INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

**Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600**

UMI®

MULTIPLE ACCESS SCHEMES FOR THE
MICROWAVE LIGHT ORGANIZED NETWORK
(MILTON)

By
Mathys Walma

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING
AT
OTTAWA-CARLETON INSTITUTE FOR ELECTRICAL AND COMPUTER ENGINEERING
SYSTEMS AND COMPUTER ENGINEERING
CARLETON UNIVERSITY
OTTAWA, ONTARIO
APRIL, 2000

© Copyright by Mathys Walma, 2000

**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

395 Wellington Street
Ottawa ON K1A 0N4
Canada

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-52395-0

Canada

The undersigned hereby recommend to the Faculty of Graduate
Studies and Research acceptance of the thesis

Multiple Access Schemes for the Microwave Light
Organized Network (MILTON)

submitted by **Mathys Walma** in partial fulfillment of the requirements
for the degree of Master of Engineering

Dated: April, 2000

Thesis Supervisor:

_____
Prof. David D. Falconer

Chair:

_____
Dept. of Systems and Computer Engineering

Abstract

The MILTON system is targeted to provide residential homes with high-speed wireless Internet access. Each base-station is divided into 32 sectors, every fourth sector reusing the same frequency bands. Downlink and uplink use separate frequency bands. All subscribers in a sector must share the same uplink frequency band(s).

The MAC proposed in this thesis attempts to allow the subscribers to arbitrate access to the common uplink channel in a fair, efficient, and simple manner. No attempt is made to provide different quality of service levels.

Traffic in a real LAN environment has been captured and used in simulations.

The collision avoidance scheme dynamically changes the transmit probability based on recent offered load, thereby reducing the number of collisions. This is seen to work better than exponential back off and about the same as spanning tree.

Overall, the MILTON MAC satisfies the original requirement of a fair, efficient, and simple protocol.

Acknowledgements

Thanks to:

- Mom and Dad for their moral support.
- John Sydor for providing a supportive working environment.
- Professor Falconer for providing me with contacts, and helpful advice.
- NSERC for funding me during my first year at Carleton.

Table of Contents

Abstract	iv
Acknowledgements	v
Table of Contents	vii
List of Tables	viii
List of Figures	ix
1 Introduction	1
1.1 Uplink of MILTON	3
1.2 Media Access Control (MAC) protocol	4
1.3 Thesis Layout	6
1.4 Thesis Contributions	7
1.5 Conclusion	8
2 Wireless MAC layers as they exist today	9
2.1 Aloha MAC protocols	9
2.2 Packet Reservation Multiple Access (PRMA) MAC protocols	10
2.3 Inhibit Sense Multiple Access (ISMA)	11
2.4 Code Division Multiple Access (CDMA) versus Time Division Multiple Access (TDMA)	13
2.5 Collision Avoidance (CA)	14
2.5.1 Collision Avoidance - maximizing the probability of a single transmission	15
2.5.2 Tree algorithm	16
2.6 Conclusion	17

3	Problem Statement	19
3.1	Why a new MAC protocol?	19
3.1.1	Traffic characteristics	19
3.1.2	Multiple uplinks in one petal	20
3.1.3	Collision Avoidance (CA)	20
3.1.4	Common downlink for more than one petal	21
3.1.5	Interference avoidance	21
3.1.6	Flexibility	21
3.2	Conclusion	22
4	Performance Comparisons of Potential MAC Protocols	23
4.1	Traffic characteristics	23
4.1.1	WWW Traffic	24
4.1.2	Voice over IP (VOIP) Traffic	28
4.2	Effect of multiple uplinks	31
4.3	Collision Avoidance - the first approach	34
4.3.1	Comparing known distributions	35
4.3.2	Finding a better distribution	40
4.4	Collision Avoidance - Dynamic P-Persistence (DPP)	43
4.5	Collision Avoidance - Spanning Tree Random Backoff	46
4.6	Comparing DPP to Spanning Tree backoff using the WWW traffic model	54
4.6.1	Reservation - Is It Needed?	55
4.7	Comparing DPP to Spanning Tree backoff using the Voice over IP model	56
4.8	Collision Detection	57
4.9	MILTON MAC protocol	58
4.9.1	Change made to DPP method	59
4.9.2	MILTON MAC messaging downlink	61
4.9.3	MAC messaging uplink	65
4.9.4	MILTON MAC algorithm in Base Station	68
4.9.5	MILTON MAC algorithm at Subscriber	70
4.9.6	MILTON MAC simulation results	71
5	Conclusion	80
	Bibliography	82

List of Tables

4.1	Number of slots needed to resolve contention of 500 subscribers . . .	45
4.2	Average delay in seconds for DPP and Spanning Tree collision avoid- ance algorithms using WWW traffic model	54
4.3	Percentage of time channel is busy with data using WWW traffic model	54
4.4	Percentage of time channel is in collision state with WWW traffic model	55
4.5	Average delay in seconds for DPP and Spanning Tree collision avoid- ance algorithms using VOIP traffic model	56
4.6	Percentage of time channel is busy with VOIP traffic model	57
4.7	Percentage of time channel is in collision state with VOIP traffic model	57
4.8	Opcodes for Downlink	62
4.9	Opcodes for Uplink	65
4.10	Difference in delay for 100 and 1000 seconds of simulated time	72

List of Figures

1.1	MILTON Cell layout	3
1.2	MILTON frequency reuse	4
2.1	Hidden Station Problem	11
2.2	Uniform distribution from 0 to 5	15
2.3	Negative exponential distribution $y = x \cdot \exp(-x)$	16
2.4	Binomial Distribution	17
4.1	Uplink WWW packet size PDF	25
4.2	Downlink WWW packet size PDF	26
4.3	Downlink data rate vs. time	26
4.4	Uplink data rate vs. time	27
4.5	Downlink WWW packet size PDF for VOIP	29
4.6	Uplink WWW packet size PDF for VOIP	29
4.7	Downlink data rate vs. time for VOIP	30
4.8	Uplink data rate vs. time for VOIP	30
4.9	S vs G for different number of uplink channels	31
4.10	Average Packet Delay vs. Offered Load	32
4.11	Packet Delay Variance vs. Offered Load	33
4.12	Modified ISMA Collision Avoidance	36
4.13	Modified ISMA Collision Avoidance with a Limited Transmit time (δ)	37
4.14	Bisection Algorithm	38
4.15	Uniform Collision Avoidance	38

4.16 Negative Exponential Collision Avoidance	39
4.17 Comparing Negative Exponential with Uniform	39
4.18 Optimal CDF Search	40
4.19 Average Delay for 5 Subscribers vs. Optimization Time	41
4.20 Best Average Packet Delay vs. Number of Subscribers	42
4.21 Dynamic Collision Avoidance	44
4.22 Sequence of events in Spanning Tree Algorithm	49
4.23 Spanning Tree Graphical Representation	50
4.24 Number of Subscribers vs. Utilization using WWW traffic model for Spanning Tree	51
4.25 Number of Subscribers vs. Average packet delay using WWW traffic model for Spanning Tree	52
4.26 Number of Subscribers vs. Packet delay variance using WWW traffic model for Spanning Tree	53
4.27 Format of a DVB frame	61
4.28 Example of MILTON MAC protocol	73
4.29 Comparing 1, 2, and 10 uplinks mean delay for VOIP traffic model .	74
4.30 Comparing 1 uplink and 2 uplinks variance of delay for VOIP traffic model	75
4.31 Comparing 1, 2, and 10 uplinks throughput for VOIP traffic model .	76
4.32 Comparing Exponential Backoff vs. Dynamic P-Persistence Average Delay	77
4.33 Comparing Exponential Backoff vs. Dynamic P-Persistence Variance of Delay	78
4.34 Comparing Exponential Backoff vs. Dynamic P-Persistence Throughput	79
4.35 Collision vs. Success ratio	79

Chapter 1

Introduction

The objective of the MILTON project is to develop new wireless techniques and technologies capable of providing data delivery densities of 100 to 1000 Megabits per second per square kilometer of urban area. This technology will be highly bi-directional supporting high-speed data exchanges between subscribers and providing multi-media interactivity between users. As designed, the system will distribute high-quality video and audio information as well as standard internet-type data.

The system uses very low power signals to relay data between a base-station antenna and urban users located within a 1-2 kilometers radius of the antenna (for rural the distance can be up to 5-10 km radius). Different deployment techniques are used for urban and rural environments. In an urban environment, there is the expectation of obstruction by foliage and buildings, hence the operational radius is smaller. In a rural environment, less obstruction is expected, thus resulting in longer communications distances. For rural applications it is envisaged that a single base-station will provide the data requirements of a sparse, widely distributed population (within 10 km of the base-station). In an urban environment, numerous base-stations would be required to meet the requirements of a higher population density.

