Realtime Multimedia Performance on Wireless LANs

By Richard Akester

A dissertation submitted in partial fulfilment of the

requirements for the degree of

Doctor of Philosophy

of the

University of London

Department of Computer Science

University College London

University of London

This material is based upon work supported by BT sponsorship. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of BT.

Acknowledgements

It has been a long journey, which I could not have made without the assistance of relatives and friends, colleagues and institutions, who supported and encouraged me throughout this process.

It would be impossible to acknowledge everyone who helped me throughout this project. I am grateful to all of them and I would especially like to acknowledge the following people:

British Telecom, to whom I am deeply thankful, supplied the financial support for the completion of this project. I would like to thank all the people I worked with at BT for the encouragement to pursue the PhD, especially Mr. Ian Dufour, Mr. Roger Buck and Mr. Paul Martin who made the sponsorship possible.

I would like to thank staff and colleagues at UCL for their support and advice while conducting the PhD. Special thanks go to Prof. Jon Crowcroft who never failed to remember obscure references to relevant papers, and to my supervisor Dr. Stephen Hailes for his constant guidance particularly regarding research methodology.

Thanks must go to my Parents, especially my mother who provided much support in the early stages of the PhD but unfortunately was not able to see the completion of it. My parents-in-law also deserve thanks for their belief in my ability and of the value of the PhD. Finally, I have to give special thanks to my wife Patricia without whose organisation skills this thesis may never have been completed. Her backing and motivation were invaluable.

Abstract

This project aims to improve the performance of realtime multimedia applications over wireless LANs (Local Area Networks). This is achieved by modifying the multiple access, data-link and application protocols. The realtime multimedia applications studied are interactive (eg telephony) or non-interactive (eg video-on-demand). Performance is said to have improved if delay and loss are reduced according to the needs of the realtime application.

In pursuance to the above stated aim, this dissertation starts by examining multiple access protocols revealing the problems faced by realtime applications. There follows the simulation of the IEEE802.11 MAC protocol and the validation of the results thus obtained.

Point Coordination Function (PCF) is an optional part of the IEEE802.11 MAC protocol and a deterministic multiple access protocol which offers a guaranteed service to real time applications. It requires the use of a wireless access point to coordinate the transmissions of the hosts and thus is only valid in an infrastructure wireless network and not in an ad-hoc wireless network. Therefore, PCF was not included within the scope of this dissertation. Application level redundancy has also been used to reduce loss over all types of network. This technique is analysed and simulated in the dissertation.

The hypothesis is that the performance of realtime applications on an ad-hoc wireless network can be improved by a cross-stack approach to reducing packet loss and delay. The cross-stack approach focuses on improvements at the data-link and application layers. Delay, delay variance

and loss are used to measure realtime application performance.

The originality of this dissertation stems from the proposal of a new distributed multiple access protocol. The protocol is proposed, specified and simulated and it demonstrates improved performance characteristics for realtime applications.

A further original feature of this work consists of a new multicast acknowledgement extension, which is proposed, specified and simulated.

A comparison with redundancy as an alternative approach to reliability is simulated and compared to the new multilple access protocol and the new multicast acknowledgement extension.

It is concluded that the cross-stack approach improves both delay and loss metrics and hence improves the performance of realtime applications over wireless IEEE802.11 LANs therefore proving the hypothesis.

Summary Table of Contents

Table of Contents

Chapter 1- Introduction

1.1 Introduction

Wireless LANs are developing rapidly. Bandwidth and range are increasing while error control techniques such as Forward Error Control (FEC) have improved reliability. Despite limited radio frequency availability, new techniques are delivering greater capacity from limited bandwidths. The "spread spectrum" and "frequency hopping" approaches to radio frequency usage allow wireless hosts to operate even in "noisy" wireless environments. Wireless LANs now provide equivalent capacity to their wired counterparts of just a few years ago. Ethernet, for example, used to operate at a maximum of 10Mbps until recently, and Wireless LANs conforming to the IEEE802.11b standard now work at 11Mbps. Wireless networks extend packet switching technology into areas with low accessibility, can be installed quickly in emergency situations and are self-configurable. They enable seamless roaming and mobility, maintaining broadband connectivity while moving between networks. They are likely to play an important role in the future of computer communication.

Mechanisms for carrying realtime, multimedia data over a wired Internet designed for non-realtime data traffic are still being perfected. Routers and switches are often provisioned with very large buffers and the queuing discipline is simply first-in first-out (FIFO). The result can be long delays and even packet loss when queues overflow. Resources, such as bandwidth, can be dedicated to a realtime multimedia flow to provide a guaranteed or controlled service for the application and reduce delay and loss, but has been held back by the increase in complexity, for example with accounting and billing. In contrast with these methods that are applied at a central point (the router), ad-hoc wireless LANs have distributed control and have a dynamic topology, which limits the opportunities for resource reservation. When wireless links are added to an end-to-end path, they introduce their own problems that also need solving in order to increase the performance of realtime applications. In particular, the method by which hosts compete for access to the wireless medium affects the delay and loss characteristics that are crucial for realtime performance.

In the following chapters, the problems raised by shared wireless networks will be identified and solutions will be proposed and tested. The feasibility of wireless LANs carrying a mixture of realtime and non-realtime data while meeting realtime performance bounds will be investigated.

1.2 Rationale

Carrying realtime traffic over data networks presents many obstacles:

1. Operating System software in host machines is generally not designed to process realtime data. Processor sharing causes irregularities in realtime schedules.
2. Software in routers typically places packets in a single queue per output port (a particular hindrance for realtime packets) and usually do not include active queue management (a technique to signal congestion loss from a router to the data sender) to prevent the overflow of queues and subsequent packet loss.
3. Shared networks - that is networks with a single channel shared by more than one host - are typically optimised for high throughput at the expense of delay. This is due to the presence of the channel "capture effect" which allows a host that has won control of the channel to transmit many frames in a row without having to compete again for the channel.

It is not only the individual effects but also the combination of these factors that can result in unsatisfactory performance for realtime applications.

Host machines can use realtime operating systems, or dedicated processors to ensure realtime bounds are met. These are not widely used at present, though, and lack driver and application support. Routers can use fair queuing and active queue management to reduce delay and packet loss. The performance penalty in terms of packets routed per second may well prevent the use of such algorithms especially in the core network. Shared medium networks require a redesign of the medium access protocol to redress the imbalance in the trade-off between delay, loss and throughput. This is the problem which is the most challenging and for which an accepted solution does not exist.

While hosts connected to a wired network can share a broadcast medium, it is increasingly common for wired networks to be switched to increase capacity per host. Wireless networks on the other hand are broadcast by nature, and although they have access points that can be considered switches, transmissions to the access point are still overheard by all hosts within range, and so a multiple access protocol is still required. Ad-hoc wireless networks allow hosts to communicate directly without the need for an access point. Any multiple access protocol developed should preferably work in an ad-hoc wireless environment, and consequently, should not require an access point to operate.

As wireless networks become more widespread the utility of protocols that allow realtime applications to be used within user's acceptability will increase, as will the significance of such protocols in the overall effort to improve realtime application performance. Since wireless networks will simultaneously be used for non-realtime applications, the ability of such protocols to perform well with the mix of realtime and non-realtime data will be imperative.

It is believed that when solutions to all the above problems have been developed and are widely deployed, realtime applications over data networks will provide acceptable performance for end-users.

1.3 Hypothesis

It is claimed that the performance of realtime multimedia applications on an ad-hoc IEEE802.11b wireless network can be improved by a cross-stack approach to reducing packet loss and delay. The cross-stack approach focuses on improvements at the MAC sub-layer, the data-link and application layers. Delay, delay variance, and loss are used to measure realtime application performance.

Realtime applications only operate within a bounded delay. Hard realtime applications fail when the delay bound is exceeded. Multimedia realtime applications are usually classified as soft realtime applications in that they do not fail when delay bounds are exceeded but degrade beyond user tolerance levels.

1.4 Contribution to Knowledge

The first contribution to knowledge consists of an analysis of the suitability of wireless LANs to carry realtime multimedia traffic within performance bounds while simultaneously carrying non-realtime data. In particular, multicasting performance is examined and frame loss is found to be the main problem. Multicasting over a wireless network is, therefore, unsuitable for both interactive and non-interactive realtime multimedia applications. Examples of applications in this area could be audio conferencing (interactive) and television broadcasting (non-interactive) over a wireless LAN.

The second and most important contribution is a new distributed multiple access protocol demonstrating improved performance characteristics for realtime applications. It is shown that delay, delay variance and loss can all be decreased to within realtime multimedia application bounds for both unicast and multicast cases. The viable use of audio and video streaming and conference call scenarios is now possible due to the benefits of this protocol.

A third contribution is a new multicast acknowledgement extension to IEEE802.11. It is shown that by applying the extension, loss is decreased to within realtime multimedia bounds with little effect on delay and delay variance in relation to interactive delay bounds. Multicast acknowledgements are therefore a possible means of improving realtime multimedia performance without excessive increase in bandwidth requirements.

The fourth and final contribution is an analysis of the benefits of application layer redundancy over a wireless LAN. The loss distribution characteristics of a wireless LAN are very well suited to a simple data redundancy technique; the results indicate that loss can be virtually eradicated. The detrimental effect on bandwidth, especially for more data intensive media such as video, is noted.

Chapter 2 - Background

2.1 Introduction

The following sections present a description of ad-hoc wireless networks in order to show the network environment in which the realtime applications have to operate, an examination of realtime multimedia application issues to understand the needs of such applications, the technique of application layer redundancy which is used to reduce packet loss, and a selection of multiple access protocols to show the design issues and previous approaches to multiple access.

2.2 Ad-hoc Wireless Networks

Ad-hoc networks are self-organising, self-configuring, self-optimizing, multi-hop wireless networks without an infrastructure or backbone. The network itself automatically emerges when nodes cluster together. Nodes, however, can move in different directions at different speeds creating new networks as they move. A multi-hop ad-hoc network is created since any node can be a router and is able to forward traffic on behalf of others. The devices that can be used on an ad-hoc network vary from laptops to PDAs (Personal Digital Assistants) to headsets.

Ad-hoc networks are applicable in scenarios where an infrastructure is not wanted or cannot be deployed. Examples are for spontaneous meetings (at work, airport, etc.), battlefield communication, disaster relief (eg earthquake), listed buildings. All Internet applications should be able to operate over ad-hoc networks including realtime multimedia applications.

Ad-hoc networks present a number of challenges, including routing, security, scalability and power saving. Routing, in particular, has received a lot of attention, with many routing protocols being proposed. The research interest in solving the problem with protocols for routing in an ad-hoc environment has spawned a "mobile and ad-hoc networking" (MANET) working group within the Internet Engineering Task Force (IETF) which has the goal: "to develop and evolve MANET routing specification(s) and introduce them to the Internet Standards track" [MANET 02].

The diameter of a single cell of an ad-hoc network is usually limited. In the case of IEEE 802.11 the diameter of a cell is 100m without obstacles. In practical scenarios with walls and furniture the diameter rapidly falls; values of 30 to 40m are not uncommon and at this range bandwidth drops automatically as hosts detect the drop in power level. For a fully connected ad-hoc network the number of devices is therefore limited. A typical density of laptops, for instance may be one every five square meters. At this density a total of about 30 devices is likely in this case. It is also sensible to limit the number of devices in a single cell to ensure that the devices have enough bandwidth when a lot of the hosts are simultaneously active.

2.3 Adaptive Realtime Applications

The tolerance of modern adaptive applications is such that realtime applications can adapt to sub-optimal delay and loss conditions, and non-realtime applications can adapt to low capacity and packet loss. Applications have advanced to the point where adaptation to the effects of non-perfect network delivery can be achieved to the extent that the quality of the output is still acceptable to the user. Coupled with advances in non-deterministic multiple access protocols, such applications may still be able to offer adequate media quality to the user in the ad-hoc wireless network environment. By measuring the characteristics of the latest multiple access protocols, and comparing them to the tolerances available from adaptive realtime applications, it is possible to give a more objective opinion as to whether the use of realtime applications is possible, and under what network conditions. Ultimately, it is the subjective opinion of the user as to whether the quality of the output is acceptable or not.

In order for realtime applications to function acceptably, bounds have to be placed on packet loss. These bounds vary from media type to media type (eg audio to video), from subtype to subtype (eg voice to classical music), and within subtypes (eg first phoneme of a word, different languages). Packet loss is not so critical for non-realtime applications where retransmission can be achieved within an acceptable delay.

Subjective quality is also critically dependent on delay and variance in delay. Interactive realtime multimedia applications are most affected by delay. Specifically, the round-trip delay determines whether meaningful and stress-free interaction can occur between the parties involved. Non-realtime applications have much higher delay bounds according to the lack of reactivity the user will tolerate.

Realtime applications often require only modest capacity to achieve a given quality but require that capacity to be constantly available. This gives rise to the type of traffic known as constant bit-rate (CBR) that characterises traditional voice communication, for example. Latest developments, however, allow the network more tolerance and applications, which utilise these techniques, can be classified as variable bit-rate (VBR). Such developments include silence suppression (where packets with average energy levels below a certain threshold are not transmitted), automatic codec switching based on feedback (if the loss is too high a more aggressive codec is used), and layered multicast (which transmits different levels of information that can be combined to receive different levels of quality depending on the receiver's wishes). Non-realtime applications are characterised as using all available capacity and will try to increase capacity until loss occurs at which time the transport protocol will back off and steadily increase throughput again.

For real-time audio applications to maintain interactivity, the round trip delay should not exceed 400ms [Brady 71]. Adaptive audio applications compensate for network jitter (delay variance) by buffering, the size of the buffer depending on the current level of jitter. For interactivity the size of the buffer may be limited. If the worst case jitter is greater than the maximum size of the buffer some breaks in the media stream will be expected. By combining the average delay and delay variance, and comparing this with the interactivity bound an indication can be gained as to whether the network performance will be suitable for realtime traffic and an adaptive application. For a session crossing many links, the sum of the delays and the sum of the delay variances for each link will be the relevant measures. Therefore, the performance figures of any single link should be a fraction of the end-to-end delay and delay variance bounds.

Packet loss can reduce audio reception quality, but repair techniques can compensate. With 20ms speech samples per frame, waveform substitution can help to make speech intelligible even with loss rates up to 20% [Hardman 95]. Application level redundancy can be used to reduce loss, the trade-off being an increase in required bandwidth. Wireless MAPs can be particularly susceptible to loss and link-level retransmission techniques are sometimes used with unicast transmissions to combat this, the trade-off here being increased delay. However, as the link layer usually does not know the "time-to-live" (delay bound) of a frame it may attempt retransmission beyond the limit of usefulness of the data. Multicast retransmission is not generally attempted due to its complexity.

2.4 Application Level Redundancy

In the field of reliable audio, application layer redundancy has been used to improve voice reconstruction at the receiver [Hardman 95]. The redundant information comes from the Linear Predictive Codec (LPC), which gives compressed voice packets of limited quality but at the very low bit-rate of 4.8kbps. This information is just what is needed to fill gaps caused by packet loss providing the sound that is expected. LPC adds little overhead to RTP [Schulzrinne 96] packets, for example 12 bytes of overhead to 160 byte PCM or 80 byte ADPCM primary coded packets. The LPC information is piggy-backed onto the primary information of the following packet. The receiver can use the redundant information if the corresponding primary information is lost. Although there is a delay before the redundant information arrives, it is usual for receivers to buffer some information to allow for network variations in delay (jitter) and so the redundant information can usually be processed before the decoded information is played.

Loss events on the Mbone – a multicast backbone overlay network on the Internet - were found to be essentially non-consecutive for low to medium network loads, but consecutive at high loads [Bolot 95]. This suggests that redundant information should be placed in the packet immediately following the output of the primary coding algorithm at low to medium loads, and a few packets later at high loads. Figures 2.1 and 2.2 show the respective positions of the primary and redundant information. It is even possible to have several instances of the redundant information; the first instance for multicast receivers on light to medium load links without consecutive packet loss so that recovery does not adversely affect delay, and another instance in a later packet for hosts on heavy links that are experiencing consecutive packet loss.

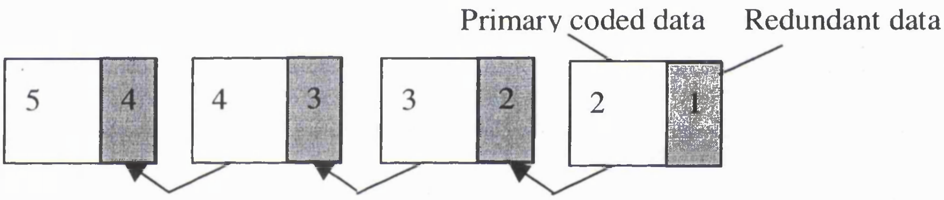


Figure 2.1 – Position of redundant and primary coded data for low to medium load conditions

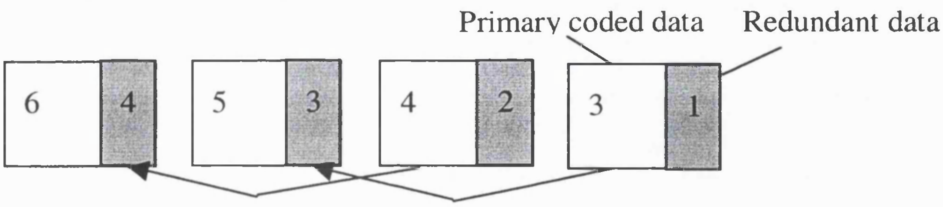


Figure 2.2 - Position of redundant and primary coded data for high load conditions

At low loss rates (up to 20%) and small packet sizes (20ms) other forms of receiver loss repair such as previous packet repetition are adequate, but for higher loss and larger packet sizes (40 and 80ms) LPC redundancy is preferred to other receiver loss repair mechanisms.

2.5 Frequency Division Multiple Access (FDMA)

With FDMA [Tanenbaum 96], the available bandwidth is divided into equal sized frequency bands, each host being assigned one such frequency band for their own exclusive use. The allocation of hosts to channels is typically manually configured into the hosts. In effect, the broadcast medium has been transformed into a number of point-to-point channels, avoiding the problem of contention between users. Although usually used with a centralised controller, a distributed implementation is possible but more complex. For satellite applications, the centralised controller is naturally the satellite, but satellites do not usually perform on-board processing, merely repeating information from the uplinks to the broadcast downlink.

Small frequency bands ("Guard bands") between channels are required to prevent interference. These guard bands can take up a substantial proportion of the available bandwidth. Hosts must also be carefully power controlled. Too much power in the main frequency band causes extra power in the side bands, which causes interference with adjacent channels.

For a small, fixed number of users with a heavy (buffered) load of traffic (eg telephone carriers' switching offices), FDMA is a simple and efficient allocation mechanism. Efficiency is likely to fall dramatically, however, in the following circumstances:

- The number of users is large (There is a higher probability of quiescent users.);
- The number of users is varying (Channels are not always used.);

- The traffic presented is bursty (A typical ratio of peak to mean data traffic is 1000:1).

If less than the total number of hosts are transmitting all the time, bandwidth is wasted which could have been allocated to other active users. To overcome this limitation, dynamic allocation of channels can be used. An example of this is the "SPADE" protocol [Edelson 72], which uses a common signalling channel to allow users to request a channel. Each (of up to 50) ground stations "owns" a slot in this channel and upon sensing that one of the 794 simplex data channel is currently idle, transmits a reservation request on its signaling channel. If, when the request is heard on the downlink, the data channel is still free, the request is assumed to have succeeded and data transmission begins. After the transmission a deallocation message is transmitted on the signalling channel. If two or more hosts attempt to acquire a channel during the same signalling period, a collision occurs and both hosts must retry later. (An alternative algorithm could have been that the host with the "lowest" frequency band wins, where lowest is redefined after every collision to allow fairness between hosts). Disadvantages with this reservation scheme are:

- manual preconfiguration of hosts to slots is needed in the signalling channel;
- the maximum number of hosts is limited;
- the delay (especially with geostationary satellites) inherent in the reservation process;
- the unfairness (one host could take all the channels, blocking other hosts).

Despite these problems, it is the expensive FDMA hardware that makes TDMA more commonly used.

2.6 Wave Division Multiple Access (WDMA)

WDMA is an example of a distributed frequency division technique commonly used on fibre optic, passive star LANs. Fibre can theoretically carry extremely high capacity but practically only a small percentage of this capacity can be utilised due to limitations in the speed of the electrical to optical interface. WDMA exploits the fibre more efficiently by dividing the bandwidth into wavelength bands that are allocated to each host.

In this section a multiple access protocol proposed by Humblet et al. [Humblet 92] is described. Each host is assigned a narrowband control channel (to receive only) and a wideband data channel (to transmit only). Both channels are divided into time slots, which are grouped into frames, and all stations are synchronised from a single, global clock. Frames are marked in a special way, the last slot in the data channel being used to report the status of a host.

