tuning the distributed coordination function in the ieee 802.11 DFWMAC draft standard. In *Int. Workshop on Modelling, Analysis and Simulation of Computer and Telecommunication Systems*, February 1996.

[17] K.Biba. *A hybrid wireless MAC protocol supporting asynchronous MSDU delivery services*. IEEE 802.11 Working Group paper 802.11/91-92, Sept, 1992.

[18] K.C.Chen. Medium access control of wireless LANs for mobile computing. *IEEE Network*, September 1994.

[19] K.C.Chen and C.H.Lee. RAP - a novel medium access control protocol for wireless data networks. In *IEEE GLOBECOM*, December 1993.

[20] Mesquite Software Inc. *CSIM17 User Guide - C Version.* available at http://www.mesquite.com/.

[21] P.Karn. MACA-a new channel access method for packet radio. In *ARRL/CRRL Amateur radio 9th computer networking conference*, September 1990.

[22] V.Bharghavan. *Access, Addressing and Security in Wireless Packet Networks.* PhD thesis, Univ. of California at Berkeley, 1995.

[23] V.Bharghavan, A.Demers, S.Shenker, and L.Zhang. MACAW : A media access protocol for wireless LANs. In *ACM SIGCOMM Conference on Communications Architectures, Protocols and Architectures*, September 1994.