The base-station consists of an integrated antenna array, digital radio, and digital switch mounted on a power pole, lamp standard, or medium height (5 story) building. The coverage pattern of the base-station antenna is best visualized as a rosette of antenna beams (see Figure 1.2), each beam being an oblong radio micro-cell. Typically a rosette contains 32-48 micro-cells. This configuration of cells allows a very high data carrying capacity. Each fully loaded MILTON rosette is capable of receiving, transmitting, and switching in the order of 1000-1500 Megabits of information per second. MILTON rosettes can work individually, supporting the above capacities within their coverage areas, or they can be interconnected by optical fiber (light) to form a large metropolitan high speed data network. Figure 1.1 shows the layout of neighboring rosettes with the eight common frequency petals for each rosette being shaded. The common frequency petals for neighboring rosettes are aligned with the objective to reduce interference. CRC¹ is in the process of applying for a US patent covering this concept.

The second component of the system is the subscriber terminal. The device mounts on the roof of the subscribers' buildings (wall or roof) and is oriented toward the rosette base-station. Perfect visibility of the base-station is not necessary and clutter by trees and even partial obstructions by one or two adjacent buildings can be tolerated. CRC has conducted extensive propagation measurements using proposed MILTON frequencies, and is developing urban installation models that will ensure high availability of the system to users.

---

¹Communications Research Centre in Ottawa, Canada

Figure 1.2: MILTON frequency reuse

The frequencies used are in the 5 GHz range. This frequency range has desirable propagation characteristics in a residential environment. As well, the demodulation hardware and other components operating at these frequencies are low cost due to their use in satellite TVRO and PCS applications. A contiguous band of 100 MHz will be used for the uplink, or 25 MHz for each petal.

1.2 Media Access Control (MAC) protocol

The uplink is shared among a number of subscribers. Consequently an efficient Media Access Control (MAC) protocol needs to be developed. Implementation complexity is a consideration. For a quantitative measure of efficiency, two parameters are normally looked at, throughput and delay.

Throughput is defined as the actual data rate being sent through the system. Offered load is normally a larger value and is the data rate that would be sent if the subscribers had unrestricted access to the channel. Normally each of these values is

represented as a fraction of raw channel capacity. Ideally throughput matches offered load as close as possible.

Delay measures the time it takes a packet to travel from subscriber to basestation. Ideally the mean and variance are kept low.

A few MAC protocols have been standardized:

1. DOCSIS [3]. IEEE 802.14, a cable modem standard. currently under review for adaptation to a new LMCS/LMDS wireless standard IEEE 802.16 [10]
2. DAVIC. designed for use in LMCS/LMDS [2]
3. HIPERLAN. an ETSI standard for a high performance radio LAN [6]
4. IEEE 802.11, in-door wireless LAN standard [12]

While the wireless LAN standards aren't directly applicable to MILTON, they use methods for collision detection and avoidance that are.

The DOCSIS standard is the most promising of all, since the architecture of a hybrid fiber/coax (HFC) system closely resembles that of the MILTON system. There are aspects to DOCSIS that would make it difficult to use in an interference limited system such as MILTON.

One difference concerns the Forward Error Correction (FEC) used in the standard. It isn't robust enough to give an acceptable Bit Error Rate (BER) at a reasonable signal to noise ratio (SNR) over a wireless link. HFC requires less coding since it's mean operational $\frac{S}{N+I}$ is considerably higher than in a wireless MILTON link. In MILTON the nominal signal to noise plus interference ratio, $\frac{S}{N+I}$, is 15dB, while in DOCSIS it is 25dB.

Other differences between MILTON and DOCSIS are discussed in Section 3.1.

In all these schemes there is the possibility of two subscribers transmitting at the same time. This results in a collision, which is undesirable because if the received power at the basestation for both is the same, then the transmission is garbled. In ethernet collision is avoided by detecting the transmission of others, this can be done even while the subscriber itself is transmitting. In the wireless environment it isn't possible for subscribers to directly detect the transmission of each other. This is because all the subscribers can see the basestation, but not necessarily each other. Even if the subscribers can see each others transmission, the difference in the power level of the subscribers own signal, and that of the colliding subscriber, make it very difficult to detect. For these reasons, the subscribers are forced to rely on feedback from the basestation for status of the uplink channel.

Usually the basestation detects collisions by a sudden increase in bit error rate (BER), as detected by Forward Error Correction (FEC) codes, or possibly a special synchronization sequence. However, sometimes the basestation itself can't detect the presence of a collision due to the capture effect. Of the colliding subscribers, one is received at the basestation with much more power than the others, greater than the minimum $\frac{S}{N+I}$ for the modulation/coding method used, and is consequently 'captured' and detected without error.

1.3 Thesis Layout

The thesis is organized as follows.

Chapter 2 provides background into the current state of wireless MAC protocols.

Chapter 3 states the problem that the thesis aims to solve.

Chapter 4 shows experimental results from simulations with actual traffic, and compares the relative benefits of various collision resolution schemes, as well as their implementation complexity. Various network topologies are also compared. At the end of the chapter a specific MAC protocol is proposed for the MILTON system.

Chapter 5 gives conclusions that are reached from the experiments done and decisions made in Chapter 4.

1.4 Thesis Contributions

The work in this thesis differs from existing work in that the application of the multiple access techniques are for the MILTON system which has different characteristics from other systems studied so far. These differences are documented in Section 3.1.

In addition, the work differs from previous work, in that real traffic statistics were measured and used in the simulations. This provides a more accurate reflection of how the MAC will operate in the final system.

Collision Avoidance algorithms attempt to minimize the number of collisions. A number of different algorithms are compared based on efficiency and complexity. The final algorithm chosen is based on p -persistence in which the value for p changes according to offered load.

The vast majority of work into wireless MAC protocols attempt to provide quality of service levels to different types of traffic so that ATM can be used. The MILTON system is targeted for IP, which in its most common implementation does not have quality of service levels. Hence the MILTON MAC protocol doesn't attempt to provide different quality of service levels.

The details of the protocol are provided along with the messages defined in the

uplink and downlink in Section 4.9 on page 58. An example is given of the sequence of events that occur at the subscriber and basestation when a subscriber receives a packet to transmit in the uplink.

1.5 Conclusion

This paper will propose a MAC protocol for the MILTON system, taking into account its targeted audience, efficiency, and simplicity of implementation.

Chapter 2

Wireless MAC layers as they exist today

Wireless communications is attractive for a number of reasons. For information providers infrastructure costs are kept low. For subscribers wireless offers mobility.

The amount of available spectrum in the past has been limited, however, recently the regulating agencies have allocated large bands of spectrum for both licensed and unlicensed use.

2.1 Aloha MAC protocols

The pioneering research in wireless MAC protocols was performed in the 1970's at the University of Hawaii[9]. The intent was to create wireless links between the campuses on different islands.

The first protocol, Pure Aloha, makes no attempt to avoid collision. Subscribers transmit whenever they have a packet. It has a theoretical maximum throughput of around 18% assuming an infinite number of subscribers, exponentially distributed random packet sources, and equal sized packets. The contention interval is 2τ , where τ is the time it takes to transmit a packet. This is because any transmission can

collide with an earlier, and also a later packet.

Slotted Aloha improves on Aloha by having all subscribers synchronizing their transmissions to slots that are the width of a packet. This reduces the probability of collision since the collision interval is now limited only to τ . Making the same assumptions as Pure Aloha, the maximum throughput of Slotted Aloha is 36%.

2.2 Packet Reservation Multiple Access (PRMA) MAC protocols

More recently, a number of schemes have been proposed. Most are variations on Packet Reservation Multiple Access (PRMA)[7]. PRMA increases the throughput of a channel at the expense of delay. Quite often these systems attempt to provide ATM like quality of service classes.

The general idea with PRMA is that if any subscriber wants to transmit, it must be granted permission by the basestation. This can be done in two ways. Either the basestation can periodically poll the subscribers, or the subscribers can go through a REQ/ACK cycle when they have data to transmit.

The REQ/ACK cycle is the method most used in practice. If an ACK isn't received from the basestation within a certain period, it is assumed that the REQ collided with another subscriber.

Once a subscriber has been granted permission by the basestation, it usually has dedicated bandwidth available to it in the form of a slot that occurs periodically. This acts to reduce contention. When the subscriber is out of data to transmit, it gives up the slot by not transmitting.

Since the slots are of fixed size, packets at the subscriber are fragmented into

slot sized pieces and are then reassembled at the basestation back into packets. This requires that the basestation have a buffer for every subscriber. This increases system cost.

The maximum throughput of PRMA reaches a higher value than Slotted Aloha. However, a packet on average sees a higher delay.

2.3 Inhibit Sense Multiple Access (ISMA)

In [9] a Collision Sense Multiple Access (CSMA) scheme with busy tone was proposed for radio channels. CSMA requires that every station be able to detect transmissions from all other stations at all times. If a transmission is detected, then all others back off. CSMA can't be used in a radio channel due to the problem of the hidden station.

Figure 2.1: Hidden Station Problem

The hidden station problem can be described as follows, see Figure 2.1¹. Say there are three stations, *A*, *B*, and *C*. Both *B* and *C* need to transmit to *A*. However, *B* and *C* are out of range of each other, so they can't detect each other's transmissions, and hence their transmissions to *A* will possibly collide. The solution is to have *A* transmit a busy tone when it is receiving a transmission, telling either *B* or *C* that a transmission is in progress. This will reduce the probability of collision significantly.

---

¹the circles represent the area in which transmissions by that station can be received

The period of possible collision, α , is two propagation delays plus some processing time. This entire scheme has since been called Inhibit Sense Multiple Access (ISMA) [11].