When a host wishes to initiate a communication, it tunes to the frequency of the data channel of the desired host and listens to the status channel. This gives information about free slots on the control channel. It then selects a free control slot, and sends a "Data on my output slot 3" message. The receiving host tunes its receiver to the initiating host's data channel frequency and reads the contents of slot 3. [If two hosts try to send simultaneous messages a collision occurs. The data in slot 3 is not picked up and the sender must notice the absence of an acknowledgement and resend a control channel message.]

Connection-oriented schemes also co-exist in the protocol. A host may send a "connection request" message to a receivers control channel. The receiving host announces (on its status data channel) the assignment of the slot in the control channel to the requesting host. [If two hosts try to send simultaneous requests a collision occurs and both try again a random amount of time later.] Both parties now have a conflict free channel of communication. When there is data to transmit, the initiating host sends a "Data on my output slot 3" message (on the connected control channel). The receiving host tunes its receiver to the initiating host's data channel frequency and reads the contents of slot 3.

After a connection has been established, a message of the form "Data on every output slot 3" can be sent. If the receiver has no other commitment for slot 3, the request can be accepted. This emulates a constant bit rate, connection-oriented service.

The concept that receivers tune to senders frequency can cause a problem when two senders instruct a single receiver to tune to their data channel for slot 3. The receiver has to choose one request at random and listen to only one of the two transmissions. Another problem is that every transfer must be proceeded by a request, response communication which imposes addition transmission and propagation delay. It may be preferable for the sender to tune to the receiver's channel and immediately send the data itself, if the frame is short.

Numerous other variations on this protocol are possible. All hosts can share a single control channel, for example. It is also possible to multiplex control and data channels together, allowing just one tuneable receiver and one tuneable transmitter to be used.

2.7 *Code Division Multiple Access (CDMA)*

CDMA takes frequency hopping wireless networks and allows different hosts to transmit simultaneously, using different pseudo-random sequences known by all hosts. Coding theory separates transmissions, using the fact that multiple signals add linearly, allowing a single signal to be extracted. Each bit is encoded into n chips (typically 64 or 128), and each host is allocated an n-bit "chip sequence". A 1 bit is transmitted as the chip sequence itself, a 0 bit is transmitted as the complement of the chip sequence. For b bits the encoded version takes nb bits to transmit which makes CDMA a form of spread-spectrum communication. Although transmissions can begin at different points in time, correlation techniques enable the start point to be determined. The use of correlation to extract a signal can even be used to separate multiple overlapping transmissions even when a single code is used.

CDMA is not easy to configure: chip sequences of all local hosts have to be allocated (manually) to each host. Another disadvantage is the large radio spectrum that is required, if a reasonable bit rate is to be achieved. Power levels also have to be carefully controlled. Finally, despite increasing the length of the chip sequence, physical limitations (eg noise levels) can significantly limit the capacity of CDMA systems.

2.8 *Time Division Multiple Access (TDMA)*

TDMA [Tanenbaum 96] is based on the same principle as FDMA, except the available bandwidth is divided into a number of time slots, rather than frequency bands. TDMA requires all users to agree on the position of slot boundaries. One way to achieve synchronisation is to have one host (a master clock) produce a signal at the start of every slot, so that the other hosts can synchronise. An additional complication is that the propagation time for the time signal to reach the hosts can vary (especially with geostationary satellites) not only because of varying distances from satellite to hosts but also because satellites drift in orbit. The effect is corrected by increasing the transmission speed to compensate for discrepancies, but this reduces usable bandwidth. A number of slots are aggregated into a "frame". The number of slots in a frame determines the maximum number of hosts, each host being allocated a slot number within each frame.

As with FDMA, if a user does not use their allocated time slot, another user cannot use it. Like FDMA dynamic allocation of channels can solve this problem, and with TDMA this can be done in a centralised or distributed way. Advanced Communication Technology Satellite (ACTS) [Palmer 90] uses a centralised mechanism, relying on one of the ground stations to be the MCS (Master Control Station) and manage time slot allocation. Each host is initially assigned one channel and has a dedicated control channel (which allows the system to be contention free). Request messages are sent to the MCS when another channel is required or a channel is to be released.

A similar protocol that allocates a single control channel (bit) to each host, but does not initially allocate data channels is the bit-map protocol. The bit is set to indicate that the host has data to transmit. The bit-map is a group of bits (one for each host) that precedes data frames. Since hosts with a position towards the end of the bit-map have a lower average channel access time than hosts with a bit position at the beginning of the bit-map, alternative proposals BRAM (Broadcast Recognition Access Method) [Chlamtac 76] and MSAP (Mini Slotted Alternating Priorities)

[Scholl 76] suggest that as soon as a host has set its bit, it should transmit its data frame. After transmission the bit-map recommences where it left off. The efficiency of these protocols is the same as the bit-map method, but the delay characteristics, and fairness between hosts is improved especially at low loads.

The access delay for BRAM/MSAP can be further reduced at low loads, (but slightly lengthened at high loads) by using MLMA (Multi-Level Multi-Access) [Rothauser 77] that transmits a host's address in a coded form and relies on the uniqueness of addresses to differentiate transmissions. "Multi Level" refers to the number of levels needed to transmit an address, and varies with the radix used to represent the address. The technique relies on the principle that a transmitted "1" will overwrite a "0". Therefore, hosts listen during "0" bits, to detect any other host transmitting a "1" bit. The protocol can be arranged such that at the end of the contest, each host knows all other hosts with frames to transmit, so all pending frames can be transmitted in order without further contention.

The limiting case where the radix is 1 is known as the binary countdown protocol. If the frame format has the sender's address as the first field of the frame, then the contention bits are not wasted. A problem arises because hosts with higher addresses have priority over lower addressed hosts. To solve this, the use of virtual host numbers has been suggested to allow host numbers to cycle after a successful transmission. The channel efficiency is better than decimal MLMA when there are many bursty stations, but slightly less under full load.

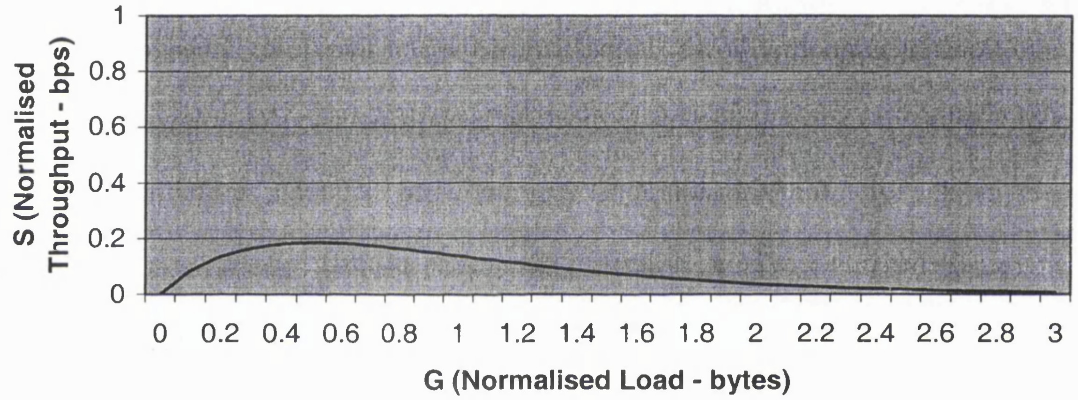
2.9 ALOHA

Pure ALOHA [Abramson 70] is the simplest multiple access protocol, allowing a host to transmit a frame at any time without restriction. It is an example of a pure contention system; meaning collisions between frames are possible. An acknowledgement from the receiving host allows the sender to determine whether or not the transmission was successful. If an acknowledgement is not received within a timeout interval, the frame is resent after a random delay.

To calculate the maximum throughput for an ALOHA channel, the inter-arrival times of the start times of packets are assumed to be independent and exponentially distributed and the traffic source is assumed to be a large enough population of users to be approximated by an infinite number of users who collectively form an independent Poisson source with an aggregate mean packet generation rate of λ packets/second. These simplifying assumptions are necessary to facilitate analysis. The traffic itself is assumed to consist of fixed length packets taking T seconds to transmit. The average number of packets generated per transmission time, $S = \lambda T$. S can also be thought of as the throughput or channel utilisation. The mean traffic offered to the channel (consisting of new transmissions and retransmissions) is denoted by G and can be thought of as the average number of packets per transmission time T, where $G \geq S$.

The channel throughput, $S = G \times P_s$, where P_s = Probability of successful transmission. A packet is successfully transmitted if there are no other transmissions within time T before or after the start time of this transmission. Under a Poisson distribution, the probability of no transmissions during 2T is $P_s = e^{-2G}$. So, $S = Ge^{-2G}$ and S reaches a maximum when $dS/dG = 0 \rightarrow (1-2G)e^{-2G} = 0 \rightarrow G = ½$ (ie when the offered load is half the channel capacity) at which point the maximum throughput, $S = 1/2e = 18\%$ (to nearest percentage.) Graph 2.1 shows throughput values over a range of offered load levels.

Graph 2.1: ALOHA Throughput Against Offered Load



2.10 Hybrid TDMA / Slotted ALOHA schemes

At low load, slotted ALOHA has similar channel efficiency to TDMA, but less access delay. At high load however, TDMA has much better efficiency than slotted ALOHA. Hybrid protocols have been developed which attempt to approximate slotted ALOHA's properties at low load (allowing more contention) and TDMA at high loads (limiting contention).

Reservation ALOHA [Crowther 73] allows more hosts than slots, by not allocating hosts a home slot. Instead hosts must compete for slots, but once a slot is won, the host may continue to use the same slot in future frames while it has data to send.

Unallocated reservation subslots can also be used [Roberts 73]. Hosts select a random subslot and broadcast a short request frame. If the frame does not collide with another request, then the next regular slot is reserved. All hosts must monitor the subslots to determine how many data slots to skip before transmitting.

A protocol to combine stream and bursty traffic was proposed [Binder 75] which allows other hosts to transmit in another host's "home" slot if the slot goes idle. Other hosts can contend for the slot. Collisions cause all non-owner hosts to backoff for one slot, allowing the owner host to recover its slot for future transmissions. Any unowned slots can be claimed by any host. The scheme still relies on there being at least as many slots as hosts.

The "Adaptive Tree Walk Protocol" [Capetanakis 79] starts by allowing all hosts to transmit. If a collision occurs, the number of hosts allowed to transmit is halved (based on bit-map position or address range). This is repeated until a single transmission is obtained. The protocol is called "Tree Walk" because the hosts are logically arranged as nodes on a binary tree, and the tree is searched ("walked") until a level is reached where only one host below that level wants to transmit. By continuing the traversal after a successful transmission, all ready hosts can be identified. The protocol is adaptive because under high load there is no point starting by allowing all hosts to transmit; there will almost certainly be a collision. If each host keeps an estimate of the current network load, the appropriate starting level in the tree can be chosen to maximise the probability of success.

The "Urn" Protocol [Kleinrock 78] applies a probabilistic treatment to the size of the window of hosts allowed to transmit. An analogy is made between the hosts and the balls in an urn. balls are either green (host has a frame to send) or red (host has no frame to send). The aim is to choose the sample size such that only one green ball is selected, based on an estimate of the number of ready stations. If the estimate is that only one station wants to transmit, all hosts are allowed to transmit and the protocol is identical to slotted ALOHA. If the estimate of the number of hosts that want to transmit is more than half the total number of stations, only one station is allowed to transmit and the protocol becomes TDMA.

Priority Oriented Demand Assignment (PODA) [Jacobs 79] is a protocol similar to Roberts' except the boundary between data slots and reservation subslots can vary with demand. Also, future reservations can be made by setting certain bits in the data frame, including frame size and priority, and are either for a single frame or a stream of frames, (the intention being to accomodate data and voice.) Scheduling takes reservation information into account when ordering transmissions, which improves efficiency compared to normal first come, first served scheduling used by the other algorithms. Contention PODA (CPODA) allows hosts to compete for subslots, as with Roberts' protocol, and is used in SATNET, a satellite network between America and Europe. Fixed PODA (FPODA) uses pre-allocated subslots, as ACTS does.

2.11 Carrier Sense Multiple Access (CSMA)

Shared networks are characterised by a single channel shared between multiple independent data sources. One of the fundamental challenges of any shared network is the design of the multiple access protocol. That is, the design of the protocol that controls the approach each host has to accessing the channel, and the method of rescheduling the transmission after an unsuccessful access attempt. The average and variation of the channel utilisation and access delay under various network loads or numbers of transmitting hosts typically measure the success of the protocol. Multiple access protocols are often optimised for throughput under either low or high load, as determined by the persistence of the protocol, and the characteristics of the backoff procedure. This is suitable for non-realtime traffic, but optimising for delay and loss are preferable requirements for most realtime applications.

Wireless local area networks have their own peculiarities that affect the multiple access protocol. Since the radio signal level has a wide dynamic range, and receiving is not possible while transmitting, collision detection is not an available option. Collision detection limits the damage of a collision, improving the channel utilisation and access delay figures. Without it, collisions must be avoided by randomising the transmission time at moments of high collision probability to obtain acceptable performance figures.

Of all the multiple access techniques, CSMA [Halsall 95] is one of the simplest, and most widely used protocols. If a host has a frame to transmit and carrier sense shows the medium is idle, the host may transmit the frame immediately. If two stations transmit simultaneously, a collision occurs. If a frame is to be transmitted and the carrier is sensed to be busy, CSMA recognises that when the carrier next goes idle there is a high probability of a collision, and so defers transmission at this time by a random number of "slot times" (a slot time is equal to the maximum round-trip time) to reduce this probability. CSMA is described as non-deterministic because of the access delay being unbounded.

In P-persistent CSMA a frame is transmitted with probability p, if the carrier is idle. If the frame is not transmitted, it defers until the next slot time and then, if the carrier is still idle, transmits again with probability p. This continues until either the frame is transmitted or another transmission begins, in which case the host acts as if there has been a collision (waits until the end of the current transmission and then repeats the algorithm.) If k hosts are ready to transmit, then the probability that one host will be successful is $kp(1-p)^{k-1}$. This probability is maximised when $p = 1/k$. If the optimum value for p is always chosen the probability of success for as few as five hosts competing is close to the asymptotic value of 1/e (~37%.) p-persistent CSMA improves channel efficiency and delay at high load, but increases delay at low loads.

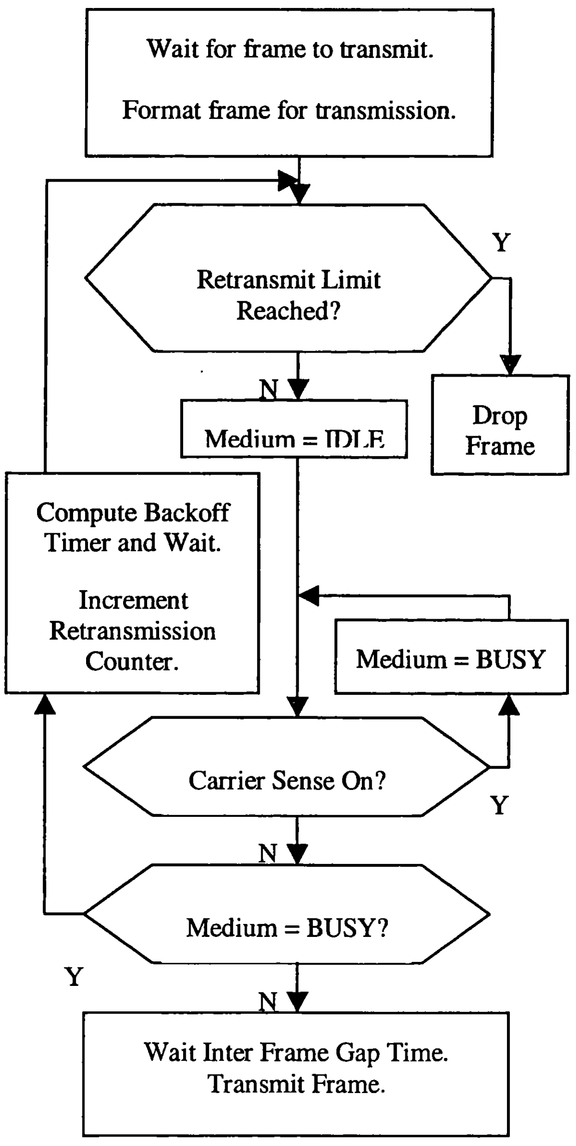


Figure 2.3: CSMA/CA Transmit Process.

Without collision detection, colliding transmissions continue to their conclusion despite the inevitable rejection of the contending frames. Due to the resulting lack of efficiency, the effort to avoid collisions in the first place is increased. By noting that the probability of a collision is highest immediately following a transmission, CSMA/CA (CSMA/Collision Avoidance) defers transmission at this time by a random number of slots to lower the probability of a collision. The mechanism is equivalent to the behaviour of CSMA/CD (CSMA/Collision Detection) after the first collision. If, after the backoff period, carrier sense indicates that another host chose a shorter backoff period, the host acts as if a collision has occurred: it waits for the carrier to go idle, doubles its contention window and chooses another random backoff period. The exact behaviour is shown in figure 2.3.

Treating another transmission as a collision leads to the effect, also noted in CSMA/CD, known as the "channel capture effect" [Almes 79] whereby a successfully transmitting host is able to send subsequent frames uncontended. The currently transmitting host will reset its contention window to the minimum giving it a significant advantage over other competing hosts that are backing off their contention windows. The result with respect to channel efficiency is not detrimental as a host that 'wins' a round of contention can transmit for longer without having to waste bandwidth in other contests. The trade-off is an adverse effect on delay and delay variance, which affects realtime applications.

The binary exponential backoff (BEB) algorithm is unfair because the longer a host has already been waiting, the longer it is likely to delay before attempting to transmit [Lazowska 79]. Alternative backoff algorithms that address this deficiency are described in the MACAW, DFWMAC and BLAM sections.

When a "collision" occurs (the host senses the carrier busy), all active hosts wait a truncated

BEB time before retransmitting the frame. Each host chooses either the first or second slot time in which to retransmit. This means that the probability of a further collision is 1/2 if two hosts are involved (and this probability increases as more hosts are involved.) On the tenth attempt, the backoff algorithm waits for between 0 and 1023 slot times (the maximum). Even at this level, the number of hosts waiting to transmit, and hence choosing an integer in this range, is crucial to the probability that a given host is successful in transmitting.

2.12 Token Passing

As opposed to CSMA, the token passing MAC method gives a bounded delay and is therefore deterministic. The token passing MAC method is independent of the physical topology. It can be applied over bus and star, as well as ring networks.

2.12.1 Token Ring

Token Ring is defined in the IEEE802.5 standard [802.5 85]. It is characterised by a physical ring architecture into which all hosts are inserted, and a token passing access control mechanism. Each station reads the frame by examining each incoming bit, and then repeating it. Thus there is a 1-bit delay added to the ring propagation delay by each host in the ring. Also, the "monitor" host carries, on average, a 27-bit buffer, to ensure that the token can be completely contained on the ring. There can be up to 250 hosts on the ring. Each station is allowed to transmit frames up to a token hold time (THT) of 10ms, by default. This figure limits the maximum size of a Token Ring frame to 4500 bytes for 4Mbps rings and 17800 bytes for 16Mbps rings. Also, an early token release scheme was utilised, which allows a host to transmit the token directly after the final bit of the final frame, without having to wait for that frame to complete the ring.

With any kind of physical ring, the medium is still shared (like Ethernet) but, because each host connects into the ring, there are no collisions. At 16Mbps, each bit takes 0.0625μs to transmit. With 250 hosts in the ring, a 250-bit delay is added (15.625μs). The 27-bit buffer in the monitor station adds an extra 1.6875μs delay.

Priority operation is possible using frame control bits within each frame that indicate the current priority. Each host is able to request that the priority of the ring be raised if it has a high priority frame to transmit. Eight levels of priority are possible.

The insertion/removal of hosts into the ring causes disruption to the ring and hence more management effort to restore order. A frame in transit during a host insertion/removal will almost certainly be corrupted.

With a THT of 10ms, if 250 hosts in a ring decide to hold the token for the maximum time, the worst-case access delay is 2.5s. This is clearly unacceptable for many real-time applications. Even with a priority system, the worst-case scenario is still the same (250 stations all with high priority frames to transmit). To reduce this figure, the maximum number of hosts on the ring can be reduced, the maximum frame length can be reduced, or the speed of the ring can be increased. Decreasing the size of the maximum frame length reduces utilisation (due to extra header overhead), but this may be the price that has to be paid for the faster access times demanded by real-time applications.

2.12.2 Token Bus

This standard is defined by IEEE802.4 [802.4 85]. A bus architecture is logically configured as a ring, allowing a token passing access control method to be used. The operation of the token bus is very similar to the token ring, except that ring management is more complex due to maintaining the logical nature of the ring. In particular, adding and removing hosts is more complex and the initialisation and lost token recovery algorithms vary considerably.

In order for a host to join the logical ring, each host occasionally opens up a "response window" (equal to the IEEE802.3 slot time), when traffic load is low, allowing other hosts to join. It sends a "Solicit Successor" frame that contains its own address and the successor's address. Any host with an address within this range can bid to enter the ring. If exactly one host bids, the host is entered into the ring and the token holder passes the token to the newly entered host. If more than one host transmits, the transmissions collide and the token holder must send a "Resolve Contention" frame. The contending hosts resolve the contention using a combination of the binary countdown protocol (two bits at a time) and collision avoidance (two random bits are used to delay each transmission by 0, 1, 2 or 3 slots). Only one host can enter the ring, per solicitation, to set a bound on the amount of time spent in ring maintenance. Under constant high load the time taken for a host to join the ring is not bounded, however.

When a host wishes to leave the logical ring, it waits until it holds the token, then sends a "Set Successor" frame to its predecessor, with its successors address. It then passes the token to its successor and drops out. A host could chose to leave by just not responding to the token but the other hosts in the ring have to detect and repair the ring.

When a host transmits the token to a host that has gone down, it waits to hear if its successor transmits either a frame or the token. If no transmission is heard, the token is transmitted a second time. If this also fails, the host transmits a "Who Follows" frame that includes its successor's address. A station recognising this address as its predecessor responds with a "Set Successor" frame indicating itself as the new successor. The token holding host then transmits the token to this new successor and the logical ring is healed. If there is no answer to the "Who Follows" frame, the host sends a "Solicit Successor" frame and follows the same procedure as opening a response window to allow hosts to enter the ring.

If a host holding the token fails, the first host to time-out will transmit a "Claim Token" frame, and the modified binary countdown algorithm with random start time is used to resolve contention. If there are multiple tokens in the network, a host that notices another transmission whilst holding a token will drop the token. This will continue until there is only one token.

Initialisation is a combination of the "Claim Token" procedure (when a host fails while holding the token), followed by the "response window" algorithm to allow more hosts to join the ring.

Token bus uses the timed token rotation protocol to control ring access time. Each host keeps a record of the time that has elapsed since it last had the token. If the time is less than a target token rotation time (TTRT) then the host is allowed to transmit up to this time. It can be shown that each host on the ring receives an equal amount of bandwidth, and hence the algorithm is fair. Unfortunately, under increasing load, a lot of hosts, upon receiving the token, cannot transmit waiting frames as the TTRT has expired. This leads to a lot of token passing overhead, and hence low utilisation and higher access delays. With any token passing scheme, the token passing overhead has to be kept low to make the method effective at low loads.

A priority mechanism is used in conjunction with the timed token rotation protocol. Each host is allowed to transmit high priority frames when it first receives the token, up to a High-Priority Token Hold Time (HP-THT). The amount of time taken to transmit high priority frames is subtracted from the Token Hold Time (THT), to determine the time remaining for lower priority frames to be transmitted. The method of transmission of high priority frames is similar to the basic token ring method and so suffers less from token overhead. The amount of time that high priority frames can be transmitted, however, is a preset default, the value of which affects access delay.

2.12.3 RETHER

The Token Bus standard covers the physical and the link layer, although the token passing MAC should be independent of the underlying physical architecture. RETHER (Real-Time Ethernet) [Venkatramani 95] demonstrates how a token passing layer can be added to standard Ethernet to offer more services to real-time applications. The network operates in CSMA/CD mode until a real-time request is received, at which point a broadcast control frame alerts all hosts to switch to token passing. In token passing mode real-time traffic is given access first, non-real-time traffic uses the token for the remaining time in a cycle. When the final real-time stream terminates, CSMA/CD mode is resumed with another broadcast message. If two hosts try to initiate a switch, the one with the lowest address wins. The successful host must receive acknowledgements from all other hosts before transmitting the token.

Real-time data is assumed to be periodic and the period must be an integral number of Token Rotation Times (TRTs), which is a system configurable variable. Each real-time node may hold the token for up to its pre-established Token Holding Time (THT). Each non-real-time node may send at most one frame. It is also possible for the token to pass round multiple streams within one host. The token holds the residual time left in each cycle, a list of currently active real-time sessions with their bandwidth reservations and a list of "dead" nodes in the subnet.

When the host holds the token it reads the current reservation information and determines whether to admit a new real-time session. Non-real-time traffic must be allocated a minimum amount in each cycle, to stop starvation. If the session can be admitted the information is added to the token and the data can be transmitted in the next cycle. Reservations can also be removed when the host holds the token. Due to the state being kept in the token and the requirement that the token must be held whilst a reservation is being made or removed, race conditions are avoided.

For robustness, each host must send an acknowledgement to its predecessor in the logical ring piggybacked onto the token sent to its successor. If an acknowledgement is not received, the node updates the list of dead nodes, and the list of reservations, and passes the token to the next live successor. When a node boots-up it broadcasts a message identifying itself. The node with the token then takes this node off the list of dead nodes.

2.12.4 Fibre Distributed Data Interface (FDDI)

The FDDI standard is formally defined in ISO9314 [ANSI 87]. Physically, the architecture is an optical fibre ring that can be up to 100km in circumference with up to 500 hosts on the ring. Operating at a speed of 100Mbps, it is often chosen for use as a MAN, but can also be used in high performance LAN environments. Due to the encoding used (4B5B with NRZI) two symbols have to be read before a byte can be decoded. Thus, there is a two-symbol delay at each host. At 100Mbps this delay is 0.08μs. The total delay at each station is rounded up to 1μs, in the standard, when internal gate operations are also taken into account.

The propagation delay for a maximum size ring of 100km at 195m/μs (the speed of light in glass) is 0.5ms. When a per host delay of 1μs with 500 hosts on the ring is added the total delay is 1ms.

The maximum frame size for FDDI is given as 4500 bytes. The transmission time for such a maximum length frame is 360μs. The worst-case access delay is when every host has a frame to transmit. With 500 hosts, this is 180ms. Limiting the maximum frame size and number of hosts significantly reduces this figure.

FDDI uses the same timed token rotation protocol that Token Bus uses. The problems with excessive token passing overhead are more apparent on a large ring, where the propagation delay is larger. A priority mechanism (similar to Token Bus) is also available.

2.13 Multiple Access Collision Avoidance (MACA)

CSMA suffers from the “hidden terminal problem” – where carrier sense is unable to detect that a transmission is taking place because the transmitter is out of range, see figure 2.4 - and the "exposed terminal problem" – where carrier sense stops a transmission which could take place without collision, see figure 2.5.

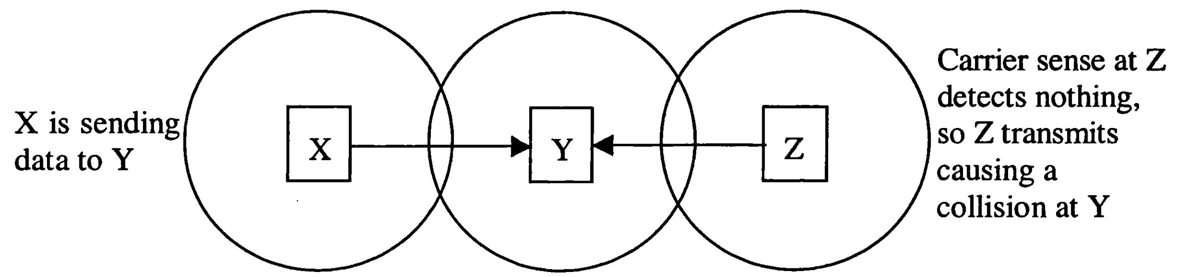


Figure 2.4 – Hidden terminal problem

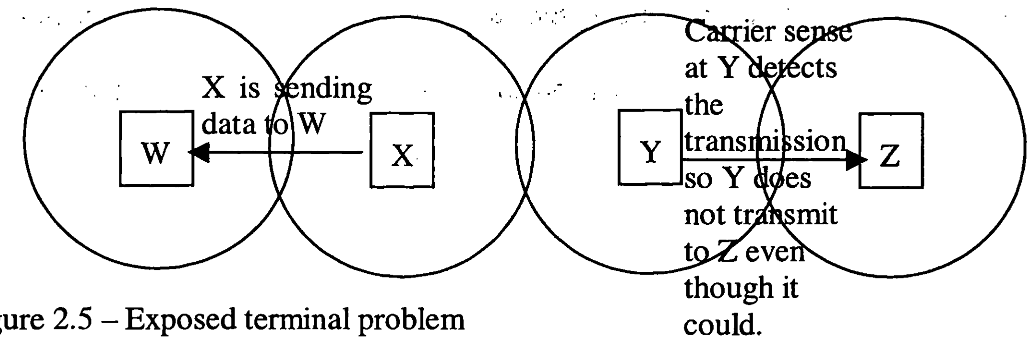


Figure 2.5 – Exposed terminal problem

MACA [Karn 90] attempts to solve these problems by transmitting a short "Request To Send" (RTS) frame preceding the actual data transmission. The receiver is required to send a "Clear To Send" (CTS) in response. Data transmission can then begin. The RTS/CTS exchange is known as "virtual carrier sense" as opposed to "physical carrier sense." Physical carrier sense is not used as it creates the exposed terminal problem – hence the name MACA (CSMA/CA without the CS).

The effect of the RTS/CTS exchange is that the CTS alerts hosts close to the receiver, but potentially out of reach of the sender, that a transmission is about to take place. The CTS includes the upcoming frame length (obtained from within the RTS message), so that the hosts in the vicinity of the receiver will know the duration for which to keep quiet, even though they will not be able to sense the actual transmission.

RTS control frames can still collide, however, and so, in the absence of CSMA, hosts use a randomised exponential backoff procedure before transmitting an RTS. If no CTS is heard within a certain time, the time is doubled before an RTS is retransmitted. This is repeated up to a retransmission limit at which point the sending host gives up. Other hosts hearing an RTS or CTS also add an extra random interval to the time they are prevented from transmitting in order to reduce the probability of collision at this time, as CSMA/CA does. MACA also states that if the data frame is as short as an RTS, then a host can decide not to use the RTS/CTS handshake as this creates unnecessary overhead in this case.

Finally, MACA describes the use of an automatic power control mechanism that adds the power level recorded when the RTS arrived in the following CTS frame. The data packet can then be sent with just enough power to reach the receiver. This allows more spatial reuse of the radio channel. Hosts that overhear a CTS frame, and know the power level necessary to reach the host that transmitted the CTS frame, may be able to transmit an RTS to another host with temporarily lower power without interrupting the forthcoming data transfer. The power level required to send a CTS may be learnt by experience (start with low power and keep increasing every time a retransmitted RTS is heard until data is sent).

2.14 Multiple Access Collision Avoidance Wireless (MACAW)

MACAW [Bharghavan 94] is a fine-tuned version of MACA. A problem with MACA is the unfairness of the truncated binary exponential backoff algorithm. MACAW recognises that backed off hosts cannot compete with a winning host (which has reduced its backoff counter) in the next round of competition, and so introduces a field in the header for senders to indicate the current value of their backoff counter to other hosts. With synchronised counters, all hosts can then compete equally in the next round. Whilst fairer, wild oscillations in backoff counters were observed, which were combated by modifying the backoff adjustment algorithm to a multiplicative increase (*1.5) and linear decrease (MILD). This was found to produce less collisions and higher utilisation, but also produced ever increasing backoff counters in the cases of localised congestion, sender or receiver local noise or unresponsive hosts. MACAW's solution was to maintain a separate backoff counter for each host, to record counters for both sending and receiving ends, and to include this information in the frame headers. It was also found to be fairer for a single host with multiple streams to run the backoff algorithm independently for each stream. The one that chooses the lowest slot time wins. If more than one stream chooses the lowest time, one of the streams is then chosen at random, rather than simulate a collision.

There is no form of error recovery in MACA; it is left to the transport layer to detect the loss and retransmit. The detection process at the transport layer, however, is end-to-end and time-outs are set accordingly. Also, the transport layer considers only congestion as the cause for loss and backs off further transmissions. For losses caused by noise the opposite strategy has to be adopted: immediate retransmission. For these reasons, MACAW uses link layer acknowledgements to allow faster error detection and recovery, in addition to better channel utilisation. The idea of adding the functionality of ACKs (Acknowledgements) into CTS messages and the possibility of using NACKs (Negative ACKs) are suggested. Carrier sense, whilst preventing transmission in exposed terminal situations, was included in MACAW to keeping hosts from transmitting RTSs simultaneously. Multicast transmissions do not use the RTS/CTS exchange, since group membership information is required and CTSs have to be scheduled so as not to collide.

2.15 Floor Acquisition Multiple Access (FAMA)

Rather than being a single protocol, FAMA comprises a family of protocols [Fullmer 95]. A FAMA protocol must include carrier sensing, an RTS/CTS exchange, and a minimum length on control packets, which is a function of the channel propagation delay. This is to guarantee that after an initial RTS/CTS has been successful and control of the channel ("the floor") has been achieved, subsequent data packets do not collide with other RTS packets.

FAMA protocols provide the benefits of MACA in the presence of hidden terminals (because of the RTS/CTS exchange) while providing comparable or better throughput than CSMA without the presence of hidden terminals (due to the use of carrier sense).

FAMA advocates the use of "packet trains" – sequences of packets sent consecutively – once the floor has been acquired. This increases throughput, especially when the transmit-to-receive turnaround time is high (a few milliseconds), and thus the overhead associated with sending control frames increases. The number of frames that may be transmitted before the floor is relinquished is limited.

2.15.1 FAMA-PJ (FAMA - Pauses and Jamming)

FAMA-PJ [Garcia-Luna-Aceves 95] consists of both carrier sense and a collision detection mechanism based on control packets and jamming that prevents collision of data packets with control packets or other data packets. The minimum control packet size required to enforce floor control is specified as a function of the channel propagation delay and the transmit-to-receive turnaround time. FAMA-PJ provides better throughput, more stability and better delay characteristics than non-persistent CSMA.

A host that is just initialised must wait for twice the maximum channel propagation delay plus the transmit-to-receive turnaround time before transmitting. This allows the host to learn about ongoing packet trains, if they exist. The host may then send an RTS frame that must be longer than twice the maximum channel propagation time. After transmission, the host pauses for one maximum channel propagation delay. If the carrier is idle for this period, then it is assumed that the transmission was successful and one or more data frames may be transmitted. If a transmission is heard while pausing, the host sends a jamming signal for one maximum propagation delay. Sending a jamming signal after failing to send an RTS is called "active jamming" and is used to inform other hosts that a collision of control frames has occurred. This technique can fail if the transmit-to-receive turnaround time is greater than the maximum propagation time, when two RTS frames can collide without detection. To prevent this FAMA-PJ uses "passive jamming". This involves hosts, other than the RTS sender, detecting carrier and waiting to receive a valid RTS frame. If one is not received, the host sends a jamming signal for the transmit-to-receive turnaround time plus twice the maximum propagation delay. This ensures the RTS sender will detect the collision. If a collision is heard, the host backs off a random amount of time taken from the uniform distribution between zero and ten times the time taken to transmit an RTS frame.

The minimum size of an RTS frame must be larger than twice the maximum propagation delay. If it is not, it is possible for an RTS frame to be correctly received by a host before the RTS then collides with another control frame. No jamming would occur and it is then possible that a data frame could collide with a control frame.

2.15.2 FAMA-CR (FAMA – Collision Resolution)

FAMA-CR [Garcés 96] uses non-persistent CSMA and an RTS-CTS-DATA three-way handshake. Its distinguishing feature is the use of "collision resolution" for RTS frames, which is designed to resolve collisions between RTS frames faster than the CSMA backoff algorithm.

All hosts in FAMA-CR are manually configured with a unique ID. The MAC protocol assumes a fully-connected ad-hoc network to ensure that all hosts are aware of all transmissions. The multiple access scheme then works as follows. A host with a frame to transmit must first sense the carrier for one maximum round-trip time. If the carrier is busy, the host backs off before attempting to transmit an RTS frame. If the carrier is idle, the host may transmit an RTS. The host then waits for one round-trip time plus the time needed to transmit a CTS frame. If the CTS frame is received after this interval, data frames may be sent, up to a limit. If a CTS frame is not received, collision resolution is used. All hosts divide an interval (lowID, hiID) in half, where the lower half contains IDs of hosts that must backoff and the upper half contains IDs of hosts that may retransmit the RTS. The backoff half of the interval is stored on a stack. This procedure is repeated each time a collision is detected. During collision resolution, if a host detects the medium idle for one round-trip time it removes an interval from the stack and uses this as its new list of hosts able to transmit. When one host successfully transmits an RTS, receives a CTS and transmits its data frame, the hosts then take the next interval off the stack and the contest continues until all hosts have transmitted their RTS frames.

A host that has received a data frame must wait for one maximum propagation delay, to allow the sender of the data frame to transmit more data frames if it has any. This allows packet trains to be transmitted by the sender.

2.16 Distributed Foundation Wireless Multiple Access Control (DFWMAC)

IEEE802.11 [802.11 96] aims to standardise the Medium Access Control (MAC) and Physical Layers of Wireless LANs. The currently proposed MAC protocol for the standard is the "Distributed Foundation Wireless MAC".

DFWMAC, like other modern multiple access protocols, has recognised the trend towards using data networks for a mix of both traditional elastic data traffic and realtime data flows over the same channel. To address the different requirements of these types of data, two “coordination functions” are layered onto the basic underlying access mechanism. The functions correspond to the two groups of data traffic aforementioned, elastic traffic being envisaged to be carried by the distributed coordination function (DCF) and the realtime traffic carried by the point coordination function (PCF). The two functions coexist and operate simultaneously on the same channel.

In an ad-hoc wireless network the DCF function is the only choice available since there is no central, fixed access point to act as a coordination point for polling and token passing schemes are inappropriate for the more error-prone, non-static and not necessarily fully-connected wireless ad-hoc network. The DCF operation has, however, been altered from the original CSMA/CA protocol from which it derives to improve the transport of delay-bounded traffic compared with the original CSMA/CA.

The DCF is the "foundation" in DFWMAC and consists of a CSMA/CA protocol with additional reliability provided by acknowledgements, and an RTS / CTS handshake (which is only used if the length of the data falls above a fixed threshold), if the payload length is above a set threshold, to alert hidden stations of a forthcoming frame transmission [Tobagi 75]. The potential unfairness for hosts that choose large deferral periods before transmitting is addressed by "freezing" the period at the point another transmission begins and restarting it after the transmission has finished. This gives the host an advantage over others choosing a random number from the larger contention window, ensuring that a host that has been waiting for longer is more likely to reach the end of its backoff period and transmit than newly contending hosts and hence removing the channel capture effect. When a collision occurs, the BEB algorithm is used to determine the size of the new contention window. Initially the contention window is set to 16 slot times. A flowchart for the DCF transmit process is shown in figure 2.6.

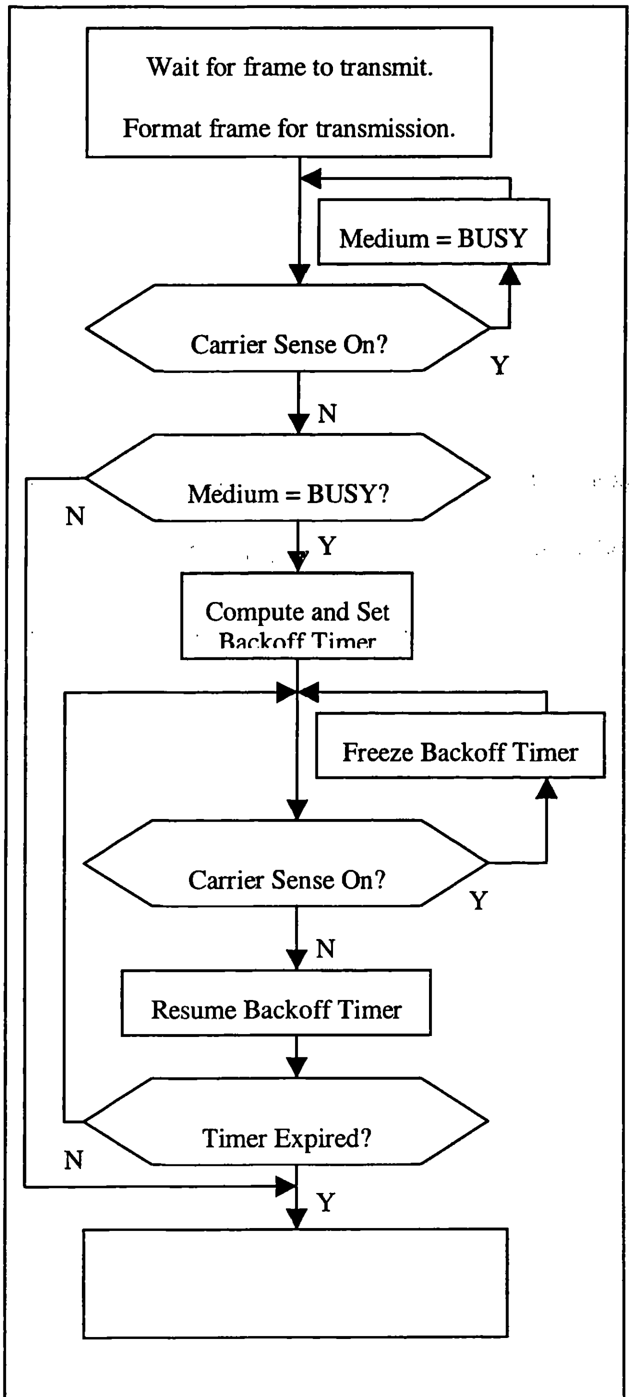


Figure 2.6: IEEE802.11 DCF

DCF collision detection is implemented, for unicast transmissions, as follows. The receiver of a data frame must return an acknowledgement to the sender, putting the acknowledgement at the front of the transmission queue, before any other data packets that are waiting to be transmitted. The sender must retransmit the data frame if an acknowledgement is not received within the timeout interval that is a fixed value and a function of the propagation delay (for both the data frame to arrive and for the acknowledgement frame to be returned), the short inter-frame space and the transmission delay for an acknowledgement. The data sender only starts the timer after the frame has been sent. The sender cannot transmit other frames while in backoff, but if a data frame is received, it must acknowledge it before returning to the backoff state. Receivers of data frames can pass the data up to higher protocol layers before sending the acknowledgement (acknowledgements are not themselves acknowledged). Once an acknowledgement for a data frame that the host has sent has been received it must go into backoff before returning control to higher protocol layers that may be waiting to transmit other frames.