Since the feedback is binary, either busy or idle, subscribers can still collide for the duration of one packet if they both start transmitting within α seconds of each other.

Numerous variations are possible. The use of slots acts to reduce the collision interval α . This requires the subscribers to time their transmissions to arrive at the basestation (in this case A) at the beginning of a slot.

Unlike the previous PRMA algorithm, a subscriber is given access to the channel for only one transmission. Future transmissions require going through the contention phase again. Also, in ISMA the subscribers can transmit the entire contents of their transmit queue once they have channel access, whereas in PRMA they have to fragment their queue into pieces that fit into one slot, the pieces are then reassembled at the basestation. This makes implementing ISMA easier hardware wise, although it is not very conducive to constant bit rate sources. This isn't a problem since there aren't any constant bit rate sources in the MILTON system.

The busy tone can be updated in two ways [1], fast and slow. In fast the busy tone is updated before the end of the slot, in slow the busy tone is updated at the end of each slot.

2.4 Code Division Multiple Access (CDMA) versus Time Division Multiple Access (TDMA)

CDMA spreads the transmitted signal over a frequency range much larger than the baseband information channel bandwidth. This makes it much more resistant to interference. The degree of spreading is called the processing gain or spreading factor. A factor of two means that the transmitted signal takes twice the spectrum as its baseband information channel bandwidth. Each subscriber uses a different spreading code. In the case of a spreading factor of 2, the spreading code would be a two bit sequence that is transmitted in the place of each original bit, at twice the old bit rate. If each subscriber uses a different orthogonal spreading code, then at the receiver it is possible to isolate each subscriber's signal from the others. In general, the number of simultaneous transmissions supported is on the same order as the spreading factor.

In a multiple access scheme this is attractive because each subscriber can then transmit independently from each other and not fear collision.

In order to support 100 users in a petal of the MILTON system, a spreading factor of around 100 would be needed. Since there is only about 25 MHz of bandwidth available for each petal in the uplink, this means each code would be allowed to transmit at 250 kbps. This rate is insufficient to support the kind of services that the MILTON system is targeted for.

To increase the rate, the spreading factor can be reduced, decreasing the number of users supported in a petal, but also increasing the base rate each subscriber is able to transmit at. If 10 users are allowed in each petal, the spreading factor becomes 10. Each subscribers transmission rate then becomes 2.5 Mbps. This is getting closer to the rate that would be acceptable for the MILTON system, however the number of

supported users is now too small.

Each of these 2.5 Mbps links would be dedicated to that specific user. Because of this limitation, CDMA systems have a lower utilization efficiency than TDMA systems that use the same bandwidth. This is corroborated in [13] where it is shown that CDMA schemes with moderate spreading factor have a lower utilization efficiency than TDMA schemes. This then eliminates CDMA as a candidate for the MILTON system.

2.5 Collision Avoidance (CA)

In all the methods, collisions still do occur. When a collision occurs, there are a number of methods that are used to resolve the contention.

The simplest method is to have each subscriber back off for a random period of time before retransmitting. This random period could be generated using one of the many different known random distributions. The random distribution used in practice is usually uniform, negative exponential, or binomial. These random distributions are shown in Figures 2.2, 2.3, 2.4. However, any of these become unstable if the number of users increases beyond a certain point. An unstable positive feedback cycle is created when the probability of collision exceeds some threshold value. The backlog of stations wishing to transmit then increases even more with the resulting probability of collision then becoming even greater.

To counter this instability, another scheme is used in which the average of the random distribution increases with each collision, usually it is doubled. This acts to reduce the probability of collision, and hence bring the system back to a stable operating region. The name given to this modification is exponential backoff. This

Figure 2.2: Uniform distribution from 0 to 5

method is used in a lot of the current MAC protocols, including DOCSIS [3] , and 802.11 [12].

The attraction to this method is the distributed nature of the scheduling algorithm, each subscriber makes its own decisions with minimal feedback from the basestation. It is resistant to errors in the communications channel from the basestation because no feedback is required. As well, it is simple to implement.

2.5.1 Collision Avoidance - maximizing the probability of a single transmission

Given that N subscribers have packets to transmit, what is the optimal transmit probability for each contending subscriber that will maximize the probability that only one will transmit? This is known as the Control Tower problem in [5]. It turns

Figure 2.3: Negative exponential distribution $y = x \cdot \exp(-x)$

out from simple calculus that the optimal transmit probability is $1/N$.

2.5.2 Tree algorithm

Spanning Tree Random Backoff works on the principle of divide and conquer. Each time a collision occurs, the contending subscribers randomly assign themselves to one of a number groups, the number of groups is known as the splitting parameter Q . Each group then becomes isolated from the rest. As contention continues the groups further subdivide themselves until only one subscriber is left in a particular group, at which point contention is resolved for that subscriber. [8]

Figure 2.4: Binomial Distribution

2.6 Conclusion

The ideal MAC protocol would have zero delay for all offered load levels up to the link capacity. At the same time throughput would be equal to all offered loads, again, up to the link capacity. This is theoretically impossible, even with an optimum scheduler. The best possible is to have offered load equal to throughput up to the channel capacity, but the average delay increasing exponentially as the link capacity is reached. The fact that the delay increases exponentially is intuitively clear since an offered load slightly greater than the link capacity represents an unbounded average packet delay, therefore the link capacity represents an asymptote for average delay.

The protocols described attempt to provide these ideals. There will always be a

protocol overhead. The overhead is usually expressed in a higher delay, or lower maximum throughput. It is observed that it is often possible to trade delay performance for throughput performance.

Chapter 3

Problem Statement

The MILTON system requires a MAC protocol that supports Internet traffic efficiently under varying loads and number of subscribers. As well, implementation complexity and cost need to be kept down.

As mentioned earlier, efficiency is defined as a combination of minimizing delay and maximizing throughput.

3.1 Why a new MAC protocol?

There are a number of MAC protocols that currently exist, however, the MILTON system has unique characteristics that necessitate a customized protocol.

3.1.1 Traffic characteristics

The first unique characteristic consists of the fact that the traffic will be Internet generated. This traffic has an inherent asymmetry between the uplink and the downlink. The number of packets on the uplink and downlink are approximately the same, however the downlink packets are usually several times larger.

Most of the existing MAC protocols are designed for generic traffic types. It is

thought that designing a MAC protocol specifically with Internet traffic in mind will create better performance.

As well, the traffic is bursty, making certain MAC protocols less ideal than others. In the authors experience from designing a wireless ethernet bridge, delay plays a crucial role in the overall throughput of a single TCP/IP socket connection. For example, on an 8Mbps link, throughputs of an FTP file transfer went from under 1Mbps with a ping¹ value of 40ms to around 3Mbps with ping values of under 10ms.

To make the simulations more realistic, packet traces of typical Internet sessions have been captured and are used in the simulations of different MAC protocols.

3.1.2 Multiple uplinks in one petal

Secondly, there is the possibility of multiple uplink channels. Simulations have been done with the same aggregate uplink bandwidth, but with varying numbers of uplink channels, to see how performance is affected.

3.1.3 Collision Avoidance (CA)

In the event of collision, each subscriber backs off for a random time to avoid future collisions. Different probability distributions are looked at that minimize packet delay for fixed offered loads.

As well, a method in which the transmit probability is changed dynamically based on prior statistics of collisions and successful transmissions. This is given the name of dynamic p -persistence (DPP). To the best of the authors knowledge this is a unique algorithm. It is further described in Section 4.4 on page 43.

Exponential backoff is also looked at and compared to the above two CA schemes.

---

¹ping measures the round trip delay of a small packet

3.1.4 Common downlink for more than one petal

Finally, there is the possibility that the downlink might be shared between multiple petals, each petal with its own uplink. In this case, the downlink has to be of sufficient capacity to support the multiplexing of traffic from different petals.

As well, some method has to be thought up where uplink status information for each petal is included in the common downlink stream. This requires that subscribers be able to identify the petal that they are in.

3.1.5 Interference avoidance

The MILTON system will be interference limited [4]. Intra-cell multipath from neighboring rosettes will be the primary source of this interference. Part of the MAC protocol will have to facilitate dynamic negotiation between the basestation and the subscribers for an uplink band with low interference levels.

As well, power levels in the uplink will be monitored by the basestation and correction factors will be sent back to the subscribers to minimize their transmit power, thereby lowering interference in neighboring petals and cells.

3.1.6 Flexibility

The primary reason for implementing a MAC protocol from the ground up is flexibility. Implementation issues beyond the ones mentioned might appear. If existing technology like cable modems is used, it would be difficult to get inside the hardware to make modifications. Hence, having complete control over the design gives the system designers the power to resolve all technical issues which might occur.

3.2 Conclusion

In summary, such a MAC protocol will have to adapt to different offered loads. The number of users will be a slowly varying quantity, however the traffic they generate will be bursty in nature. Packet traces of actual traffic will be used in the simulations.

Reducing the number of collisions is critical to the performance of the system. The period of time during which two or more subscribers collide is a waste. Reducing the number of collisions will increase the efficiency of the system as a whole.