To make the protocol more efficient, IEEE802.11 gives priority to acknowledgements in two ways. Firstly by varying the IFS (inter-frame space) to allow priority acknowledgement frames to transmit first. The normal inter-frame space defined as 16μs; acknowledgements use an 8μs "short inter-frame space" (SIFS). Secondly acknowledgements are allowed to be transmitted without using CSMA/CA. in other words acknowledgements do not follow the usual backoff procedures. Another small amendment further protects the acknowledgement from other station's transmissions. Stations that hear data frames may be out of range of the receiver and so will not hear the subsequent acknowledgement. Upon receiving the data frame, however, these stations should allow extra backoff time for the acknowledgement to be sent.

The PCF is built on the functionality of the DCF and uses a priority transmission mechanism to enable an "access point" (AP) to seize control of the medium and poll stations, listed on its polling list, thus controlling access to the medium, and providing a contention-free service. The order in which hosts are polled (the scheduling) is determined by the implementer, and is therefore an area where products can compete on channel efficiency grounds.

Stations may reserve bandwidth within the contention-free period by submitting a request to the

AP (during a contention period), giving two parameters:

- A time bounded frame period (which must be an integer multiple of the "SuperFrame Period" (SFP));
- The maximum frame size (per the time bounded frame period).

The AP replies, accepting or denying the request, based on its admission control decision, and assigning a "connection ID" to identify the flow. The station is then guaranteed a bounded delay for the duration of the session.

The DFWMAC SuperFrame Period (SFP) is identified by a beacon from the AP and defines the maximum possible frame rate. This parameter affects the granularity and hence the services which can be supported. The AP dynamically varies the duration of the contention free period according to the current load. It has to be possible to transmit at least one contention frame, per SFP.

[Weinmiller 95] fine-tune the DCF parameters, and suggest improvements to the backoff and RTS/CTS algorithms. The observation is made that newly entering hosts can still choose a low value and compete with previously competing hosts, increasing the likelihood of collision. One solution is to have newly entering hosts select slots that are not likely to have been used in previous competitions. By observing previous competitions, the range of frozen counters can be calculated, and newly competing hosts will chose from values above this range. Lower collision rates at all load levels are observed, in simulations, compared with the original DFWMAC scheme. The use of RTS/CTS threshold was found to depend on more than the payload length.

Configuration and geometry of the hosts, the network load and preamble length were also relevant to the decision.

2.17 Elimination Yield Non-preemptive Priority Multiple Access (EY-NPMA)

EY-NPMA [ETSI 95] is the MAC protocol used by ETSI's HIPERLAN wireless LAN standard. It uses CSMA/CA as a base but includes an extra elimination stage before the yield stage (which defers to other transmissions by a number of slot times, as DFWMAC does.) It also includes a more complex priority transmission scheme.

The first stage of channel access is the priority phase. The priority consists of two parts, a user priority that can be either normal or high, and a "maximum transfer delay" or "lifetime", which is dynamic and decreases as the frame waits in the queue. Higher priority frames are transmitted first, based on the "channel access priority" (CAP) that is a combination of the two parts, and can take values from 0 to 4.

The next stage is the elimination phase. Each competing host sends between 0 and 12 bursts, each 256 bits in length, where the number of bursts is chosen randomly from the geometric distribution with mean 0.5. The host then listens for one burst, and if a transmission is heard the station defers.

The final stage is the yield phase. Each remaining host listens for 0 to 14 bursts, each 64 bits in length, where the number of bursts is chosen randomly from the geometric distribution with mean 0.875. The host choosing the lowest value will transmit first and other hosts will defer.

This complex contention resolution procedure ensures the number of collisions is never more than 3.5% for any number of hosts (hidden nodes excluded), but takes on average 2300 bits and in the worst case 5000 bits. For small packets this overhead reduces channel efficiency considerably.

2.18 Binary Logarithmic Arbitration Method (BLAM)

BLAM [Molle 94] is a protocol designed to improve the CSMA/CD IEEE802.3 protocol. The specification describes a number of alterations to the basic CSMA/CD protocol, some of which are dependent on the wired medium. The backoff method, however, can be applied in a wireless environment. The backoff extension used by BLAM is called "logarithmic arbitration" and is based on the observation that the size of the contention window is an estimate of the number of hosts currently attempting to transmit, which will be denoted by Q. If the estimate is accurate, then Q hosts are contending for the same number of slots. Thus, if no activity is sensed while backing off for two slot times, it is assumed that the initial estimate for Q was too large and the number is halved. It is this halving of the contention window that gives the algorithm its "logarithmic" name (following the tradition of "exponential" as used in the doubling of the contention window after collision detection in CSMA/CD.) This new contention window is then used to calculate a new backoff time, and the algorithm repeats until transmission is scheduled within two slot times.

BLAM is designed to decrease the access delay but also increases the chance of collision. Since the number of hosts waiting to transmit is more accurately estimated, however, the benefits of lower access delay offset the higher chance of collision and overall throughput is increased. Lower access delay is a key requirement of interactive realtime multimedia applications, which makes BLAM particularly suitable for this class of application.

2.19 Critical Analysis

The multiple access protocols from ALOHA to DFWMAC have been described in chronological order. These protocols were also ordered in terms of complexity, each protocol adding an extra element to solve a problem or improve performance. One of the first wireless LAN products on the market was NCR's "WaveLAN" which used the CSMA/CA protocol. The first MAC protocol to become a standard was DFWMAC in the IEEE802.11 standard. A comparison of these protocols is useful to show the specific shortcomings of the CSMA/CA protocol, and the performance metrics that benefit from the changes made to DFWMAC.

For non-realtime TCP (Transmission Control Protocol) connections, the important metric is throughput. For interactive realtime applications the important metrics are delay, delay variance, loss and loss distribution. The original design optimisations of many best-effort MAPs were for throughput (TCP flows) at the expense of delay (realtime flows). CSMA/CA increases throughput by allowing a host that has won control of the medium to transmit all the packets it has queued without having to regain control of the medium between transmissions. This channel capture effect increases throughput at the expense of delay (since other hosts must wait longer to transmit delay sensitive data). DFWMAC makes changes to the CSMA/CA backoff algorithm in order to remove the capture effect, so the delay statistics for DFWMAC should be lower than CSMA/CA. The trade-off is that with more competition comes more chance of collision and hence frame loss. The effect of the DFWMAC backoff algorithm should be to reduce throughput, reduce delay and increase loss.

DFWMAC also includes a reliability mechanism whereby the receiver sends an acknowledgement back to the sender of a data frame. The effect of this is to increase throughput (since loss is reduced), increase delay (and delay variance in particular), and reduce loss. The effect of these changes on realtime and non-realtime applications depends on the magnitude of the changes. These metrics can be measured using simulation techniques.

	Throughput	Delay	Loss
DCF Freeze/Resume backoff feature (removes channel capture effect)	⇓	⇓	⇑
DCF reliability feature (acknowledgements and retransmissions)	⇑	⇑	⇓

Table 2.1 – The effect of DCF mechanisms on network performance

2.19.1 Simulation of CSMA/CA and DFWMAC

To study the performance of these protocols further it is necessary to simulate them. As can be seen from Table 2.1 analysis shows that DCF features have contradictory effects on network metrics, and a simulation makes it possible to observe the overall effect of these mechanisms on network performance.

The simulation package "ns" version 2.1b2 was used to simulate the performance of the protocols. A number of nodes were created with "CBR" (Constant Bit Rate) connections used to transmit 1640 byte packets every 33ms. This is similar to the size and frequency of video packet generation, including a 20 byte IP header, an 8 byte UDP (Uniform Datagram Protocol) header, a 12 byte RTP (Realtime Transport Protocol) header and 1600 bytes of H.263 video (at 384kbps and 30 frames per second). Other nodes were created with "TCP Reno" connections used to transmit 1000 bytes packets (including a 20 byte IP header, a 20 byte TCP header and 960 bytes of data) from a simulated FTP (File Transfer Protocol) source application. These parameters are shown in table 2.2.

Connection type	Data size	Header overhead	Total packet size	Frequency
CBR	1600	40	1640	33ms
TCP Reno	960	40	1000	Depends on TCP error, flow and congestion control.

Table 2.2 – Connection parameters

Streams were created in pairs (one CBR and one TCP) up to 22 pairs (equivalent to 100% load). Figures 2.7 and 2.8 show the first two simulation topologies. Adding one more host with two more connections creates further topologies.

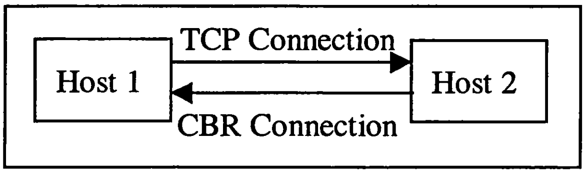


Figure 2.7 - Initial simulation topology

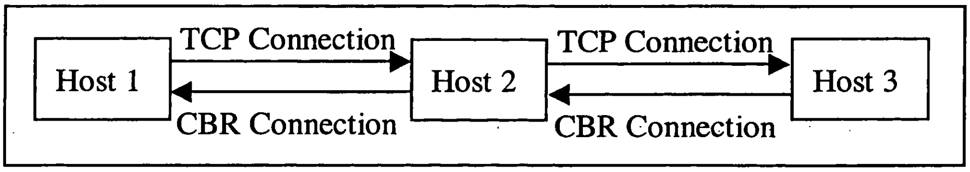


Figure 2.8 - Second simulation topology

The simulation was limited to 44 transmitting hosts due to the physical limitations of an ad-hoc, wireless cell. As was described in section 2.2, the number of hosts in a single cell is limited and under normal circumstances only a small number of hosts will be simultaneously transmitting. In this simulation not all the hosts are in range of each other extending the permissible distance between hosts and testing the hidden terminal scenario. The presence of TCP transmitters ensures that the network is at full load by the fact that TCP uses all the available bandwidth. As more TCP transmitters are added the network load remains full as TCP shares the available bandwidth amongst the competing streams. For the H.263 streams, the amount of bandwidth they use (384kbps) is a small fraction of that available in an IEEE 802.11b network (11Mbps). Taking MAC headers and inter-frame spaces into account, it takes 22 CBR transmitters to drive the network to capacity.

The Gilbert [Gilbert 60] wireless error model was used to simulate the error prone wireless media. The simulation includes no forward error correction (FEC), as this is not used in the IEEE802.11b standard.

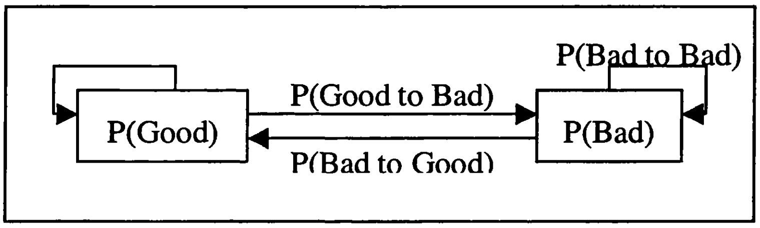


Figure 2.9 – Probabilities in the Gilbert model

In the simulation, the probability of having a bit error in the good state [P(Good)] was 0.1, while the probability of having an error in the bad state [P(Bad)] was 0.4. The probability of staying in the good state [P(Good to Good)] was 0.96, while the probability of staying in the bad state [P(Bad to Bad)] was 0.94. The other probabilities are derived from these [eg P(Good to Bad) = 1 – P(Good to Good)].

The time at which each stream was started was staggered (within the first 20ms) to allow the multiple access protocol to take effect. Starting all the streams at exactly the same time is unrealistic and only shows the behaviour where CBR transmissions collide every 33ms. The TCP connections were started first and allowed to stabilise for 15 seconds (converge on the available bandwidth) before the CBR connections were started. The simulation was run for a total of 30 seconds. The simulation was repeated 25 times and average metrics obtained. Each simulation used random numbers seeded with a different random number from a different instance of a random number generator to ensure adequate variation in the starting conditions of the simulator. Figure 2.10 shows the starting times of the connections.

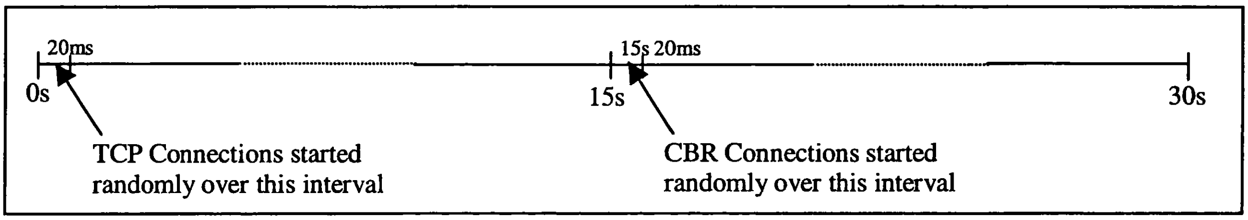


Figure 2.10 – Diagrammatic representation of the connection starting times

The network parameters of the IEEE802.11b standard were used, which include a wireless network bandwidth of 11Mbps. At this rate, each CBR transmission takes 1164μs to complete and each TCP transmission takes 727μs. Each CBR connection accounts for 3.5% of the theoretical network capacity. The TCP connections should use all the remaining bandwidth. A fixed propagation delay of 4μs was used between all hosts. A 1ms link layer delay was included to simulate operating system latencies. A minimum contention window of 16 was used, as stated by the IEEE 802.11 standard. Figure 2.11 shows the format of the packets used.

MAC Header	IP Header	UDP Header	RTP Header	CBR DATA

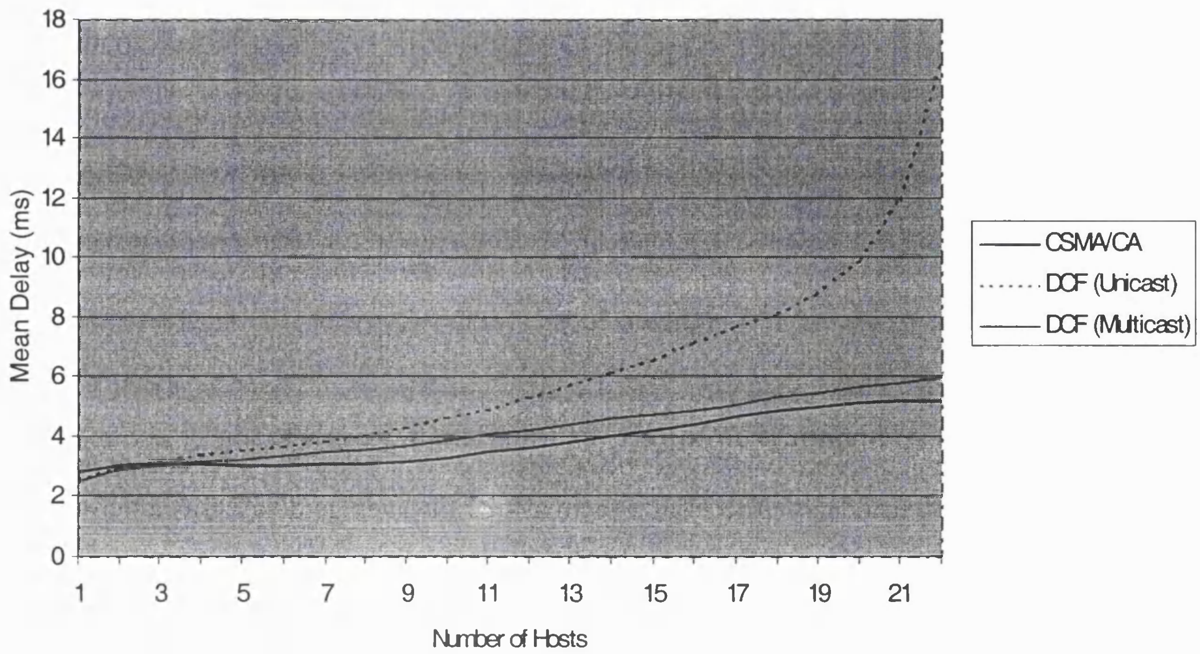
MAC Header	IP Header	TCP Header	TCP DATA

Figure 2.11 – Format of CBR and TCP packets used in simulation

For the DFWMAC connections, all frames fall below the RTS/CTS threshold, so this exchange is not done. For the CBR connections, both unicast and multicast destinations were tested. For the multicast transmissions, the DFWMAC protocol states that acknowledgements should not be used. Unicast transmissions are acknowledged in the same way as TCP.

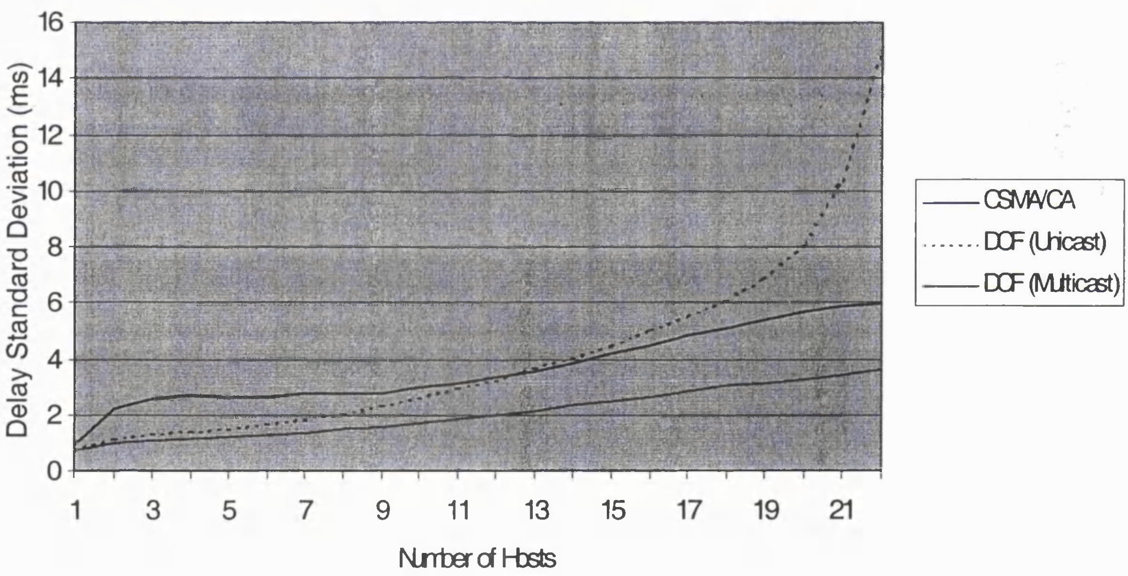
The results were analysed to extract average delay, delay variance, loss and loss variation for the CBR connections and throughput for the TCP connections.

Graph 2.1: Mean Delay against Offered Load



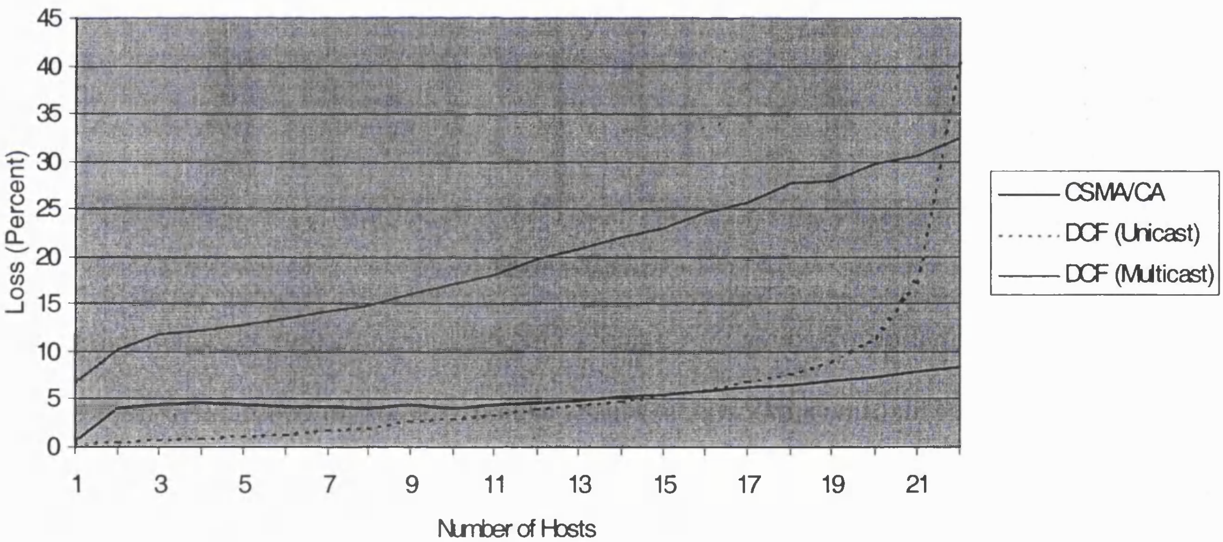
The first results are of average delay for all the CBR frames in all the CBR streams. Delay is measured for CSMA/CA, DCF unicast and DCF multicast. CSMA/CA generally has the lowest delay as it does not attempt retransmission in the event of a collision. Retransmissions increase the average delay as can be seen in the DCF unicast figures. DCF multicast has average delays close to CSMA/CA as it does not use retransmissions to improve reliability. DCF multicast freezes and resumes the backoff counter delaying transmission, whereas CSMA/CA finishes the backoff quicker leading to slightly lower average delays. After 25 runs, the average delays had converged to within 3%.

Graph 2.2: Delay Standard Deviation against Offered Load



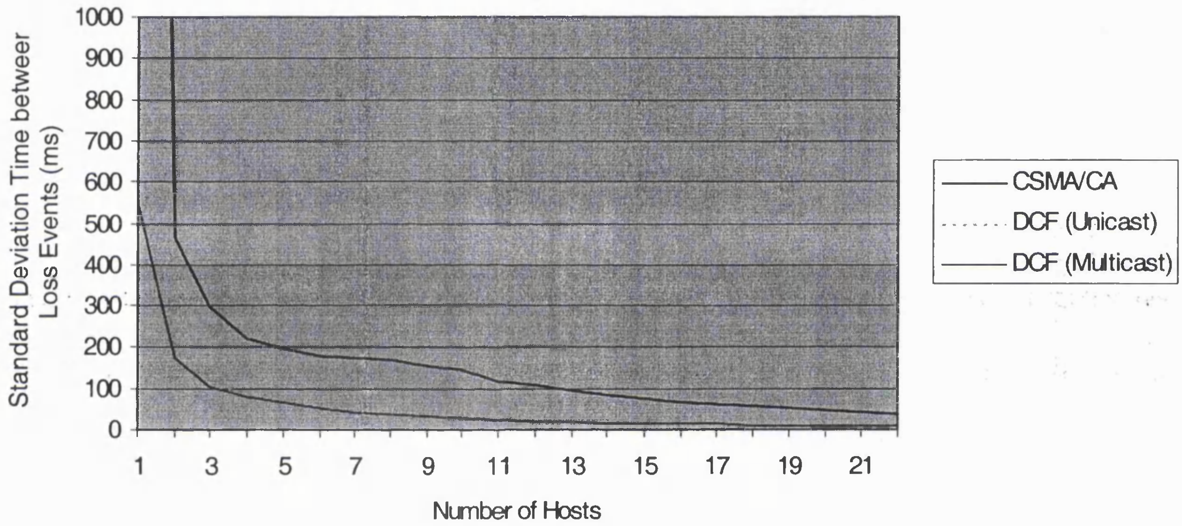
DCF unicast uses retransmission to recover from lost frames. Although these retransmissions occur rarely, the extra delay in receiving them affects the average delay. Since they also produce a few outlying values in the delay distribution, they also affect the delay standard deviation. Realtime applications have to add delay variation onto any buffering requirements to smooth out the effect of network jitter. So it is the sum of average delay and delay variance that are taken into account by realtime applications. Graph 2.2 shows that at low loads CSMA/CA has more delay variance than both DCF unicast and DCF multicast. This is due to the more consistent backoff times achieved by DCF under low load when collision loss is low. At high loads, however, DCF unicast retransmits colliding frames, which adversely affects delay variation leading to worse performance than CSMA/CA. DCF multicast does not retransmit and so diverges from DCF unicast maintaining low delay variance even at high load. After 25 runs, the delay standard deviation had converged to within 3%.

Graph 2.3: Frame Loss against Offered Load



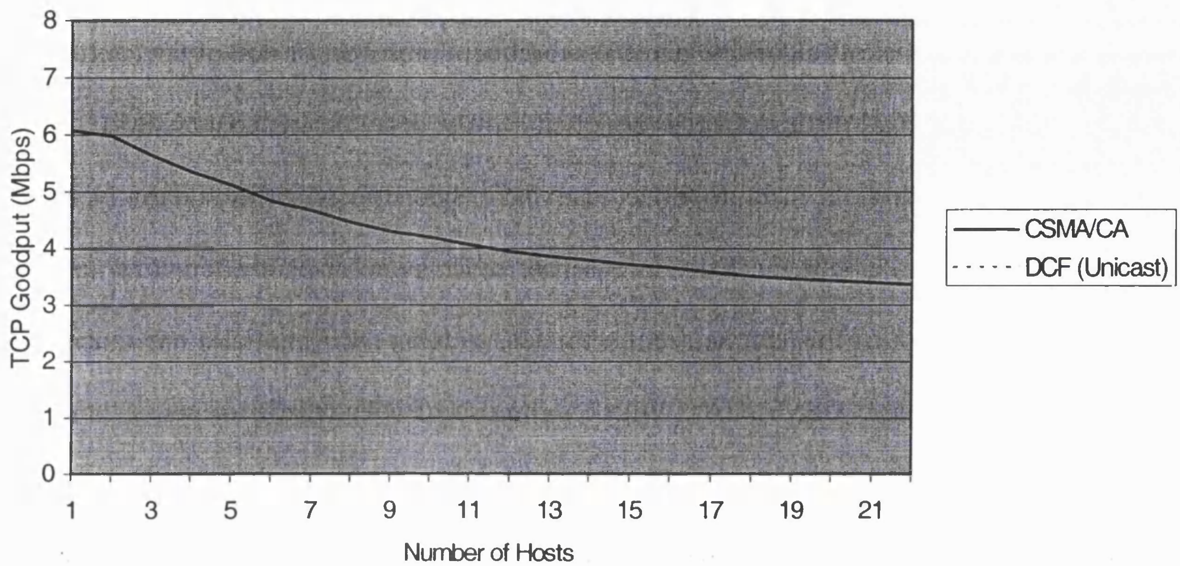
From graph 2.3 it can be seen that DCF multicast (without acknowledgement) has frame loss rates from about 6% to 32% over the load range. This loss rate explains the adoption of the link-layer acknowledgement protocol for unicast DCF transmissions; the loss rate is too high for TCP and realtime applications to tolerate. Graph 2.3 also shows the effect of adding the acknowledgement protocol: the loss rate is reduced to almost zero at low load values but deteriorates rapidly at high loads performing worse than CSMA/CA over approximately 60% load values. Therefore, at high loads the use of acknowledgements does not increase reliability. After 25 runs, the frame loss had converged to within 2%.

Graph 2.4: Variations from the Mean Time between Loss Events against Offered Load



The loss distribution measures the amount of clustering of lost frames, which is important to the performance of realtime applications. This is because the loss of consecutive frames has a greater impact on user perception of quality than frames lost in an evenly distributed manner. The standard deviation time from the mean time between loss events shows that CSMA/CA, DCF unicast and DCF multicast have similar low loss distributions. DCF unicast and DCF multicast have a less variable loss distribution than CSMA/CA which improves the performance of loss repair algorithms like redundancy for applications running over DCF. After 25 runs, the standard deviation time between loss events had converged to within 2%.

Graph 2.5: TCP Goodput against Offered Load



Graph 2.5 measures the effect of the MAC algorithm, the effect of differing numbers of sources and the effect of the CBR sources on the aggregate throughput of the TCP streams. The goodput is mainly affected by frame loss, so it is expected that CSMA/CA should not perform as well as DCF unicast, which has lower loss. The figures for goodput are high, considering the error rates at high loads, since frames which the link layer fails to resend, are themselves retransmitted at the transport layer, increasing the chance that the information eventually reaches the receiver. DCF multicast is not shown since TCP is always unicast and the effect on TCP throughput of changing the CBR streams to DCF multicast is negligible. After 25 runs, TCP throughput had converged to within 5%.

The results show that the choice of MAP does alter the characteristics of the network performance available to realtime streams. In particular, the frame loss due to DCF in the multicast case would be a problem for realtime multimedia applications. CSMA/CA and DCF are both within the non-deterministic category and are both in the same class of collision avoidance, carrier sense multiple access protocols. The only significant difference is the behaviour of the backoff algorithm and the subsequent presence, or not, of the capture effect. As expected, the removal of this effect is beneficial to the delay characteristics. Still, the choice of method for avoiding the capture effect can have side effects; other alternatives have been proposed [Weinmiller 96]. DCF is, in a sense, more aggressive in attempting to capture the medium. The result of this can be seen in the loss graphs; DCF multicast has much higher loss figures than the basic CSMA/CA. Whether the use of acknowledgements is the best way to provide a more reliable medium for realtime traffic is debatable. Other techniques, such as adding redundant information to subsequent frames, and multiple transmissions of the same frame may provide higher reliability whilst keeping the delay and delay variance figures at the desirable levels noted for the DCF protocol. Despite the loss levels, which can be overcome, the DCF variant of the CSMA/CA protocol has been shown to be a significant improvement for the transport of realtime data in a wireless, bandwidth limited environment.

2.19.2 Validation of Simulation Results

When simulating a complex protocol such as IEEE802.11, where many hosts are virtually independent but are running on the same physical machine, the possibility for error in either the simulation or the processing of results is high. To provide some assurance that the simulation results are accurate, a comparison with values from a real environment is necessary. Even though the real environment may be limited due to resource restrictions, confirming the results for a simple case greatly improves the probability that more complex simulation results are also correct.

One of the difficulties with validating simulation results is that it depends on the simulation writer and the hardware implementer having the same interpretation of the standard in question. A good simulation cannot be validated against a bad real implementation and thus anomalous results can sometimes be used to detect anomalies in real implementations. For example, some older Cisco 350 IEEE802.11 wireless cards that were tested were found to have very low loss in the multicast case. Further examination showed that duplicate frames were being received, leading to the conclusion that link-layer reliability was being used for multicast traffic. The standard states that link-layer acknowledgements should *not* be used for the multicast case, so these cards were not useful in validating any multicast simulations. In addition, the lack of ability to test the network without reliability prevents examination of the underlying performance of the network.

To test the network, traffic generator applications are needed. One generator is needed for TCP connections and another for CBR connections. Each generator includes both a client to produce data and a server to consume it. The consumer also measures network performance metrics and delivers the results. The CBR generator sends empty RTP packets of a specified size at a specified frequency whereas the TCP generator sends TCP data as long as the TCP layer is prepared to accept it, keeping the network fully occupied.

For the CBR connection, the important metrics are delay and loss. Measuring absolute delay requires clock synchronisation between the sender and receiver, preferably at sub-millisecond granularity. To achieve this requires the addition of the local time to each packet, and periodic updates of clocks with the receiver estimating the round-trip time in order to add this to the advertised time. These periodic clock updates must take place before the traffic generators start and not during the experiment as they themselves add traffic to the network and slightly change the results. Measurement of loss is simply a matter of adding a sequence number to each packet and the receiver checking continuity of sequence numbers and keeping a record of discontinuities. Out-of-order packets should not be included as loss. For the TCP connections the important metric is throughput and this is simply measured by the receiver as the number of bytes received over the period of time the experiment has been running.

To duplicate the simulation environment, two hosts were used with a CBR and a TCP connection transmitting in opposite directions (each host running one sender and one receiver). Adding more streams does not test the environment any further as the number of MAC protocols and hence the level of contention is still the same. After running the TCP connection for 15 seconds the CBR connection was started and both were allowed to run for a further 15 seconds. This also duplicates the simulation environment. Repetitions were made and the average taken until the average converged to within a 90% confidence interval. The position of the hosts relative to each other was found to be important. A difference of 10m was found to account for 0.5Mbps difference in TCP throughput.

For the unicast CBR case, frame loss was found to be 0.07%. This compares with 0.02% found from the simulator. In the multicast CBR case, frame loss was found to be 5.8%. In the simulator the value was 6.4%. For the TCP stream, the throughput was measured to be 5.04Mbps. The simulator produces 7.31Mbps. Delay figures were not obtained due to the inability to synchronise hosts to sub-millisecond granularity. With both hosts generating TCP streams and no CBR streams, the TCP throughput was measured to be 2.51Mbps and 2.45Mbps, with an aggregate of 4.96Mbps (down from 5.04Mbps for one host, a difference of 0.08Mbps). The simulator aggregate throughput with two TCP hosts was 7.25Mbps (down from 7.31Mbps, a difference of 0.06Mbps). These results are shown in tables 2.3 – 2.5.

	CBR Loss (%)	TCP Throughput (Mbps)
CBR Unicast	0.02	7.31
CBR Multicast	6.4	N/A

Table 2.3 – Simulator results for two hosts

	CBR Loss (%)	TCP Throughput (Mbps)
CBR Unicast	0.07	5.04
CBR Multicast	5.8	N/A

Table 2.4 – Real implementation results for two hosts

Number of TCP flows	Aggregate TCP Throughput (Mbps)
1	5.04
2	4.96

Table 2.5 – Real implementation TCP throughput results

The loss figures give validity to the simulation results, the discrepancy being of small magnitude.

The throughput figures, however, require explanation as the difference is over 2Mbps. Several factors could produce this result. Firstly, the simulation does not include the ad-hoc beaconing scheme described in the specification. This would not affect throughput of other streams significantly, though. Secondly, the simulation does not include power management, one feature of which is to control how often the receiver listens for transmissions. Adding this feature to the event based simulation model is non-trivial. This is likely to have a more detrimental effect on throughput and is the most likely cause of the discrepancy observed. The effect of adding TCP hosts, however, was a similar drop in aggregate throughput, increasing confidence in the simulation model.

This analysis has shown that loss rates for realtime multimedia applications transmitting multicast data on an IEEE802.11 wireless LAN will degrade the quality of service unless some addition measures are taken to improve the reliability of the network.

Chapter 3 - Hypothesis

3.1 Introduction

This chapter presents a complete hypothesis including sub-hypotheses and discusses proposals to investigate the sub-hypotheses. Simulation methodology is also established and conclusions are drawn.

3.2 Detailed Hypothesis

It is claimed that the performance of realtime multimedia applications on an ad-hoc IEEE802.11b wireless network can be improved by a cross-stack approach to reducing packet loss and delay. The cross-stack approach focuses on improvements at the MAC sub-layer, the data-link and application layers. Delay, delay variance, and loss are used to measure realtime application performance.

Realtime applications only operate within a bounded delay. Hard realtime applications fail when the delay bound is exceeded. Multimedia realtime applications are usually classified as soft realtime applications in that they do not fail when delay bounds are exceeded but degrade beyond user tolerance levels.

The described hypothesis includes the following sub-hypotheses:

1) Adaptations can be made to the multiple access protocol to improve realtime multimedia application performance.
2) A multicast link-layer acknowledgement scheme improves the performance of multicast realtime multimedia applications. It is not possible to simply use the existing unicast method for multicast traffic due to acknowledgement collisions and lack of knowledge of

the receiver's existence.

3) Application level redundancy can be used to improve realtime multimedia application performance. The performance of redundancy in comparison to the earlier two sub-hypotheses provides a full picture of the best method, or combination of methods to provide the best quality environment for realtime multimedia applications.

3.3 Detailed Sub-Hypotheses

3.3.1 Adaptations can be made to the multiple access protocol to improve realtime multimedia application performance.

There is potential for development of a new MAC protocol as an evolution of DCF but drawing on the methods proposed in BLAM, to improve the performance of realtime multimedia applications. Decreasing access delay and frame loss can substantially improve the quality of service experienced by the end user. It has already been mentioned that the removal of the capture effect in DFWMAC improves fairness and hence reduces access delay times. Further alterations to the DCF backoff algorithm may also decrease delay or loss. In particular, the high loss observed in the analysis of DCF in the multicast mode should be addressed. A higher initial contention window, for example, should lower frame loss but will also increase access delay. Conversely, applying logarithmic arbitration should lead to lower access delay but also higher loss. A combination of these techniques could result in lower loss and lower delay.

Changes cannot be made to real implementations of MAC protocols, as they are proprietary and located in firmware and therefore not accessible. While source code for wireless LAN drivers is available from open source operating systems it typically implements higher levels of the protocol stack, management layers and standard application programming interfaces (APIs). Also, measuring absolute delay in a real environment is not easy. Scaling the environment up to many hosts requires expensive resources. Therefore, simulation had to be used to test this sub-hypothesis.

The simulation environment was the same as in section 2.9.1. The DCF protocol (with unicast and multicast CBR connections) was used as the base against which the sub-hypothesis was measured.

3.3.2 A multicast link-layer acknowledgement scheme improves the performance of realtime multicast applications.

The analysis of DCF showed the low performance of the multicast mode. DCF improves reliability in the unicast case by acknowledging data frames and retransmitting colliding frames. Although the unicast reliability scheme cannot be directly applied to multicast, a variation may provide improvements in frame loss at the expense of a small increase in access delay.

For the same reasons as above, simulation was used to measure this sub-hypothesis. Using the same environment also allows comparison of results between sub-hypotheses. The results were compared with DCF in reliable unicast mode and unreliable multicast mode.

3.3.3 Application level redundancy can be used to improve realtime multimedia application performance.

Redundancy in the most general sense means transmitting the same information (possibly transformed) more than once. It differs from forward error correction (which also uses redundant information) in that application level redundancy can be thought of as “inter-packet” whereas FEC is “intra-packet”. In other words, FEC protects the contents of a packet from corruption by allowing bits within a packet to be reconstructed if they get corrupted (within limits). Application level redundancy allows a packet to be recovered by using redundant information held within another packet.

This technique has the primary aim of reducing loss but does so with very little increase in delay, the trade-off in this case being decreased throughput. Retransmissions, on the other hand, trade-off loss with possibly larger increases in delay, especially on an end-to-end basis. Being applied at the application level means only realtime applications with a need for better reliability will use redundancy. For applications with a low bandwidth, redundancy can be applied over the relatively wideband wireless LAN link without consuming too much bandwidth from other applications. In limited bandwidth environments such as dial-up 56kbps links, however, the use of redundancy may be inappropriate for the application to use, the resulting delay and loss being worse than if redundancy were not used.

The same simulation environment was used to provide a basis for comparison with the other two sub-hypothesis. Application level redundancy used multicast DCF at the link-layer in order to avoid any reliability mechanisms that are not necessary for this sub-hypothesis.

3.4 Conclusions

This chapter has introduced the detailed hypothesis and three detailed sub-hypotheses. Although each sub-hypothesis can improve realtime performance, the possible trade-offs between delay, loss and throughput make the comparison between the sub-hypotheses important in deciding which provides the best improvement in realtime application performance.

Chapter 4 – Multiple Access Protocol Modification

4.1 Introduction

This chapter describes a new multiple access protocol called DCF/LA based on the DCF protocol and the BLAM protocol. The protocol is simulated and the results presented. The performance of the protocol and any improvements relative to DCF are discussed in the conclusions.

4.2 DCF/LA: Distributed Coordination Function / Logarithmic Arbitration

DCF/LA is a protocol based on DCF with an additional backoff procedure based on BLAM (Binary Logarithmic Arbitration Method). It is designed to replace DCF and offer greater performance primarily for realtime but also for non-realtime traffic.

BLAM is a protocol designed to improve the CSMA/CD IEEE802.3 protocol. The specification describes a number of alterations to the basic CSMA/CD protocol, some of which are dependent on the wired medium. The backoff method, however, can be applied in a wireless environment. One particular aspect of BLAM's backoff method is important: logarithmic arbitration. It offers reductions in access delay by reducing the amount of time spent by hosts contending for the medium. Logarithmic arbitration works by halving the size of the contention window if two slots are detected to be idle. Hosts are able to converge on a transmission slot by being better able to estimate the amount of competition, ie the number of other hosts waiting to transmit at that particular time.

DCF/LA maintains the original behaviour of DCF in the event of sensing a busy medium during backoff: the host must freeze the backoff counter until the medium becomes free again. Figure 4.1 shows the detailed operation of the DCF/LA protocol.