Taking into account the unique characteristics of the MILTON system, an efficient and easy to implement protocol will be proposed.

Chapter 4

Performance Comparisons of Potential MAC Protocols

Given an offered load, what is the best collision avoidance scheme? It is thought that a system that adapts to offered load would provide an adequate balance between delay and maximum throughput.

All simulations undertaken in the study are written using Microsoft Visual C++ Version 5.

4.1 Traffic characteristics

For something as complicated as Internet traffic, a simple model is not possible, due to the many variables governing the traffic dynamics. To get a reasonable approximation, traffic was captured. Statistics such as size, time, and whether the packet was being sent or received were recorded. All non Internet Protocol (IP) packets were discarded.

In conventional approaches to traffic modeling packets are generated according to a two state Markov model, with one state being no transmissions, and the other state generating packets according to a probability distribution such as Negative Exponential. The problem with this is that in a real network the generation of packets

aren't independent events. the generation of one packet might cause the generation of another packet according the complex rules in the upper protocol layers. For this reason real traffic traces are used in the simulations.

The traffic was captured on a LAN using a packet sniffer. The packet sniffer was written to run in MSDOS using a packet driver to interface to the ethernet card. The ethernet card was put in promiscuous mode¹ on the snoop port of a Cisco Catalyst 5500 switch. Packets were filtered based on the MAC addresses of the involved machines. The captured statistics were put into a file for later processing.

In the simulations each user started using the packet trace in a random position so that the traffic from each subscriber would appear statistically independent. As well, the traffic traces are assumed to be stationary, i.e. the statistics don't change over time. Further, it is assumed that the traffic captured is typical of the average user, which is unlikely. In the authors opinion there isn't expected to be much of a difference because for WWW traffic model the subject generating the traffic doesn't read much faster or slower than a typical person, and for the VOIP traffic the speed of talking again wasn't much different from the typical person.

4.1.1 WWW Traffic

Since the MILTON system will have predominantly WWW users, a trace of WWW traffic is needed. The uplink and downlink packet size Probability Density Functions (PDF) are shown in Figure 4.1 and Figure 4.2 respectively. The sample size is 5665 packets in the downlink and 6475 packets in the uplink. The smallest packet is 60 bytes, the largest is 1514 bytes. This traffic was generated by viewing web sites over a period of two hours. The average bit rate over the two hours is 763 bps, with a

---

¹promiscuous mode means no filtering is done based on MAC address

Figure 4.1: Uplink WWW packet size PDF

maximum of 79 kbps in one second intervals.

The downlink packet size distribution is bimodal. The uplink is unimodal with the majority of packets being the smallest possible, 60 bytes.

As well, the interpacket delay is another statistic available. This is important because there are periods when a user is reading a web page and not generating traffic. This effects how many users a particular channel can support and is often referred to as the burstiness of the data.

In Figures 4.4 and 4.3 a data rate vs. time plot is shown. It can be seen that the data rate has a high rate for short periods of time. It can also be seen that the uplink activity coincides with downlink activity (not surprisingly), and that the uplink data rate is usually less than the downlink rate.

Figure 4.2: Downlink WWW packet size PDF

Figure 4.3: Downlink data rate vs. time

Figure 4.4: Uplink data rate vs. time

4.1.2 Voice over IP (VOIP) Traffic

A packet trace of a Microsoft Netmeeting session was captured for use in the simulations. The session uses the G.723.1 audio codec, which generates a 6400 bits/second stream. Video was not used in the meeting, although that capability exists. The traffic captured represents one conversation between two people.

The overall average bit rate on the uplink was 11944 bits/second, with a peak rate of 23856 bits/second. The discrepancy with the bit rate of the codec could be explained due to headers in each packet. As seen in the coming figures, most of the packets are 78 bytes in length, 59 bytes being header², 19 bytes being payload. The packet size histogram for the downlink is shown in Figure 4.5. It can be seen that all the packets are very small in size.

Figure 4.6 shows the uplink packet sizes, again, all the packets are small.

The time trace for the uplink and downlink is shown in Figure 4.7 and 4.8, the entire conversation lasts for 8.3 minutes. It is seen that when traffic does exist, the traffic intensity reaches a ceiling value that it doesn't go beyond. This is due to the codec's maximum bit rate.

On the uplink it is seen that a similar behaviour is shown, but that activity periods in the uplink usually correspond to relatively idle periods in the downlink. The voice activity factor for the uplink is 65%. In ordinary conversation that value is around 30%, the discrepancy could be due to the fact that the threshold for the microphone is set too high and so background noise is sent as well.

---

²To be even more accurate there should be an extra 4 bytes of header for the MAC layer CRC. It is stripped by the ethernet card, and so the computer never sees it. MILTON will transmit it across the link to save having to calculate it on the other side

Figure 4.5: Downlink WWW packet size PDF for VOIP

Figure 4.6: Uplink WWW packet size PDF for VOIP

Figure 4.7: Downlink data rate vs. time for VOIP

Figure 4.8: Uplink data rate vs. time for VOIP

4.2 Effect of multiple uplinks

Using the WWW traffic characteristics shown in the last section, a slotted ISMA system using an exponential backoff scheme was simulated with different numbers of uplink channels.

The point was to see if increasing the number of uplink channels while holding the aggregate uplink bandwidth constant would improve or deteriorate the throughput. For example, a 1Mbit channel can be used by 1 user, or 5 users each transmitting at 200kbps.

As seen from Figure 4.9, there is very little difference in throughput versus offered load due to increasing numbers of uplink channels.

Figure 4.9: S vs G for different number of uplink channels

As seen in Figure 4.10, the more uplink channels, the lower the average delay. In Figure 4.11 it can be seen that the variance of packet delay follows the same trend.

This is possibly due to a trunking effect. The probability of finding a busy channel when there are more channels is lowered, and therefore the average packet delay is reduced.

As well, the average delay increases exponentially as the offered load approaches 1. This behaviour was described in Section 2.6

Figure 4.10: Average Packet Delay vs. Offered Load

Figure 4.11: Packet Delay Variance vs. Offered Load

4.3 Collision Avoidance - the first approach

Using the WWW traffic model, and a fixed number of subscribers, a simple model of back-off is optimized. This model consists of subscribers waiting a random period of time after the last transmission before transmitting (if they have a packet in their queue). The state transition diagram for this scheme is shown in Figure 4.12.

The original ISMA algorithm is modified to provide ternary feedback. In addition to BUSY and IDLE, now COLLISION is also fed back. Hence the name Modified ISMA.

This state diagram doesn't have provision for the case of one subscriber continuously transmitting, thereby not allowing anyone else to transmit. During the simulations this isn't a problem since the WWW traffic model never has this happen. If this ever happens, it means that the system has an offered load of greater than 1, more traffic than the channel can handle. At that point the system would need to be redesigned to include either more or faster uplinks.

Figure 4.13 is a modification of Figure 4.12 to solve the problem of one user transmitting all the time. Each subscriber is allowed to transmit for a maximum of δ seconds before backing off for a random period of time, regardless if there are more packets waiting for transmission.

The random time chosen depends on a certain probability distribution. For simplicity and fairness every subscriber has the same probability distribution. In order to test the collision avoidance scheme the first probability distribution chosen was the uniform distribution.

If the mean of this probability distribution is too low, there will be a lot of collisions and packet delay will go up. If it is too long packet delay will also go up. Hence,

there is a global minimum in packet delay at some average retransmission period.

4.3.1 Comparing known distributions

A simulation was done, choosing two bounds, a lower (left) retransmission average, and a higher (right) retransmission average, with the minimum between the two values. The bisection method was used to find the minimum. The bisection method works by calculating two new samples inside the left and right sample. If either of the two new values are greater than the maximum of the left and right sample, the algorithm stops. Otherwise, the maximum of the two is used as the new left or right sample. If they are both identical, then one of the two is arbitrarily chosen. The process is then repeated with two new samples between the left and right samples. Figure 4.14 illustrates the algorithm.

Figure 4.15 shows the values calculated for the uniform distribution³ with different numbers of subscribers.

Figure 4.16 shows the values calculated for the negative exponential distribution⁴ with different numbers of subscribers.

These simulations were undertaken to show whether the choice of random back-off distribution affects the minimum possible delay. In Figure 4.17 it is seen that the uniform distribution performs better than the negative exponential distribution: it has a lower minimum delay.

---

³see Figure 2.2

⁴see Figure 2.3

Figure 4.12: Modified ISMA Collision Avoidance

Figure 4.13: Modified ISMA Collision Avoidance with a Limited Transmit time (δ)

Figure 4.14: Bisection Algorithm

Figure 4.15: Uniform Collision Avoidance

Figure 4.16: Negative Exponential Collision Avoidance

Figure 4.17: Comparing Negative Exponential with Uniform

4.3.2 Finding a better distribution

In the previous section the uniform and negative exponential distribution were used, it was seen that the uniform was better. The question is, what distribution is optimal for a given offered load? Using the method of simulated annealing, a random permutation was made to a starting non-optimal CDF and it was seen whether an improvement was made to the average packet delay. If it was, then that distribution was kept, otherwise the original one was kept. This process was repeated many times until there was little improvement seen for a long time.

Using this method, an optimization was performed to find the distribution giving the smallest average packet delay with 5 subscribers. The traffic model is the same as before. The aggregate uplink data rate is 2 Mbps, while the downlink rate is 22 Mbps.

Figure 4.18: Optimal CDF Search

Figure 4.18 shows a few intermediate distributions (as a cumulative distribution function, CDF), until the final distribution when the optimization was terminated. It is unknown how much more improvement would be achievable, however, the incremental gains in average delay declined as time went on, as can be seen in Figure 4.19. Each CDF is composed of 1000 points.

Figure 4.19: Average Delay for 5 Subscribers vs. Optimization Time

The original distribution from where optimization was started was a uniform distribution centered around 11 slots delay⁵, and a width of one slot. From Figure 4.18 it can be seen that to achieve the lowest average delay with 5 subscribers, around 70% of the time there is no delay after the last packet transmission. The maximum delay is around 2 slots.

The average packet delay for the best histogram was 5.19E-04 seconds, or 7.6 slots.

---

⁵a slot being $(188 \text{ bytes} * 8 \text{ bits/byte}) / 22\text{E}6 \text{ bits/second} = 68.3\mu\text{s}$ in duration

better than that achieved by the uniform and negative exponential distributions (9.25 and 9.6 slots respectively, see Figure 4.17).

An optimization was done on 50, 75, and 150 users. Figure 4.20 shows the results.

Figure 4.20: Best Average Packet Delay vs. Number of Subscribers

The simplicity of the CA scheme in this section makes it attractive as an implementation choice.

A possible design strategy is that the maximum number of users is set beforehand, and then the histogram that is optimal for that number of users is used all the time, even if the number of users is less.

4.4 Collision Avoidance - Dynamic P-Persistence (DPP)

The algorithm described here is a p -persistent protocol, with p dynamically changing. As stated in Section 2.5.1, the optimum transmit probability for each subscriber is $1/N$, assuming N contending subscribers. Approximating the number of users contending can be done by measuring the proportion of contention slots that are idle compared to the total number of contention slots. A contention slot is idle if no user uses it. The proportion of contention slots that are idle can be approximated as the probability of future contention slots being idle. This can then be equated to $p(idle) = (1 - p)^N$, where p is the probability each user transmits in a contention slot, and N is the number of users.

The algorithm is implemented in the following way. A value of p is sent down to the subscribers from the basestation. If a subscriber has a packet to transmit, the probability of transmission in any contention slot is p . The channel can be either in contention or idle state. In idle state the value for p is 1, in contention state the value for p is initially 0.5. The choice of 0.5 is arbitrary and could be set to a different value, the usual case is two subscribers colliding, and hence the choice of 0.5. As the number of subscribers contending increases, the number of idle slots reduces. The new value for p is calculated as follows.

There is a transition from idle state to contention state with the first collision. Upon entry into contention state, the idle slot counter (ι) is set to zero and starts incrementing with each new idle slot. The slot counter (σ) is also set to zero and is incremented with each slot. When σ reaches a threshold value of τ , p is recalculated based on the equation $p(idle) = (1 - p)^N \approx \iota/\sigma$, which when solving for N gives

$N = \log(\iota/\sigma)/\log(1-p)$. As previously derived, the new optimal value for p is then $1/N$. The condition for the channel to go back to idle state is when all τ slots are idle, at which point p is set to 1. The value for τ remains to be determined and is channel load sensitive. Figure 4.21 illustrates the previous paragraph in a state transition diagram.

Figure 4.21: Dynamic Collision Avoidance

The above algorithm was tested for the case of 500 subscribers colliding in the first slot, and seeing how many slots it takes for all 500 subscribers to transmit their packets, assuming each packet is one slot in length. When each subscriber has perfect knowledge of the number of contending subscribers, contention takes an average of 1354 slots to be resolved. In Table 4.1 the DPP algorithm is used to estimate the

τ	Number of slots needed for algorithm
5	2527
10	1571
15	1445
16	1460
20	1423
22	1429
25	1414
30	1422
50	1425

Table 4.1: Number of slots needed to resolve contention of 500 subscribers

number of contending users.

The table shows that if τ is too small, the prediction for N isn't very good and the contention ends up lasting longer than when τ is larger. The disadvantage of larger τ values is that if the number of contending subscribers changes rapidly, the system takes longer to adapt. Compared to the perfect knowledge case⁶, it takes approximately 5% more slots.

There are two options with this algorithm: either each subscriber calculates p based on channel status feedback from the basestation, or the basestation could do the calculation itself and feed it back on the downlink. The later approach has the advantage that it reduces complexity in the subscriber units, and also allows new subscribers that are just entering the channel for the first time to know the appropriate p right away. However, the attraction of the first approach is that a few bytes of space are saved on the downlink.

The algorithm in this section was compared against the average packet delay of the algorithm in Section 4.3 to see if there was any gain. For the case of 50 subscribers contending for the channel, the CA algorithm in Section 4.3 had an average delay

---

⁶each subscriber knows the exact number of contending users

of 1.36 milliseconds, while the DPP algorithm had an average packet delay of 0.86 milliseconds, thereby indicating a substantial difference.

4.5 Collision Avoidance - Spanning Tree Random Backoff

The spanning tree algorithm is described in section 2.5.2. In this particular implementation of the algorithm the splitting parameter $Q=2$. Feedback from the basestation consists of the variables *lastchannelstatus*, *level*, *element*, and *parentelement*. Initially all the values are zero and *lastchannelstatus* is IDLE. The variable *parentelement* is used to notify the subscribers which branch of the tree we just advanced from in the event of a collision. The best way to illustrate the algorithm is to provide an example.

Figure 4.22 on page 49 lists the sequence of events relating to 3 subscribers colliding in slot 0. Figure 4.23 shows the same thing graphically. In slot 1, Subscriber 0 and 1 both contend for the same spot, while Subscriber 2 is waiting for level 1, element 0. The subscribers have chosen the element number randomly. Since Subscriber 0 and 1 collide they advance to the next level where they both decide to transmit at level 2, element 0. Since level 2, element 1 is IDLE, the basestation is intelligent enough to realize that level 2 element 0 is guaranteed to be a collision and advances directly to level 3 element 1. The variable *parentelement* is used to notify the subscribers that we advanced from element 0. It's at this point that Subscriber 0 and 1 have decided on different slots, and their contention is resolved. In slot 5, the basestation backtracks to level 1, element 0 and sees if anyone had chosen that slot of transmit on, this gives subscriber 2 a chance to unload its packet. In slot 6 the contention is

finished and subscribers that received packets during the above contention interval can transmit. In this case none do, because slot 7 has *lastchannelstatus* equal to IDLE.

Upon simulation using the WWW traffic model, the results yield a linear utilization versus number of subscribers graph, see Figure 4.24. The comparison line $y = x$ was removed from the graph because the two lines were nearly indistinguishable. It should be noted that utilization is equivalent to throughput, utilization measures the percentage of time the channel is used for the transmission of subscriber data, it excludes time spent in collision.

The average packet delay is illustrated in Figure 4.25. It increases exponentially as the limit of the channel capacity is reached, which is described in Section 2.6. In Figure 4.26 the variance of packet delay is shown. As mentioned in Section 4.1.1, the average rate of the WWW traffic model is 763 bps, and the uplink is 2 Mbps, which gives a maximum of $2E6/763 = 2621$ subscribers. The system was simulated with 2600 users, showing that it can handle an offered load of close to 1, and still remain stable, albeit with an unreasonably large average delay.

Since the throughput vs. offered load graph is already optimal, further improvements would act to reduce the rate at which the average delay ramps up as an offered load of one is approached.

This particular implementation doesn't assign fixed contention slots to each subscriber. Rather, the subscribers randomly choose whether they are contending in a particular slot. This leads to sub-optimal performance, but simplifies implementation complexity [8].

The question remains as to how errors in the downlink will affect the algorithm.

Worst case is that the subscriber which is waiting for a particular *level, element* pair never sees it. As soon as the contention interval is over, i.e. when *level, element* is equal to zero, the subscriber knows that it was skipped over, and therefore it competes in the next interval.