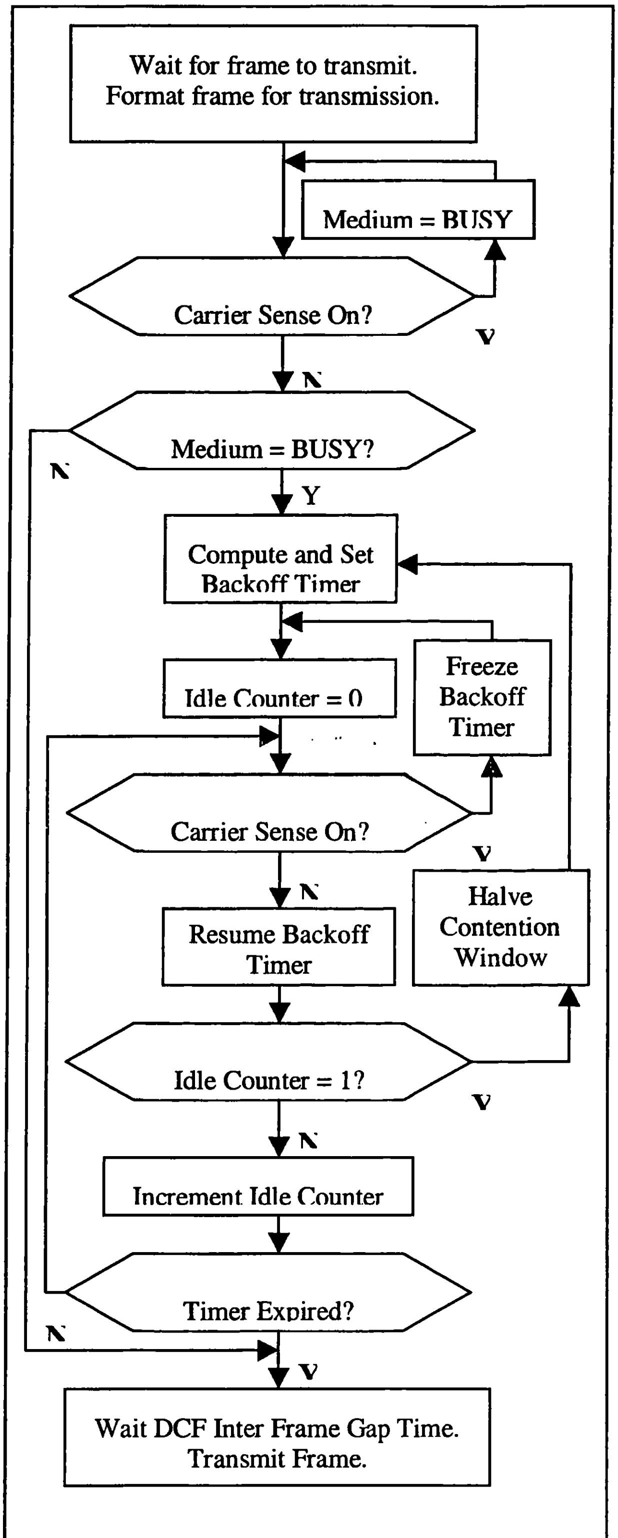


Figure 4.1: DCF/LA Transmit

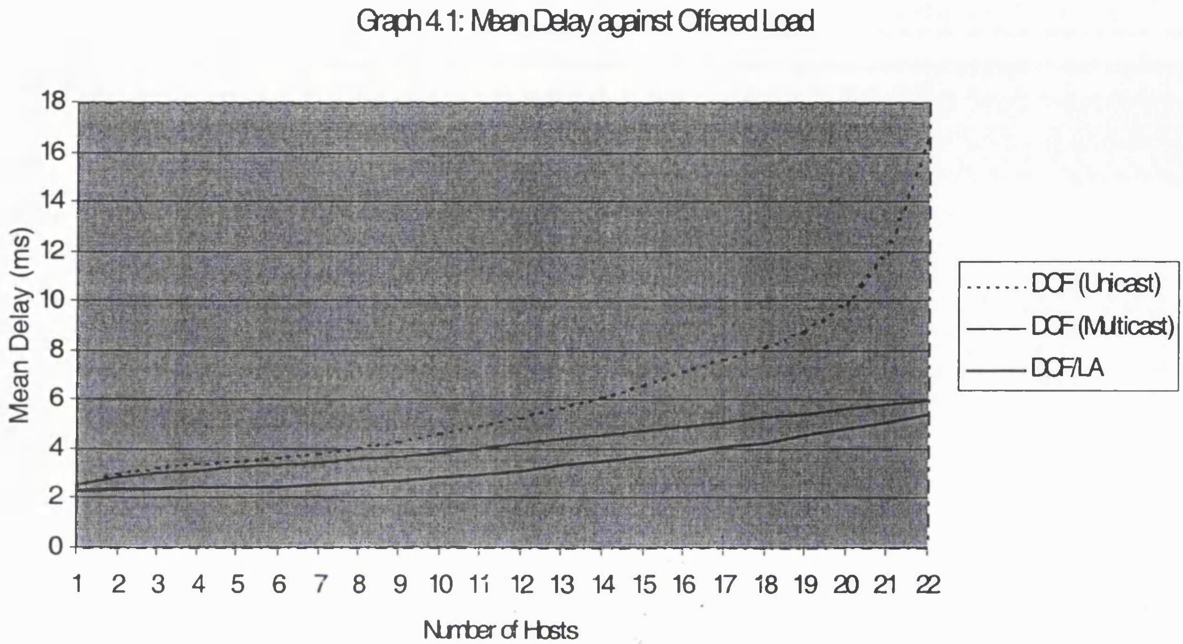
Compared to CSMA/CA, DCF increases the amount of carrier sensing in order to detect other transmissions during backoff. This gives important information regarding the current state of the medium, but requires extra power which is a limited resource for many wireless devices. Thus a tradeoff exists between the amount of carrier sensing and power consumption. Carrier sensing during backoff can provide important information to a multiple access protocol with only a limited increase in power usage. This is used by the DCF protocol to avoid further collisions and improve fairness. BLAM also requires carrier sensing during backoff to determine the amount of competition and dynamically alter the backoff times accordingly. DCF/LA, therefore, requires no more carrier sensing than DCF and hence requires no extra power, but makes more use of the information in order to estimate current network load and hence reduce delay and loss.

Logarithmic Arbitration reduces the backoff delay reducing average delay and delay variance figures but increasing frame loss. This trade-off can be rebalanced by adjusting the "contention window". The contention window refers to the range of values from which a random backoff time is chosen. This is set to a recommended minimum value of 16, and doubled upon timeout when waiting for an acknowledgement (exponential increase). For DCF/LA no acknowledgements are used, so the minimum value is the only value to which the contention window will be set (exponential increase is not used). By increasing the size of the minimum contention window, delay should increase while loss and throughput should decrease. Increasing the contention window by large amounts is not a problem since the logarithmic behaviour of DCF/LA reduces the window very rapidly. A value of 512 has been used in the following simulations.

4.3 Simulation Methodology

The simulation environment was the same as in section 2.9.1. DCF was used as the protocol of comparison, in both unicast and multicast modes. Changes were made to the backoff algorithm of DCF to create the logarithmic arbitration behaviour. Link-layer retransmissions were disabled, as these are not part of DCF/LA, so all data packets were unacknowledged. This applied to the TCP streams, which did their own retransmissions, as well as the CBR streams. The contention window was increased to 512 as described above.

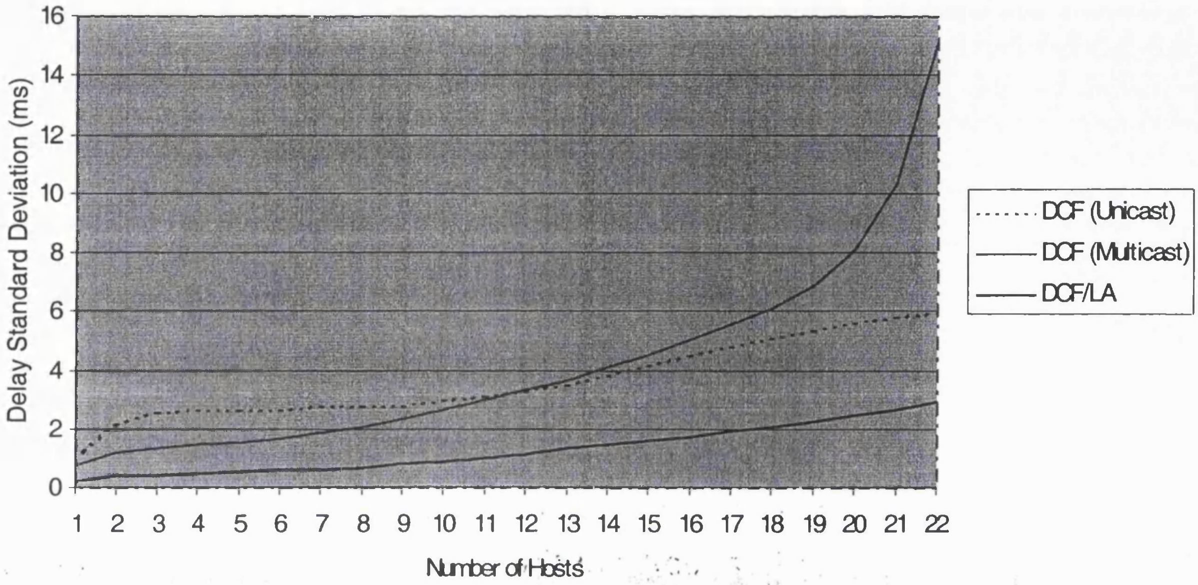
4.4 Simulation Results



Graph 4.1 shows the mean average delay times in milliseconds for various numbers of sources. As the number of sources increases the amount of contention increases, backoff periods increase and hence average delay increases.

Generally, DCF/LA produces better performance than DCF especially at higher loads. On a wider scale, DCF/LA produces results of the same order as DCF, which is expected, given that the results are almost as low as can be achieved with propagation and transmission delays taken into account. DCF and DCF/LA delay times are suitable for realtime applications at all tested load levels.

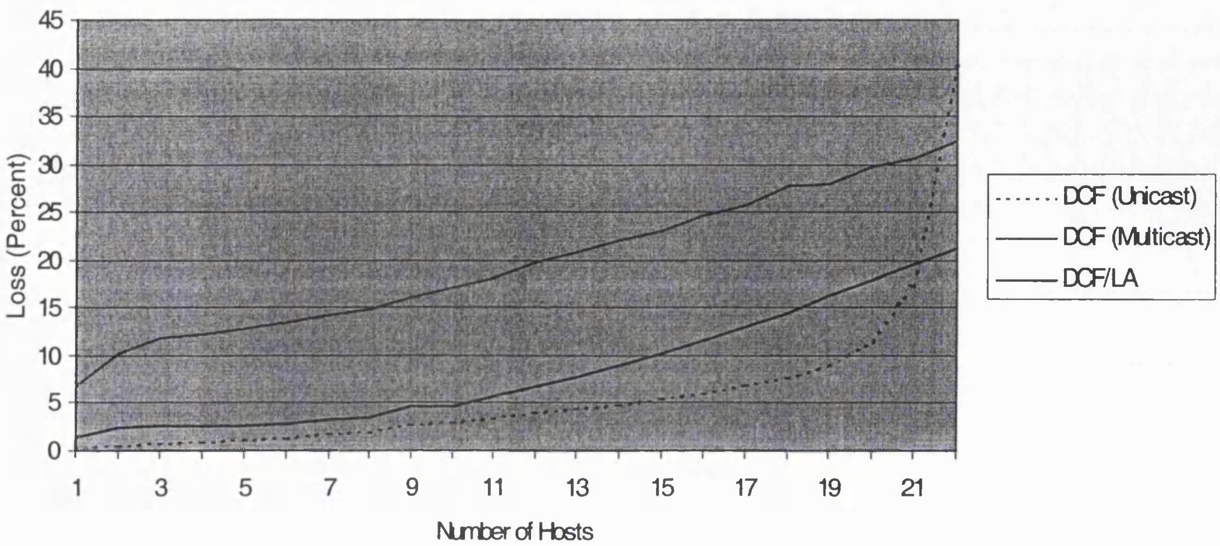
Graph 4.2: Delay Standard Deviation against Offered Load



Graph 4.2 shows the standard deviation of the access delay as the load increases. The amount of variation in the delay is as important as the actual delay for realtime applications that have to provision buffer space to smooth the effect of delay variance. The addition of the average delay and the delay standard deviation can be used to give an estimate of the total delay.

DCF/LA delay variance is better than DCF over the entire load range. At high loads the difference in delay variation exceeds 10ms. The more rapid convergence on a transmission slot due to logarithmic arbitration in DCF/LA accounts for this result.

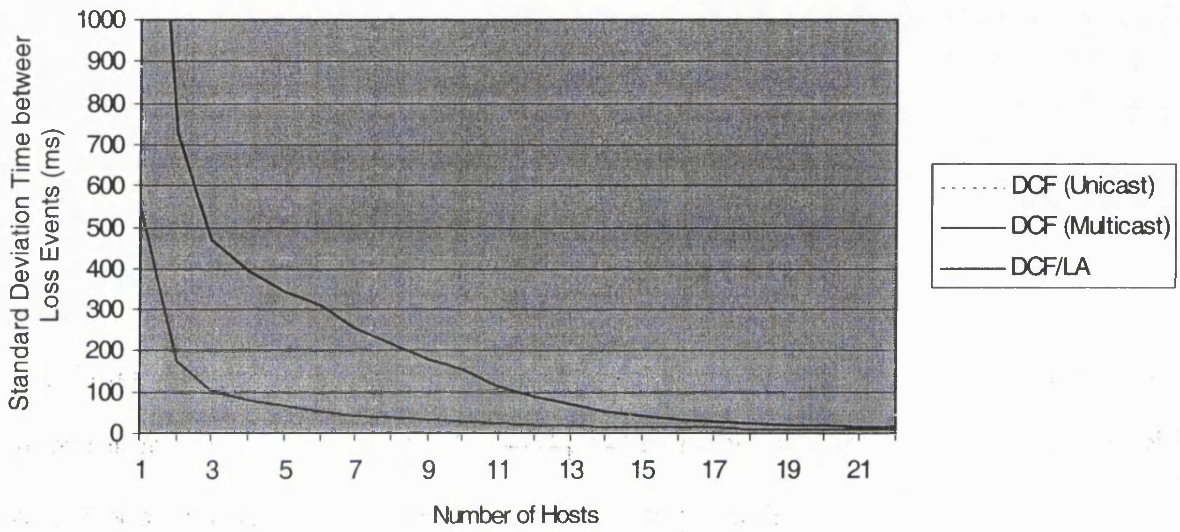
Graph 4.3: Frame Loss against Offered Load



Frame loss is a fundamental metric for a MAC protocol. It not only determines the throughput for TCP but also the performance of both interactive and non-interactive realtime multimedia applications. Graph 4.3 compares loss figures at increasing network load. A multiple access protocol may be able to control loss over all load values, but if not will employ retransmission schemes to compensate for high loss.

DCF multicast does not use any retransmission scheme and so has loss between about 7% and 33%. Even with loss repair and concealing techniques, these values will be noticeable to the user of a realtime application, and depending on the user, may well not be acceptable. For DCF unicast, retransmissions are used to compensate and reduce loss over most of the range. High high load, however, DCF unicast performs worse than DCF multicast. DCF/LA controls loss over the load range, almost matching DCF unicast and substantially improving upon DCF multicast, without the need for retransmission. At very high load DCF/LA produces the lowest loss results.

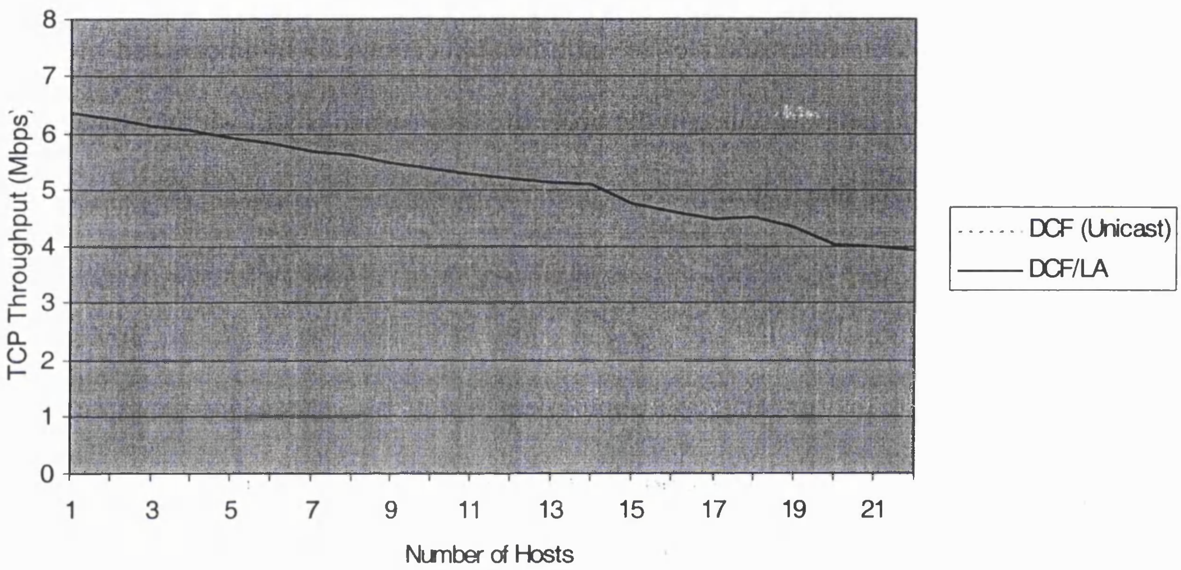
Graph 4.4: Variations from the Mean Time between Loss Events against Offered Load



Percentage frame loss, whilst important, only shows the percentage of lost frames against transmitted frames. It does not give any detail about the loss distribution that is vital to the quality of the output of many realtime multimedia applications. Generally, the less clustered lost frames are to each other the better the application will perform. For example, an audio application may be able to interpolate the contents of a lost frame from the contents of the previous and next frames. It can easily be seen that if two consecutive frames are lost this technique cannot conceal loss so well as the "gap" to be interpolated is wider. Graph 4.4 measures the loss standard deviation by recording the variations around the mean average time between lost frames. At low load, loss rates are low and variations are more erratic, but as the load, and hence loss, increases a low variance is a measure of equally distributed lost frames.

DCF/LA has more variation than DCF multicast and DCF unicast. However, at high load levels, where loss rates are significant, variations about the mean are low, indicating that loss events are well spaced which allows realtime multimedia applications to recover from loss.

Graph 4.5: TCP Throughput against Offered Load



Graph 4.5 compares DCF unicast (which includes link-layer acknowledgements) with DCF/LA in terms of TCP throughput. DCF multicast is not shown since TCP only operates over unicast links. The effect of link-layer acknowledgements on throughput can be seen; DCF unicast gives better TCP throughput at low load levels. At high load levels, though, the efficiency of DCF retransmission technique decreases and so the TCP throughput of DCF/LA is greater than that of DCF.

4.5 Conclusions

The results show that the choice of multiple access protocol does have an impact on network performance and consequently influences the suitability of the network for realtime applications. DCF and DCF/LA are within the non-deterministic category and belong to the same class of collision avoidance, carrier sense multiple access protocols.

The DCF/LA delay figures (both average and variance) are very low, and remain constant across all load levels making them suitable for realtime applications. DCF unicast and multicast have combined delay and delay variance in the order of 30ms (which is still within typical application bounds), whereas DCF has 10ms.

The frame loss figures for DCF are high especially at high load values (about 30% to 40% loss) and exceed realtime multimedia application tolerance levels. By using carrier sense information to better use, DCF/LA exhibits low loss (up to 20% which is within realtime application bounds when loss repair algorithms are used) without requiring additional error control mechanisms.

DCF/LA shows it is possible to retain the low loss figures obtained with DCF unicast whilst producing delay rates better than DCF over the entire load range.

Chapter 5 – Multicast link-layer acknowledgement

5.1 Introduction

First generation wireless LAN products had no collision detection. Collision detection requires a host to "listen" while transmitting; wireless transmissions are so much stronger than received signals, and background wireless noise is so common that the host cannot reliably "hear" any other transmissions while transmitting. Instead, collision avoidance was used which applied more effort in avoiding collisions in the first place, but at high load the loss rate due to collisions still escalated. It was left to higher layer transport protocols to retransmit lost packets. It was found, however, that higher layer transport protocols recover slowly from such errors due to long timeouts before retransmissions take place, dramatically reducing throughput and thus wasting valuable wireless bandwidth. As a result, second generation wireless LAN products, conforming to the IEEE802.11 standard, incorporate collision detection in the form of acknowledgement and retransmission. While increasing link performance, such retransmissions do not always help end-to-end efficiency. This is because TCP measures round trip time (RTT) and uses it to adjust its timeout period; link layer retransmissions increase the RTT leading to longer TCP timeouts. So although packets lost on the wireless link are recovered quickly, packets lost in routers or on other unreliable links may take longer for TCP to recover from than if the wireless link layer did not perform retransmissions itself. However, the benefit of fast link-layer wireless error recovery was seen to outweigh any reduction in transport protocol performance.