Base station last channel status level element parent element	Slot 0		Slot 1		Slot 2		Slot 3		Slot 4		Slot 5		Slot 6		Slot 7	
	IDLE	TRANSMIT	COLLISION	TRANSMIT	COLLISION	TRANSMIT	IDLE	TRANSMIT	IDLE	TRANSMIT	IDLE	TRANSMIT	IDLE	TRANSMIT	IDLE	TRANSMIT
0	0	1	2	3	1	2	3	3	0	1	0	0	0	0	0	0
1	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
2	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0

Subscriber 0	CONTENDING	CONTENDING	CONTENDING	CONTENDING	CONTENDING	CONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING
slot	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE
transmit state	0	1	2	3	3	3	0	0	0	0	0	0	0	0	0	0
transmit level	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0
transmit element	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0

Subscriber 1	CONTENDING	CONTENDING	CONTENDING	CONTENDING	CONTENDING	CONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING
slot	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE
transmit state	0	1	2	3	3	3	0	0	0	0	0	0	0	0	0	0
transmit level	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0
transmit element	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0

Subscriber 2	CONTENDING	CONTENDING	CONTENDING	CONTENDING	CONTENDING	CONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING	NOTCONTENDING
slot	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	TRANSMIT	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE
transmit state	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
transmit level	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0
transmit element	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0

Figure 4.22: Sequence of events in Spanning Tree Algorithm

Legend: (level,element) channelstatus

Transmit - single subscriber transmitting

Idle - no subscriber transmitting anything

Collision(x) - x subscribers transmitting at same time

Figure 4.23: Spanning Tree Graphical Representation

Figure 4.24: Number of Subscribers vs. Utilization using WWW traffic model for Spanning Tree

Figure 4.25: Number of Subscribers vs. Average packet delay using WWW traffic model for Spanning Tree

Figure 4.26: Number of Subscribers vs. Packet delay variance using WWW traffic model for Spanning Tree

Number of Subscribers	DPP delay mean	DPP delay variance	Spanning Tree delay mean	Spanning Tree delay variance
50	8.68E-4	2.52e-7	8.80E-4	2.23E-7
500	1.27E-2	6.55E-6	8.89E-3	5.69E-6
2000	5.96E-1	3.43E-1	5.30E-1	2.47E-1
2250	1.168	2.10E0	1.98	1.73E0
2400	3.8	6.22E0	4.19	5.80E0
2500	7.98	1.28E1	10.47	1.22E1
2600	44.8	1.34E2	54.7	1.30E2

Table 4.2: Average delay in seconds for DPP and Spanning Tree collision avoidance algorithms using WWW traffic model

Number of Subscribers	DPP (%)	Spanning Tree (%)
500	18.50	19.60
2000	77.30	77.00
2250	84.70	87.16
2400	91.80	91.80
2500	95.20	95.50
2600	99.20	99.20

Table 4.3: Percentage of time channel is busy with data using WWW traffic model

4.6 Comparing DPP to Spanning Tree backoff using the WWW traffic model

In comparing the spanning tree random backoff algorithm to the DPP algorithm of Section 4.4, it is seen that both have nearly linear throughput versus offered load graphs and that the average delay is of the same order for both as well, as show in Table 4.2.

Table 4.3 gives an idea of what percentage of time is spent transmitting data for increasing numbers of users. The remainder of the time the channel is idle.

4.6.1 Reservation - Is It Needed?

Reservation is used to reduce contention. This is done by allowing constant bit rate subscribers to reserve bandwidth for themselves, eliminating their need to constantly compete for the channel. As it is, contention takes up at most 5% of the time using the WWW traffic model, as shown in Table 4.4.

Number of Subscribers	DPP (%)	Spanning Tree (%)
500	5.15	1.60
2000	5.41	3.40
2250	3.95	2.26
2400	2.20	1.59
2500	1.35	0.93
2600	0.27	0.28

Table 4.4: Percentage of time channel is in collision state with WWW traffic model

The WWW traffic model consists of many bursty low bit rate users, so it really isn't representative of a situation where reservation would be used to benefit. In fact, there are no constant bit rate sources in TCP/IP. While there are some modifications to TCP/IP which provide support for different classes of service, the majority of routers and bridges make no attempt to give packets different priorities.

In Table 4.4 a maximum can be seen around 2000 subscribers for the Spanning Tree data. It is thought that this is due to the fact that the simulations were done using unlimited sized transmit buffers in each subscriber. As the offered load increased, each subscriber's transmit buffer increased with a backlog of packets, so that when a subscriber finally did gain access to the channel it would be busy for a long time transmitting the contents of the buffer. Consequently the percentage of time the channel spent in collision state went down as the backlog of packets increased. This also explains the linear nature of the throughput vs. offered load curves, the unlimited

Number of Subscribers	DPP delay mean	DPP delay variance	Spanning Tree delay mean	Spanning Tree delay variance
25	7.52E-3	1.53E-5	5.52E-3	1.30E-5
50	1.32E-2	5.30E-5	1.10E-2	4.98E-5
100	2.61E-2	3.86E-4	2.32E-2	2.13E-4
160	7.20E-1	2.02E1	7.28E-1	1.99E1
166	3.164	1.20E2	3.149	1.18E2

Table 4.5: Average delay in seconds for DPP and Spanning Tree collision avoidance algorithms using VOIP traffic model

sized transmit buffers acts to increase the amount of time the channel spends actually transmitting data instead of in collision state as the offered load approaches 1.

In summary, the maximum benefit of reservation with the WWW model would be to reduce the 5% of the time the channel was in contention. This is not significant enough of an improvement to warrant the additional complexity.

4.7 Comparing DPP to Spanning Tree backoff using the Voice over IP model

As mentioned before in Section 4.1.2, the average bit rate for the VOIP model is 11944 bits per second. With an uplink speed of 2 Mbps, a maximum of 167 users is supported before the offered load is greater than 1. Table 4.5 compares the average packet delay for both collision avoidance schemes. It is seen that the variance of packet delay is initially very small, but it increases as the offered load approaches one.

Table 4.6 compares the percentage of time spent actually transmitting data. It seems to be relatively the same for both CA schemes, which makes sense since they are both using the same packet trace. The VOIP traffic was captured over several minutes versus 2 hours for the WWW traffic model, hence the transmit percentage

Number of Subscribers	DPP (%)	Spanning Tree (%)
25	16.20	16.20
50	32.40	32.40
100	64.58	64.70
160	96.40	96.50
166	99.39	99.39

Table 4.6: Percentage of time channel is busy with VOIP traffic model

converges fairly rapidly to it's final value for the VOIP simulations.

Table 4.7 shows the percentage of time spent in collision state. As with the WWW model, the amount of time spent in collision state decreases as the offered load approaches one.

Number of Subscribers	DPP (%)	Spanning Tree (%)
25	3.6	2.00
50	4.49	4.2
100	6.53	8.42
160	1.15	1.59
166	0.261	0.427

Table 4.7: Percentage of time channel is in collision state with VOIP traffic model

From the last several tables it seems that the two algorithms perform approximately the same under the VOIP model.

4.8 Collision Detection

All the CA algorithms so far require feedback as to whether a collision is occurring. The problem is how are collisions accurately detected? The easiest way is to check for errors in the transmission. If there are errors, it can be assumed that there was a collision. That information can then be fed back to the subscribers.

Usually errors are detected by appending a Cyclic Redundancy Check. With a 32

bit CRC, the probability of an undetected error is around $2.33\text{E-}10$, which makes it possible to detect most errors⁷.

An alternative method is to use Reed Solomon forward error correction which as a side benefit can be used to determine whether an error has occurred, even if the error is uncorrectable.

4.9 MILTON MAC protocol

So far the ISMA algorithm has been used, which works well for the case when feedback from the basestation is much quicker than the transmission of a packet. However, with the hardware we are using in the MILTON system, there is a very significant processing delay in the modulator and demodulator of both the uplink and downlink.

This delay makes the ISMA scheme inefficient. Collisions will occur for a long period until feedback about the channel status gets back. This is shown in Figure 4.28 on page 73 where it takes approximately 13 slots before the data transmitted by the basestation is received by the subscriber, correspondingly it takes approximately 2 slots for data transmitted by the subscriber to be received by the basestation.

A way around this is to adopt the method used in DOCSIS [3] of having contention periods on the uplink, and periods when successful contenders transmit their data. This MAC is relatively insensitive to the total delay through the system, but assumes that the processing delay is the same for all subscribers, which it should be if we use the same hardware for everyone. In the case where the processing delay becomes different for various users, ranging needs to be done at subscriber startup. This is being left out for now to keep things simple.

---

⁷ $1 - \frac{2^{32}-1}{2^{32}}$

As before, slot boundaries coincide with DVB frames in the downlink. A DVB frame occurs every 188 bytes in the downlink, and is marked by the 8 bit value 0x47 (0x## is used to signify a hexadecimal value, where # is a value from 0-9.A-F).

The contention period is when subscribers transmit a request (REQ) for uplink bandwidth. The subscriber then waits for an acknowledgement (ACK) on the downlink. If an ACK is never received, the subscriber retransmits another request at a later time. If an ACK is received, it tells the subscriber on which slot to transmit. Requests are never partially fulfilled, subscribers are given the exact number of slots needed to transmit the amount they originally requested. This eliminates the need to reassemble packets at the basestation.

One difference from DOCSIS is that backoff is done differently. In DOCSIS the subscribers use exponential backoff. In this system, something similar to DPP (as described in Section 4.4) is used.

4.9.1 Change made to DPP method

The overall objective remains the same, the number of users contending is predicted, and the probability of transmitting is then the reciprocal of that. The method used to calculate the number of users contending has changed though.