It is important to differentiate between collision detection and error control. For wireless LANs corruption errors are reduced as far as possible by FEC (Forward Error Correction). Collisions, being an almost complete overlap of two or more packets, are not protected by FEC. In the absence of physical collision detection, lack of receiver acknowledgement is the only way to detect a collision. Given the use of FEC, an assumption is made that lack of an acknowledgement represents a collision and not frame corruption. This is similar to the TCP assumption that loss is due to congestion and not corruption error. (While ECN allows TCP to differentiate between congestion and loss when loss is due to router queue overflow, no such similar way to distinguish between collision and corruption currently exists for Wireless LANs.)

The response of TCP to packet loss is to retransmit according to an exponential backoff scheme (the period between retransmissions doubles on each subsequent attempt). This is to prevent retransmissions themselves from causing further congestion. DCF (and other link-layer protocols) also use exponential backoff, since collisions are an indication of congestion.

The IEEE802.11 specification states that acknowledgements are used for unicast transmission only, multicast transmissions have no further protection against packet loss. Although many multicast applications can tolerate a certain amount of loss, the tolerance is limited and depends on the user. When loss is excessive, either the application or the network or both must take action to reduce the loss rate for the application to function.

Delay sensitive realtime applications include multimedia applications (ie include audio, video, etc) which can be interactive (eg telephony) or not (eg television). Although interactive applications can be multiparty (eg voice conferences), the class of non-interactive applications is most likely to have more than one receiver within a single physical network. Since the analysis of DCF showed the high multicast loss rates, any reduction in this will increase the quality of the output from these multicast applications.

IEEE802.11

5.2 *Considerations for a Multicast Collision Detection Protocol*

Multicast transmission collision detection is not a simple extension of the unicast case. If unicast error control is used without amendment for multicast transmissions the following problems emerge:

1. Receiver acknowledgements will almost certainly collide. To quote IEEE 802.11-1999 "After a successful reception of a frame requiring acknowledgement, transmission of the ACK frame shall commence after a SIFS (Short Inter-frame Space) period, without regard to the busy/idle state of the medium."

2. If there are no receivers in a group, there will be no acknowledgements. If there are no acknowledgements, the sender will wrongly assume that the data frame has been lost, will go into exponential backoff resending the data frame repeatedly up to the retransmission limit, delaying other queued frames and wasting wireless bandwidth.

It can be seen that, for collision detection, acknowledgements from every recipient are not required. It is only necessary for at least one recipient to produce an acknowledgement for the sender to discover that the data frame did not collide with another. This is important as it removes the requirement for the sender to maintain a list of every recipient in a group. For an example of the complexities involved in maintaining an accurate list of hosts see the description of "Token Bus" in section 2.9.2.

It is obvious that with multiple recipients a method of scheduling acknowledgements is required to reduce the risk of collision. For example, the same CSMA/CA protocol used for data frames could be reused to schedule acknowledgements. This results in less predictability regarding the time at which acknowledgements will be received and hence sender retransmission timers have to be substantially longer. This, in turn, increases the delay, decreases the throughput and has to be taken into account when trading off delay, loss and throughput. As acknowledgements are not themselves acknowledged, colliding acknowledgements are lost and the sender will retransmit the data. The probability of this is offset by the fact that with multiple receivers even if two acknowledgements collide another may be received successfully. This works because although all receivers are listening for acknowledgement frames, they will not hear two colliding frames, and so will send a frame themselves. There must be at least three receivers of a group for this to work.

Hosts should take into account the multicast acknowledgement protocol even if they are out of range of any of the receivers. Hosts hearing a transmission with a multicast destination should allow extra backoff time to account for the contest and transmission of an acknowledgement. This should protect acknowledgements from other host's data transmissions, extending the unicast case. The extra backoff time must take into account the longest possible backoff case, although the backoff may be shortened if the host receives an acknowledgement.

Multicast transmission is often associated with unreliable transport protocols and real-time payloads. In these cases it is common to consider excessively delayed packets as effectively useless and a small percentage of loss bearable. It is therefore important to consider the trade-off between delay and loss when deciding the value of acknowledgement timeouts and retransmission limits.

It has been shown earlier, from the simulation of DCF, that under heavy load, the loss rate for unacknowledged multicast traffic on an IEEE802.11 network exceeds 15 percent. Under these conditions it would seem preferable to trade some delay budget in order to reduce loss.

5.3 A Protocol for Multicast Collision Detection

The rules for the proposed multicast collision detection protocol are as follows:

1. Transmission of the ACK frame following a multicast transmission should observe the normal rules of CSMA/CA except that the STA (Station) goes into backoff regardless of the initial state of the medium. In addition, if other data receivers in the same group detect another STA has sent an ACK for this data frame, it shall suppress its own pending transmission, and return to an idle state.

2. Extra backoff time should be allowed by all receivers of multicast data messages who are non-members of the group. Receivers should set the backoff time taking into account the longest possible backoff timer that an acknowledging STA may choose. This prevents

other data transmissions contending with acknowledgements, reserving the period following data transmission for acknowledgement contention.

Group membership is maintained with a link layer version of the IGMP (Internet Group Management Protocol) [Deering 89] protocol. Receivers periodically broadcast "join" messages if group membership is requested from a higher layer. Receivers suppress join messages if a join message for the same group is received from another receiver. Receivers broadcast a "leave" message if a higher layer indicates this. Senders record join requests, and only transmit if the group has members.

This particular feature of the design is not preferred but is necessary to prevent multicast senders consuming bandwidth from other sources especially on the same host. It is not preferred because building higher layer information into a link-layer is a layer violation and does not scale to all current higher layer protocols or future higher layer protocols. A preferable solution would to perform the same check at the IP level, ie for IP multicast senders to require group membership before transmitting.

Hosts receiving broadcast messages should follow the same acknowledgement protocol. Multicast group considerations are not important in this case as all hosts are required to take part in the acknowledgement protocol. Therefore, the case where the host transmits but no acknowledgments are received means that no other hosts are within range and therefore no bandwidth is being wasted.

Figures 5.1 and 5.2 show flowcharts for hosts receiving and sending multicast data. The method by which hosts discover whether groups have members is not shown, but requires link-layers to "snoop" on higher level messages looking for join/leave messages and update state accordingly.

Fig. 5.1: Receiving Packets

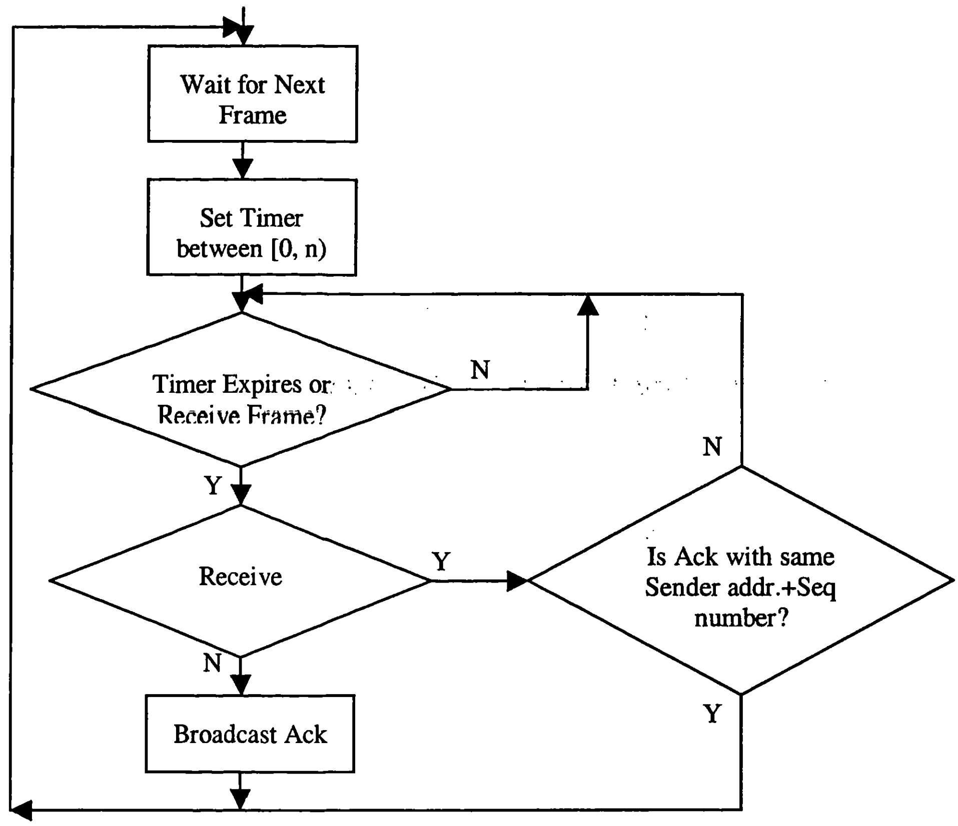
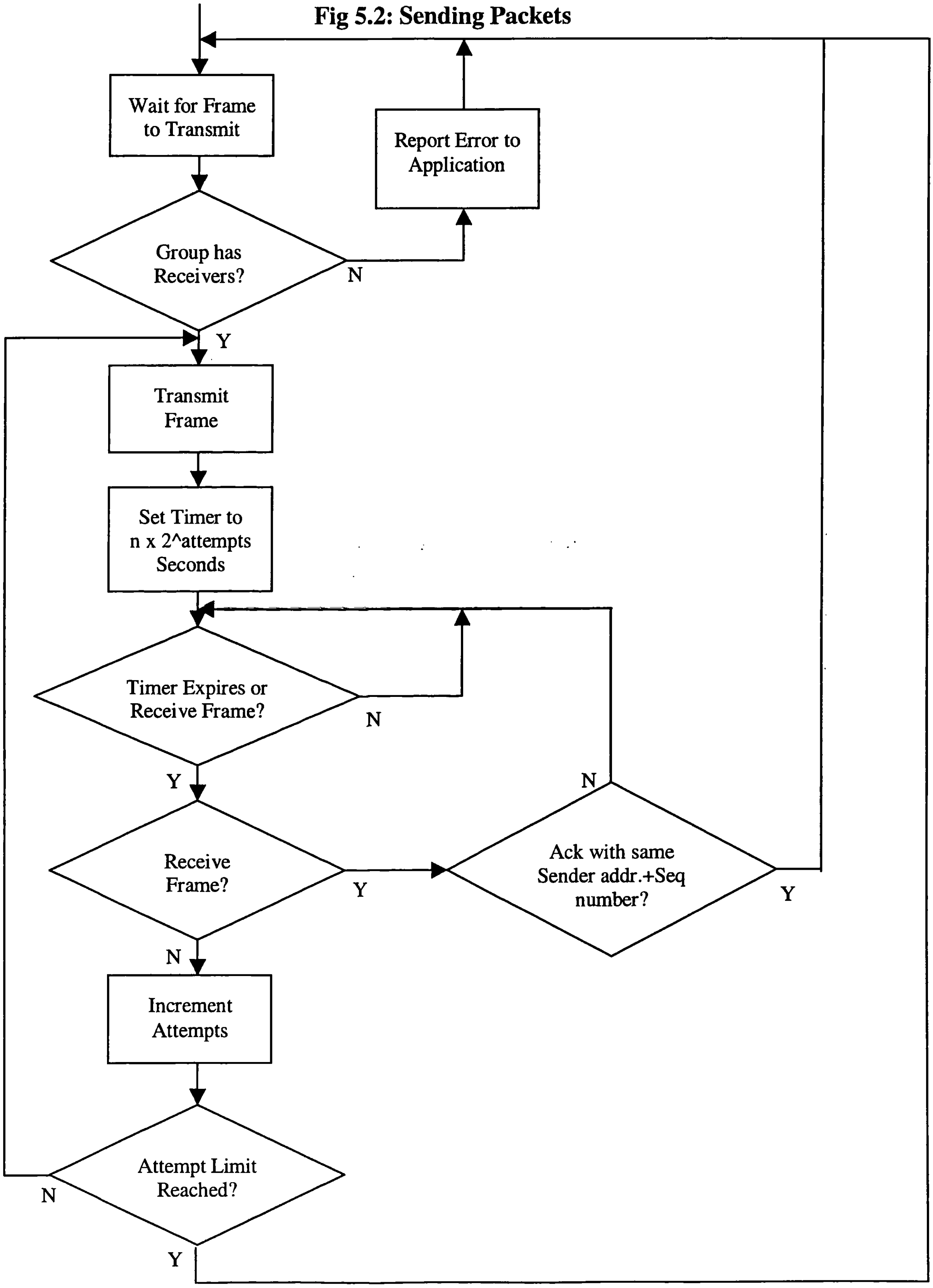


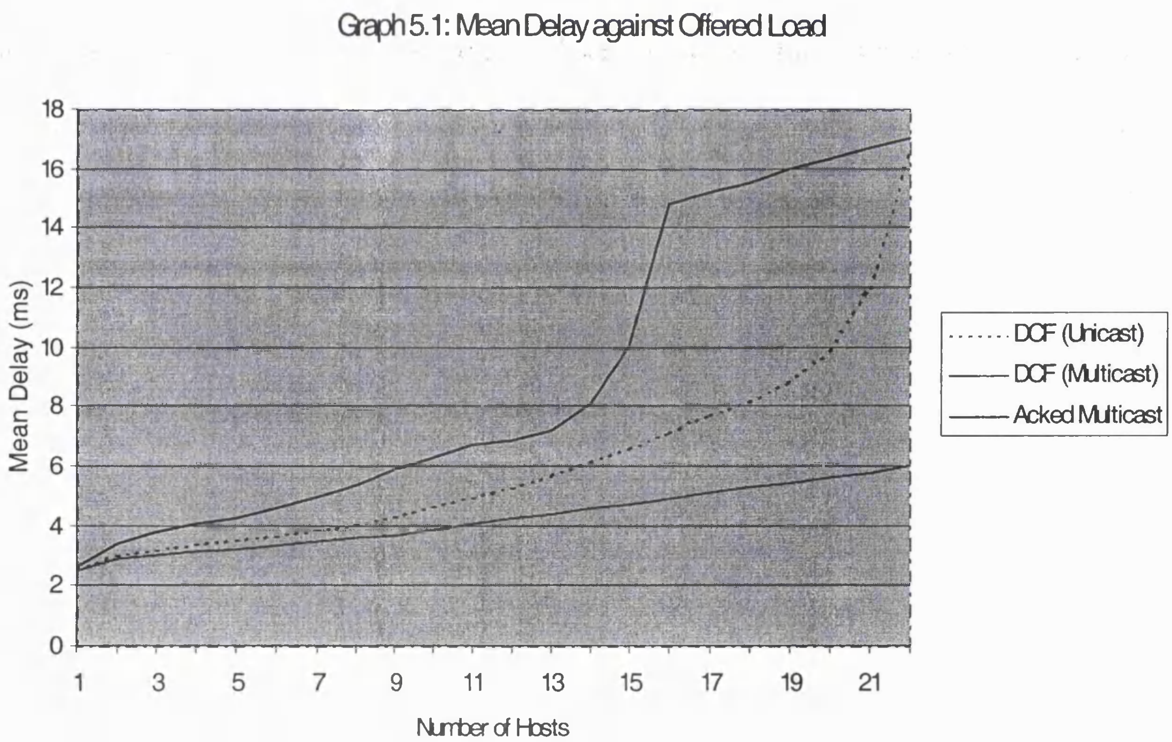
Fig 5.2: Sending Packets



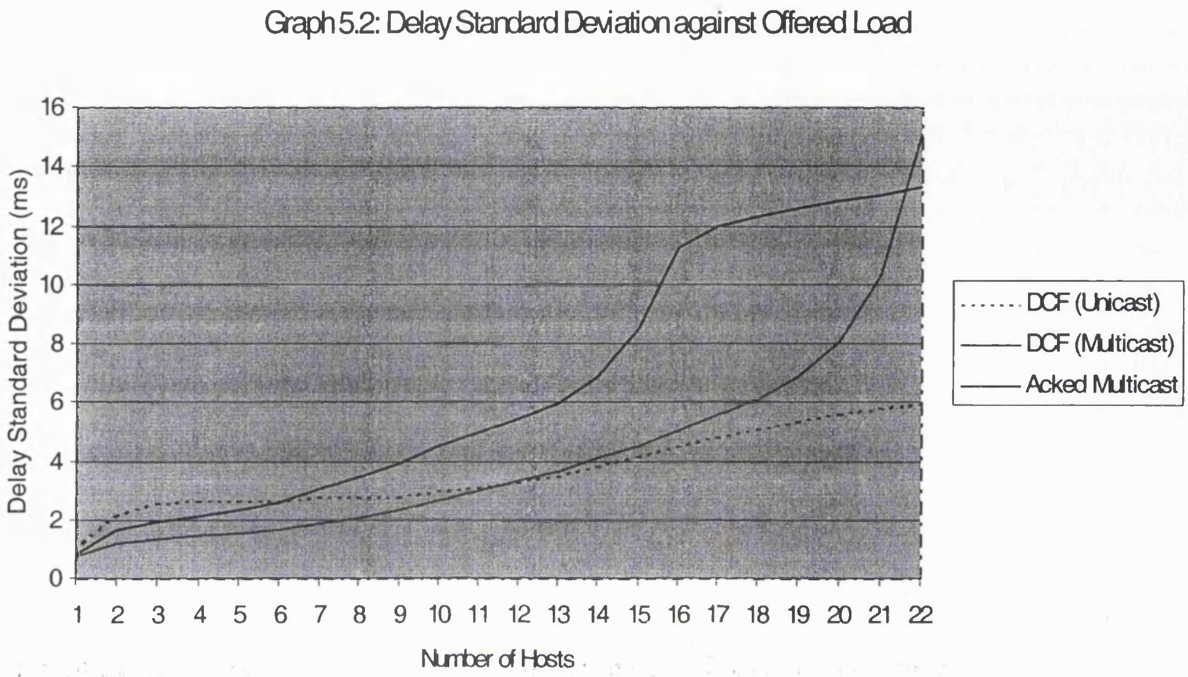
5.4 Simulation Methodology

The simulation environment was the same as in section 2.9.1. DCF was used as the protocol of comparison, in both unicast and multicast modes. DCF unicast TCP traffic continued to use the normal DCF retransmission protocol, while the multicast CBR traffic used the newly proposed retransmission protocol. All CBR hosts were members of all multicast groups. A comparison with DCF/LA and Application level redundancy will be made in chapter 7.

5.5 Simulation results

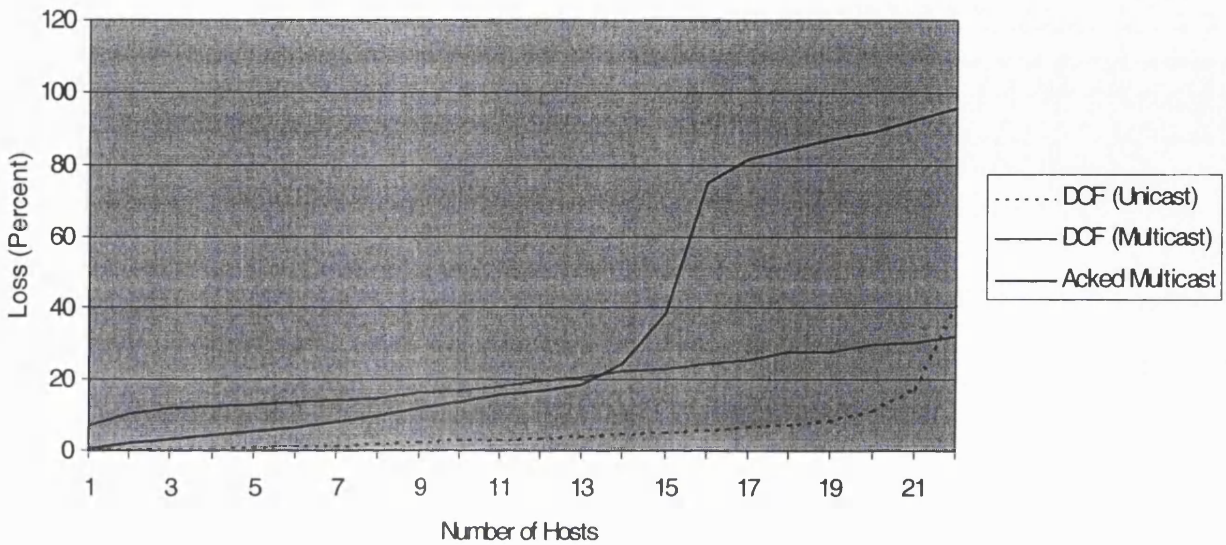


It has already been seen that DCF unicast average delay is greater than DCF multicast (without acknowledgements) due to the overhead of retransmission of colliding frames. DCF unicast employs a similar retransmission scheme to the proposed acknowledged multicast protocol, except that acknowledgements are given priority in the unicast case. The effect of this can be seen in graph 5.1, which shows that the acknowledged multicast frames take slightly longer, on average, to reach the receiver. This does not mean that correctly received frames take any longer because the data is passed up to higher layers before an acknowledgement is sent back, so these frames have the same delay as before. The difference is for those frames that collide and have to be retransmitted. It is these few frames with much longer delay that increase the average. The overall difference in average delay is small, however, frames taking a few milliseconds longer when the network is heavily load. It is very unlikely that this would affect the user experience of any of the realtime applications under consideration. It should be noted that with the multicast acknowledgement protocol the network becomes fully loaded when 15 hosts are simultaneously transmitting due to the extra delay associated with the staggered acknowledgement scheme. At this load level, the delay rises in a more linear fashion as the number of hosts increases.