The problem with the old method is that the number of idle contention slots must be known. Before it was assumed that all idle slots are under contention, however that is not the case if nobody has anything to transmit. To get around this difficulty, two other pieces of information were considered, the number of successful requests, and the number of failed requests (or collided requests). The probability for each is:

$$p(1 \text{ transmit}) = \frac{\text{Number of successful slots}}{\text{Total number of contention slots}} \quad (4.9.1)$$

$$p(\text{more than 1 transmits}) = \frac{\text{Number of unsuccessful slots}}{\text{Total number of contention slots}} \quad (4.9.2)$$

The 'Total number of contention slots' includes slots where subscribers have something to transmit, but don't. Since it is impossible to have access to that value, we take the ratio of Equations 4.9.1 and 4.9.2. This cancels out the 'Total number of contention slots' term in each equation, and leaves us with Equation 4.9.3.

$$\frac{\text{Number of successful slots}}{\text{Number of unsuccessful slots}} = \frac{Np(1-p)^{N-1}}{1 - (1-p)^N - Np(1-p)^{N-1}} \quad (4.9.3)$$

The only unknown in the equation is N , p is the reciprocal of the old value of N . Once the number of successful and unsuccessful slots have reached statistical significance, a new value for N can be calculated. The value for N is always kept greater than or equal to 2 so that subscribers won't collide indefinitely.

The advantage of this method is that the basestation sees all the traffic, it therefore can make a better judgement as to what the probability should be. With exponential backoff each subscriber makes adjustments in back off based on the limited information of its own transmissions, and hence doesn't adapt as quickly to overall changing traffic conditions.

At the basestation a collision is detected when a transmission occurs with power greater than a certain threshold and the CRC doesn't match the transmitted data.

4.9.2 MILTON MAC messaging downlink

The downlink messages are placed right after the DVB frame indicator 0x47. The first value right after 0x47 is an 8 bit header size value. If it is zero, then there are no header bytes in this frame, if it is non-zero, the header is parsed by the MAC unit at the subscriber. Only the 7 least significant bits of the header size is used, the eighth value is the Transport Error Indicator (TEI) bit set by the demodulator if an error occurred in decoding the DVB frame. The DVB frame is ignored if the TEI bit is set. Seven bits provides a maximum header size of 127 bytes out of the 188 bytes in a DVB frame. The format is illustrated in Figure 4.27.

Figure 4.27: Format of a DVB frame

Messages are used in the downlink to:

- 1. notify subscribers when they can transmit (ACK)
- 2. notify which slots are for contention (CONTENTD)
- 3. update master slot counter so that it is common to all subscribers (UPDATE-COUNTER)
- 4. transmit basestation and petal ID (BSID)

Opcode	Message Type
0x00	UPDATECOUNTER
0x01	ACK
0x02	CONTEND
0x03	BSID
0x04	IASW
0x05	LOGON

Table 4.8: Opcodes for Downlink

5. request for administration information from a particular subscriber (IASW)
6. log onto the system (LOGON)

Table 4.8 lists the opcodes for the downlink messages. The opcodes are 8 bits in size, providing room for a total of 256 opcodes.

The format of each message is as follows. they are listed in byte order.

UPDATECOUNTER:

1. opcode 0x00
2. most significant byte of count
3. least significant byte of count

ACK: The subscriber ID is a 48 bit number. similar to the MAC address in ethernet cards

1. opcode 0x01
2. byte 6 of subscriber ID, most significant byte
3. byte 5 of subscriber ID
4. byte 4 of subscriber ID

5. byte 3 of subscriber ID
6. byte 2 of subscriber ID
7. byte 1 of subscriber ID. least significant byte
8. most significant byte of slot number we can transmit on
9. least significant byte of slot number we can transmit on
10. channel to transmit on
11. most significant byte of number of slots for transmit
12. least significant byte of number of slots for transmit
13. most significant byte of REQ slot number we are ACKing
14. least significant byte of REQ slot number we are ACKing
15. power control value used to limit interference in uplink

CONTEND: The N value is generated by the dynamic p -persistent algorithm. Since N is 8 bits in size, up to 255 contending users can be supported at the same time.

1. opcode 0x02
2. most significant byte of slot contention begins
3. least significant byte of slot contention begins
4. number of contention slots

5. N value to use for contention

6. channel number

BSID: BSID is a 16 bit value. The most significant 10 bits are the basestation number, the least significant 6 bits are the petal number.

1. opcode 0x03

2. most significant byte of BSID

3. least significant byte of BSID

IASW: Internet Antenna Status Word request

1. opcode 0x04

2. byte 6 of subscriber ID, most significant byte

3. byte 5 of subscriber ID

4. byte 4 of subscriber ID

5. byte 3 of subscriber ID

6. byte 2 of subscriber ID

7. byte 1 of subscriber ID, least significant byte

8. most significant byte of slot number we can transmit on

9. least significant byte of slot number we can transmit on

10. channel to transmit on

Opcode	Message Type
0x00	REQ
0x01	DATA
0x02	RIASW
0x03	RLOGON

Table 4.9: Opcodes for Uplink

LOGON: Accept log on attempt

1. opcode 0x05
2. byte 6 of subscriber ID, most significant byte
3. byte 5 of subscriber ID
4. byte 4 of subscriber ID
5. byte 3 of subscriber ID
6. byte 2 of subscriber ID
7. byte 1 of subscriber ID, least significant byte

4.9.3 MAC messaging uplink

In the uplink three types of messages can occur, a request (REQ), a data transmission (DATA), and subscriber unit status (ADMIN). The last message ADMIN is used by the basestation to get administrative information.

Each of the messages is differentiated by an 8 bit opcode at the beginning of the message. These opcodes are listed in Table 4.9.

The format of each message is as follows. They are listed in byte order.

REQ: Reed Solomon error correction is used, and so consequently errors can be discovered in the event of a collision

1. opcode 0x00
2. most significant byte of BSID
3. least significant byte of BSID
4. byte 6 of subscriber ID. most significant byte
5. byte 5 of subscriber ID
6. byte 4 of subscriber ID
7. byte 3 of subscriber ID
8. byte 2 of subscriber ID
9. byte 1 of subscriber ID. least significant byte
10. most significant byte of number of bytes in queue
11. least significant byte of number of bytes in queue
12. most significant byte of slot number we are making request on
13. least significant byte of slot number we are making request on
14. channel to transmit on
15. Quality of Service most significant byte. purpose to be determined
16. Quality of Service least significant byte. purpose to be determined

DATA:

1. opcode 0x01

2. most significant byte of BSID
3. least significant byte of BSID
4. most significant byte of number of bytes to follow
5. least significant byte of number of bytes to follow
6. data...
7. most significant byte of piggy back request
8. least significant byte of piggy back request
9. most significant byte of slot number we are making request on
10. least significant byte of slot number we are making request on

RIASW: Return Internet Antenna Status Word

1. opcode 0x02
2. byte 6 of subscriber ID. most significant byte
3. byte 5 of subscriber ID
4. byte 4 of subscriber ID
5. byte 3 of subscriber ID
6. byte 2 of subscriber ID
7. byte 1 of subscriber ID. least significant byte
8. most significant byte of BSID

- 9. least significant byte of BSID
- 10. data to be determined

RLOGON: Request Log On

- 1. opcode 0x03
- 2. byte 6 of subscriber ID, most significant byte
- 3. byte 5 of subscriber ID
- 4. byte 4 of subscriber ID
- 5. byte 3 of subscriber ID
- 6. byte 2 of subscriber ID
- 7. byte 1 of subscriber ID, least significant byte
- 8. most significant byte of BSID
- 9. least significant byte of BSID
- 10. data to be determined

4.9.4 MILTON MAC algorithm in Base Station

In a petal each uplink channel operates as a separate entity from all others. the only thing they have in common is the slot number value which is sent down periodically in an UPDATECOUNTER message for new subscribers coming online.

The BSID is also transmitted periodically for new subscribers. The purpose of the BSID is to be able to identify sources of interference. The BSID is unique for each base station/petal combination in the near vicinity of the basestation.

Requests for transmission are made by subscribers during contention slots. The contention slots are 2 slots⁸ in duration in order to accommodate a propagation delay of up to 10km.

In the downlink, CONTEND messages for each uplink channel are sent to notify subscribers when they can send a request. Right now, the simulations are done with a contention interval duration of 1 contention slot. At the end of a contention interval, the slot number of the next contention interval is broadcast. If no requests were made for that channel, the contention intervals will be back to back.

So in an unloaded channel, there will be one steady stream of contention intervals. As soon as a request is made, there is a break in contention intervals in order to transmit the data, and then the contention intervals occur again.

In the end system a subscriber will have a finite queue size. This means that no one user will be able to utilize the entire bandwidth of the channel, allowing other subscribers to make requests.

At the end of a DATA transmission a piggy back request for more bandwidth occurs, this reduces contention in the uplink REQ slots. The piggy back request is for the same channel.

Once the number of successful and unsuccessful requests reaches a threshold, a new value for N is calculated. Currently the new value for N is calculated using a lookup table. The threshold used so far is if the sum of the successful and unsuccessful slots is equal to 40, this is arbitrary. The lowest value for N is 2, this means that when a subscriber has something to transmit, it will choose a contention slot to transmit a REQ in with 50% probability.

---

⁸two DVB frames, or 2*188 bytes in the downlink

An example of the MAC protocol in operation is shown in Figure 4.28. On the left is two columns representing the slot number and corresponding messages sent down by the basestation. On the right is the subscriber slot number and the corresponding messages sent by each subscriber on the uplink. The arrows point out the relationship between one event causing another.

The sequence of events starts with two messages sent on the downlink, a new count value of 0 and a next contention message saying that the next contention slot is slot 0. A contention slot lasts for two slots, so when slot 2 comes along a message is sent saying that slot 2 is a contention slot as well. This is because no REQ's have been received on the uplink yet. This continues until slot 12 because all this time slot 0 was being processed inside the mod/demodulators in the downlink. Once slot 0 has finally reached the subscribers they contend for the uplink channel. The first contention slot results in a collision between subscribers 1 and 2. In the second contention slot subscriber 3 makes a successful request for 4 slots. The basestation replies that transmission can begin on slot 18, so that is what subscriber 3 does. Subscribers 1 and 2 time out with the requests in slot 16 and retry in slots 22 and 24 respectively.

4.9.5 MILTON MAC algorithm at Subscriber

When a subscriber has something to transmit, it transmits a REQ on the first channel with a contention interval. If more than one channel has a contention interval, one of the channels is chosen at random. This has the effect of distributing the traffic evenly among the channels.

This may prove to be difficult due to the synthesizer in the subscriber unit taking time to settle on the channel frequency. If that is the case, the channel is first chosen,

and then a contention interval is waited for. This has the effect of slightly increasing the amount of time a packet waits for transmission. The traffic will still be distributed evenly among the channels.

As mentioned before, a subscriber transmits a REQ in a contention slot if the random integer chosen in the range $1..N$ is equal to 1.

Also, data may arrive from the subscribers ethernet during the period while waiting for an ACK and before transmitting the data in the original REQ. If this occurs, a piggy back REQ is made at the end of the DATA transmission. By doing this the subscriber avoids contention in requesting bandwidth to transmit the new data.

If a REQ is lost due to a collision, this is detected as follows. The subscriber looks at all ACKs in the downlink. They are sorted according to channel. If an ACK occurs with a REQ slot number greater than the slot number we made our request on, then a collision occurred and we should make another REQ. As well, a nominal timeout period will be set in case there is no other traffic on the channel.

4.9.6 MILTON MAC simulation results

A simulation was run that takes into account all the messages in the uplink and downlink discussed so far. The VOIP traffic model was used.

Amount of Simulation Time Needed

At first, it was not certain how long a simulation should be before the values would stabilize. A simulation for both 100 seconds and 1000 seconds of simulated time and the values are documented in Table 4.10. There doesn't seem to be much of a difference, and so 100 seconds of simulated time is used for all other results.

Number of Subscribers	100 seconds	1000 seconds
166	6.75E-1	6.69E-1
168	7.76E-1	8.02E-1

Table 4.10: Difference in delay for 100 and 1000 seconds of simulated time

Effect of Number of Uplinks on Delay

The results comparing 1, 2, and 10 uplinks is shown in Figures 4.29, 4.30, and 4.31. The 1 uplink case has one uplink of 3 Mbps, the 2 uplinks case has 2 uplinks of 1.5 Mbps, the 10 uplinks case has 10 uplinks of 300 kbps. The total aggregate bandwidth for all cases is the same. The point is to see whether multiple uplinks affects average delay the same way in the MILTON MAC as with the MAC used to generate Figure 4.10.

Figure 4.30 shows the variance of packet delay for the 1, 2, and 10 uplinks cases.

Figure 4.31 shows the throughput for the 1, 2, and 10 uplinks case.

It is seen that the Throughput vs. Offered Load graph is fairly linear, while the delay curves increase exponentially. This is corroborated by the earlier simulations shown in Figures 4.10 and 4.9. The earlier simulations used exponential backoff and a much simpler access protocol. It's encouraging to see that the more complicated MILTON MAC protocol behaves in a similar fashion.

Figure 4.28: Example of MILTON MAC protocol

Figure 4.29: Comparing 1. 2. and 10 uplinks mean delay for VOIP traffic model

Figure 4.30: Comparing 1 uplink and 2 uplinks variance of delay for VOIP traffic model

Figure 4.31: Comparing 1, 2, and 10 uplinks throughput for VOIP traffic model

Exponential Backoff compared to DPP

The case for using exponential backoff as the collision avoidance algorithm was also simulated and compared to DPP. This is shown in Figure 4.32 for average delay, Figure 4.32 for variance of delay, and Figure 4.34 for throughput. It is seen that exponential backoff, at least this implementation of it, isn't as optimal as DPP.

Figure 4.32: Comparing Exponential Backoff vs. Dynamic P-Persistence Average Delay

In Figure 4.35 the ratio between the number of collisions and the number of successful transmissions is shown for both exponential backoff and DPP. It is seen that the number of collisions relative to the number of successful transmissions increases significantly as the offered load increases beyond what seems like 0.35. It may be that this particular implementation of exponential backoff is not optimal. What it does show is that the ratio of collisions to successful transmission reaches a limit as

Figure 4.33: Comparing Exponential Backoff vs. Dynamic P-Persistence Variance of Delay

N increases while using DPP.

The value that DPP levels off to can be derived by setting p to $1/N$ in Equation 4.9.3 and taking the limit as N goes to infinity. This gives a value of $e - 2$. This means that as the number of contenders, N , becomes large, the number of collisions is approximately 72% of the number of successful transmissions.

Figure 4.34: Comparing Exponential Backoff vs. Dynamic P-Persistence Throughput

Figure 4.35: Collision vs. Success ratio

Chapter 5

Conclusion

The original problem statement in Chapter 3 was:

The MILTON system requires a MAC protocol that supports Internet traffic efficiently under varying loads and number of subscribers. As well, implementation complexity and cost needs to be kept down.

In Section 4.9 on page 58 a MAC protocol was proposed that meets the requirements of efficiency and simplicity of implementation.

Different Collisions Avoidance schemes were looked at, dynamic p -persistence (DPP) and spanning tree had the best performance characteristics and performed relatively the same. DPP was chosen based on its ease of implementation. It also required much less feedback on the downlink, and less storage in the subscriber. For these and other reasons, DPP was chosen as the Collision Avoidance algorithm for the MILTON MAC protocol.

No effort was made to provide quality of service levels, traffic is delivered on a first come first serve basis. As a consequence, very little buffering needs to be done in the system, thereby reducing complexity and system cost.

A couple of different traffic types were simulated with the proposed MILTON MAC protocol. This included World Wide Web, and Voice Over IP traffic types. It was found that the MAC has a nearly linear throughput vs. offered load curve, and delay stays relatively low for offered loads below 0.9. In practice the system should be operating at an offered load well below 0.9. Once the offered load increases beyond that point, more capacity should be added in the form of additional uplink channels.

In summary the proposed MILTON MAC protocol satisfies all the conditions originally set out to be satisfied.

In the future it may be desirable to limit the bandwidth used by any one subscriber. This may be done by sending an ACK with a TX slot number far in the future. This acts to reduce the effective throughput of that particular subscriber. The algorithm for the bandwidth limitation would exist in the basestation, the subscriber units would not need to be changed to implement this.

There exists some versions of the IP protocol that provide various quality of service levels, one example being RSVP. A future improvement to the MILTON MAC would be to also provide quality of service levels so that RSVP can be used over the MILTON system.

Bibliography

- [1] Geraniotis E. Chan W. C. and Etemad K., *Multiple-code ISMA for short burst data service in wireless CDMA networks*, Proc. VTC'99.
- [2] DAVIC. Available online at <http://www.davic.org>.
- [3] Data Over Cable Service Interface Specifications (DOCSIS). <http://www.cablemodem.com>.
- [4] J. Duggan and J. Sydor. *Highly structured rosette antenna arrays for the control of c/i in 3-7 ghz multimedia systems*. IEEE International Conference on Universal Personal Communication **Florence Italy** (1998).
- [5] MacKenzie P. D. et al., *On contention resolution protocols and associated probabilistic phenomena*. Journal of the ACM **45** (1998), no. 2. 359.
- [6] "Radio Equipment ETSI and Systems (RES): High Performance Radio Local Area Network (HIPERLAN) Type 1: Functional Specification". Final Draft. Source: ETSI TC-RES. Final Edition. June 1996.
- [7] Gayliard K. T. Goodman D. J., Valenzuela R. A. and Ramamurthi B., *Packet reservation multiple access for local wireless communications*. IEEE Transactions on Communications **37** (1989), no. 8. 885-890.
- [8] Capetanakis J. I., *Tree algorithms for packet broadcast channels*. IEEE Trans. Inform. Theory **IT-25** (1979), no. 5. 505-515.

- [9] Kleinrock L. and Tobagi F. A., *Packet switching in radio channels*, IEEE Transactions on Communications **23** (1975), no. 12.
- [10] IEEE 802.16 Working Group on Broadband Wireless Access.
<http://grouper.ieee.org/groups/802/16/>.
- [11] Prasad N. R. and Nijhof J. A. M., *Indoor wireless communications using slotted non-persistent isma. 1-persistent isma and non-persistent isma cd*, Proc. VTC'97 **3**, 1557-1561.
- [12] IEEE 802.11 wireless LAN standard. Available online for purchase at <http://www.ieee.com>.
- [13] Hafez H. M. Zhang C. G. and Falconer D. D., *Traffic handling capability of a broadband indoor wireless network using cdma multiple access*, IEEE JSAC **12** (1994), no. 4, 1141-1146.