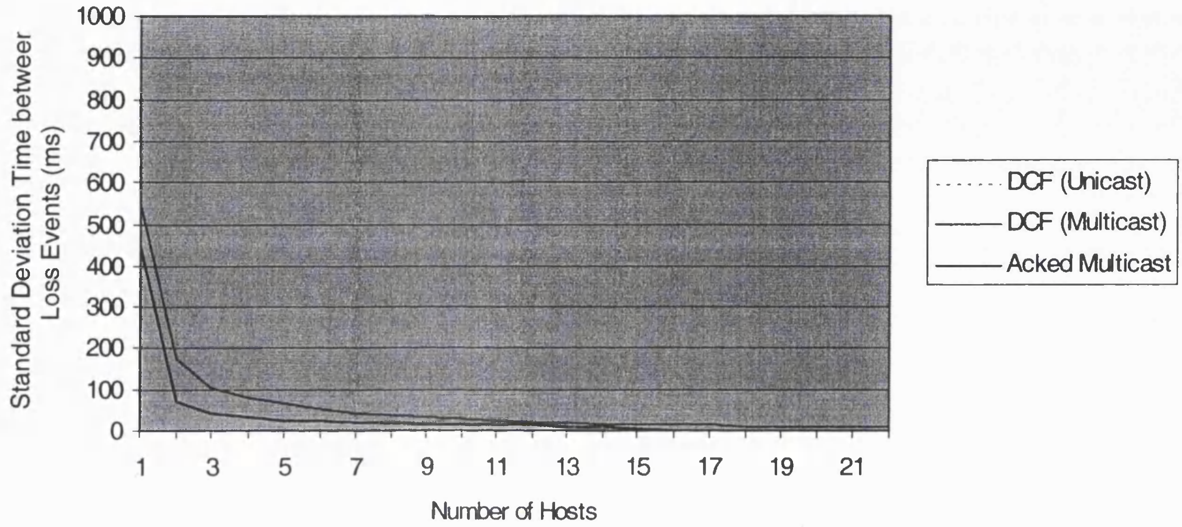
The delay standard deviation is also expected to rise when any acknowledgement protocol is used, and this has been seen with DCF unicast where the delay standard deviation was slightly higher than DCF multicast (at high loads as the retransmission scheme breaks down the delay variance stabilises though). Due to the nature of the acknowledged multicast protocol, as expected, the delay standard deviation is again slightly higher but of a small order of magnitude. Taken in combination, the rises in both average delay and delay variation would still not affect the performance of a realtime application in any significant manner.

Graph 5.3: Frame Loss against Offered Load



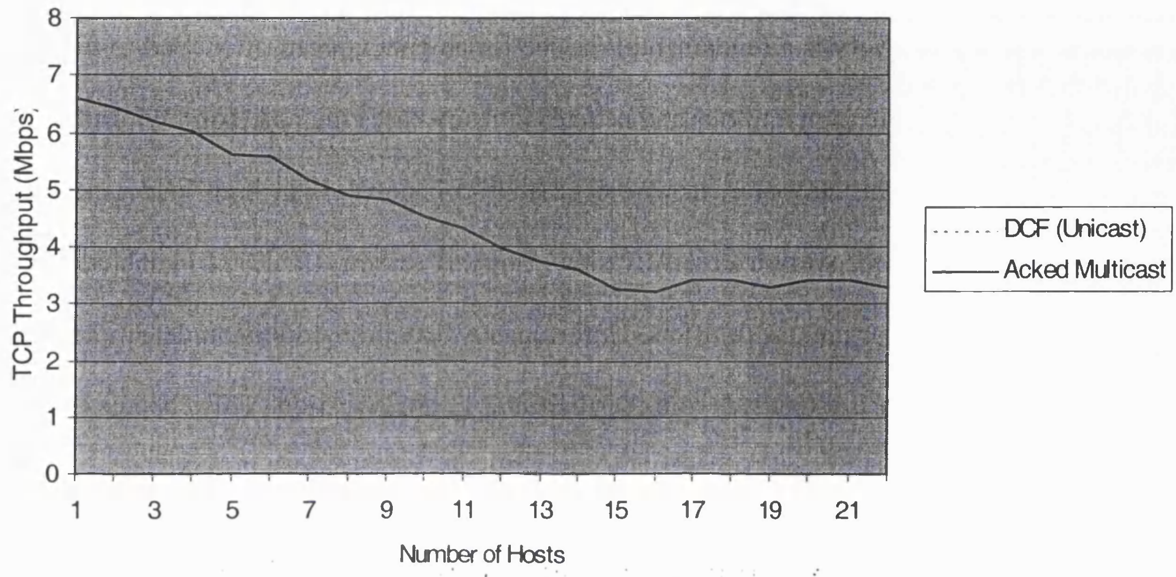
The frame loss of DCF multicast (without acknowledgement) coupled with the effect of frame loss on realtime applications motivates the design of the acknowledged multicast protocol. The reduction in loss seen for DCF unicast is partly produced by the priority given to acknowledgement frames that greatly reduces the probability of a collision of these frames with other data frames. Also, more importantly, unicast acknowledgements do not have to compete with other acknowledgements for the same data frame. Although the multicast acknowledgement protocol does include extra reliability of acknowledgement frames by other receivers sending an acknowledgement if two other acknowledgements collide, it still does not have the reliability of the unicast protocol. The results in graph 5.3 show, however, that improvements in frame loss are still significant at low load. The reduced level at which the multicast acknowledged frames reach full load, however, makes the scheme perform very poorly over the high load range of the graph compared to DCF.

Graph 5.4: Variations from the Mean Time between Loss Events against Offered Load



The loss distribution is important for realtime applications as, for example, consecutive frame losses can be more damaging than equally spaced losses. The acknowledged multicast protocol has a slightly better loss distribution than DCF maintaining even loss distributions across the load range. This, combined with the low frame loss figures at low load levels, would allow a realtime multimedia application to conceal data loss in an effective manner. At high loss levels, the even distribution of loss events may enable the application to recover from most loss events improving the quality.

Graph 5.5: TCP Throughput against Offered Load



Adding a retransmission protocol to realtime multicast traffic should have an effect on the aggregate throughput of TCP traffic. Retransmissions take a small amount of extra bandwidth that is then unavailable for TCP streams. The magnitude of the loss of throughput can be seen in graph 5.5. A loss of between about 1Mbps can be seen which although affecting the TCP streams would not significantly affect the user's experience of such applications. At very high loads, the inefficiency of the DCF unicast retransmission scheme allows the acknowledged multicast technique to achieve the same TCP throughput.

5.6 Conclusions

Reliable packet delivery is a fundamental service for any network to offer. End-to-end reliability in the form of TCP/IP is ubiquitous and negates in many cases the need for any further link-layer reliability mechanism. However, in wireless LANs collision loss at high load can reach levels that render TCP/IP recovery excessively slow. For this reason, IEEE802.11 includes link-layer collision detection for unicast transfers. Unfortunately, extending collision detection to multicast is not trivial, and so is excluded from the IEEE802.11 specification. This chapter examined the problems, suggested solutions, and proposed protocol extensions to provide multicast collision detection in IEEE802.11.

By scheduling acknowledgements with CSMA/CA with the provision of always going into backoff before transmission, and suppressing acknowledgements if another acknowledgement for the same data frame is heard, multicast data can receive the same retransmission strategy as unicast data does under the IEEE802.11 protocol. Receivers extending their backoff time to allow for multicast acknowledgment contests, and senders suppressing output if there are no members in a group are optimisations that improve the operation of the protocol. The use of higher layer messages to discover group membership is undesirable but removes the damaging scenario where a multicast sender is retransmitting to a group with no receivers. Extending the protocol for broadcast use is trivial and removes the need for a sender to require group membership information.

It was found that the multicast acknowledgement protocol increased delay and delay variance, compared with DCF in both the unicast and multicast modes, decreased loss and loss variance at low loads but increased these metrics at high loads. The throughput of the TCP streams decreased slightly but not enough to have a strong adverse affect on the performance of these applications. In conclusion, the multicast acknowledgement scheme while reducing loss at low load levels, performs very badly at high loads (worse than the non-retransmitting DCF multicast) and would not, under these circumstances improve the quality of realtime applications.

Chapter 6 - Application layer redundancy

6.1 Introduction

This chapter examines application layer redundancy as a technique to achieve reliability for realtime multimedia applications. An example of a redundancy method is then simulated over the DCF protocol without acknowledgements. DCF (in both unicast and multicast modes) is used as the protocol of comparison and delay, loss and throughput are measured to quantify differences in the protocols. Conclusions regarding the suitability of application level redundancy in relation to DCF are then drawn. A comparison with DCF/LA and multicast acknowledgements will be made in chapter 7.

6.2 Application Layer Redundancy

At the end-to-end level, packet loss signals are often not timely enough to be used by realtime applications for retransmission purposes. In a multi-hop connection the time-out required for a sender to be sure that an acknowledgement is not going to arrive has to be conservative. When the frame is eventually retransmitted and received it may be delayed beyond the usability of the data. Furthermore, since packets are transmitted from a queue, the delayed and retransmitted packet will hold up later packets.

Delay is important to interactive realtime applications. To maintain interactivity on a voice call, for example, the round trip delay should not exceed about 400ms. For non-interactive applications like streamed media, delay is not so important since a few seconds of media can be buffered at the start without reducing quality during playback. For both non-interactive and interactive realtime multimedia applications loss is important, however. Despite advanced receiver repair algorithms quality is almost certainly reduced by frame loss. This shows that the trade-off between delay and loss depends on the type of realtime application, which implies that the decision should be taken by the application, or somehow signalled to the network.

Previous chapters have focussed on lower layer attempts to reduce delay and loss, and seek to reduce loss but also to reduce delay for interactive realtime applications. The link layer has no way of knowing that the application being used is non-interactive and would prefer a reduction in loss while higher delay is tolerable. Attempts to directly signal between layers fall down when the link is not the first in a multi-hop chain, and placing such information inside packet headers has previously failed to be used by most link-layer technologies. Application level redundancy is a chance for the application to dictate the trade-off between delay and loss in accordance with its own requirements.

Redundancy can be used to recover from loss in a more proactive way. In terms of delay, the "retransmission" of a lost data frame will be contained in the next data frame. There is still a delay but, in a multi-hop environment, typically less than that with acknowledgements. In the single-hop wireless environment the difference in delay between the two techniques may not be so great.

The loss distribution can have an effect on the success of redundancy protocols. For example, if packet losses occur in clusters, there is a high probability that two consecutive packets will be lost and so the original data and the redundant data will both be lost. If the loss distribution shows lost packets are distributed evenly over the stream, however, then there is a good chance that redundancy can recover most of these losses.

In a single-hop network the decision as to whether a redundancy or acknowledgement protocol is better for realtime applications is not so clear. Acknowledgements are given priority in DCF and so retransmissions are typically the next transmission on the network. With redundancy the receiver must wait for the next data frame, which for realtime voice is typically 20ms later. The effect of redundancy on throughput, though, can be significant since each frame is effectively retransmitted in advance, proactively anticipating loss. For all the times when the frame arrives successfully (most of the time, hopefully) the redundant information was unnecessary and a waste of bandwidth. Therefore, only applications that use a low proportion of the available bandwidth should use redundancy. Since available bandwidth can vary continuously the decision should be adaptive and constantly revised by the application, adding complexity to the implementation. Using redundancy in a congested network can only increase delay and loss, and decrease performance for a realtime application. The application trade-offs can be more complex. For example, it may be better to reduce the quality of high-quality video streams to enable more bandwidth that can then be used to carry redundant information. The trade-off here is between quality and loss. A slightly lower quality may well be preferable to losing high-quality frames every so often.

Redundancy can be implemented in different specific ways. For example, redundant data need not be transmitted in subsequent frames. If the loss distribution shows subsequent packet losses due to a router queue overflowing, it may be better to transmit the redundant data two, or even more, packets later. Doing so increases delay but may significantly decrease loss.

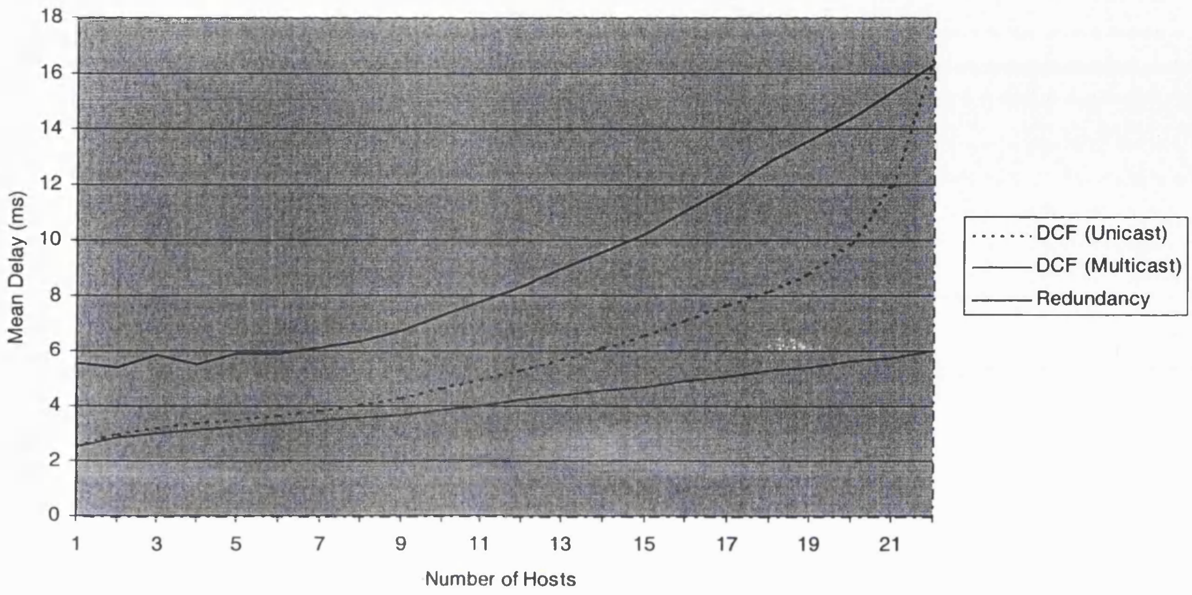
6.3 Simulation Methodology

The simulation environment was the same as in section 2.9.1. DCF was used as the protocol of comparison, in both unicast and multicast modes. DCF unicast TCP traffic continued to use the normal DCF retransmission protocol, while the multicast CBR traffic used the unacknowledged link-layer DCF protocol. CBR packet size was increased in size by an extra third to reflect the redundant payload (compressed video at 128kbps).

When the output of the simulator was analysed, a frame was only considered lost if two consecutive frames were both lost. When a single frame was lost, delay was measured from the initial transmission to the successful reception of the following frame.

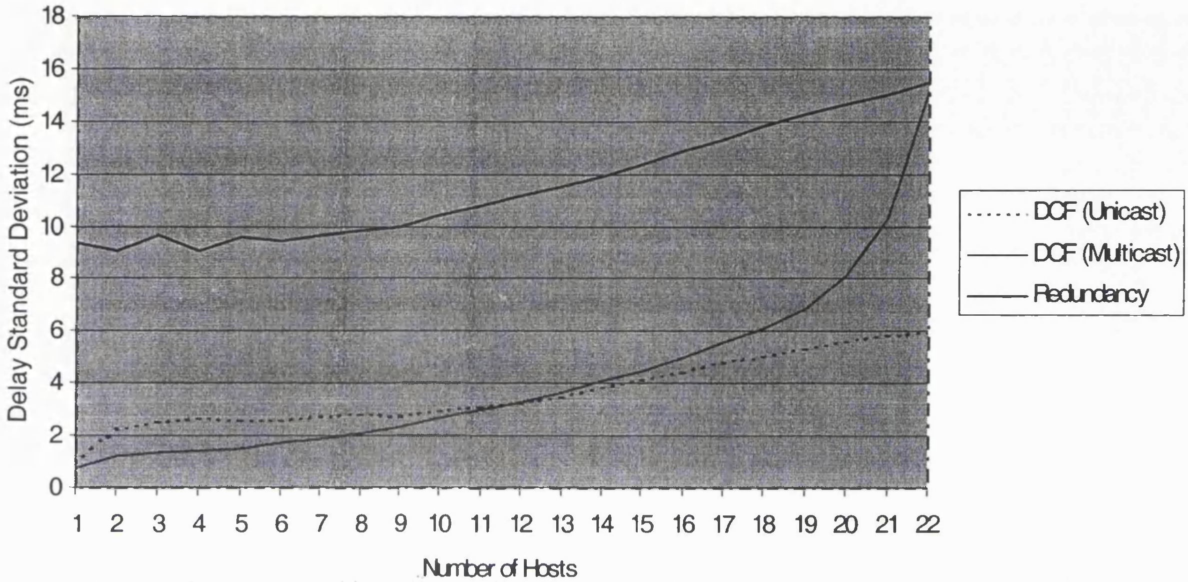
6.4 Simulation results

Graph 6.1: Mean Delay against Offered Load



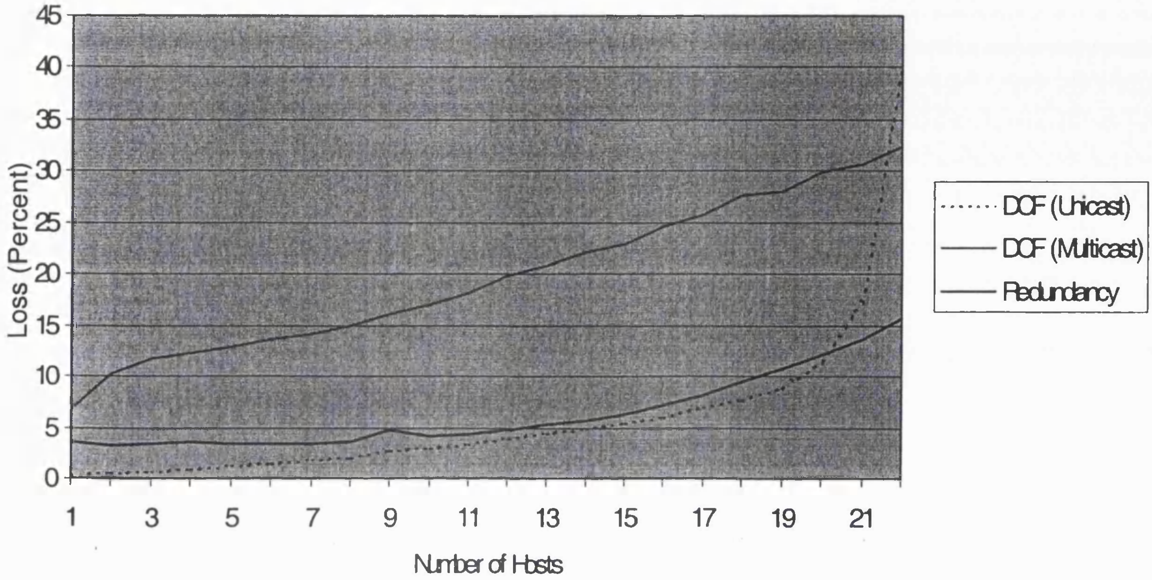
DCF unicast has been seen to increase delay due to the retransmission of colliding frames. This increase is limited, however, by the priority that is given to acknowledgement frames, which enables the sender to time-out quickly and retransmit the data as the next transmission. The redundancy approach to reliability has similar properties: if a data frame is lost the data will be repeated in the next frame, which in this simulation is 33ms later. With DCF unicast, even though the sender has to timeout and retransmit the data this is not expected to take as long as 33ms. Graph 6.1 shows, however, that the average delay for redundancy is only a few milliseconds greater than DCF unicast and the same at very high load levels. This is explained by the fact that the lost and recovered packets are rare and so don't have much influence on the average delay and that DCF unicast performance decreases at high loads.

Graph 6.2: Delay Standard Deviation against Offered Load



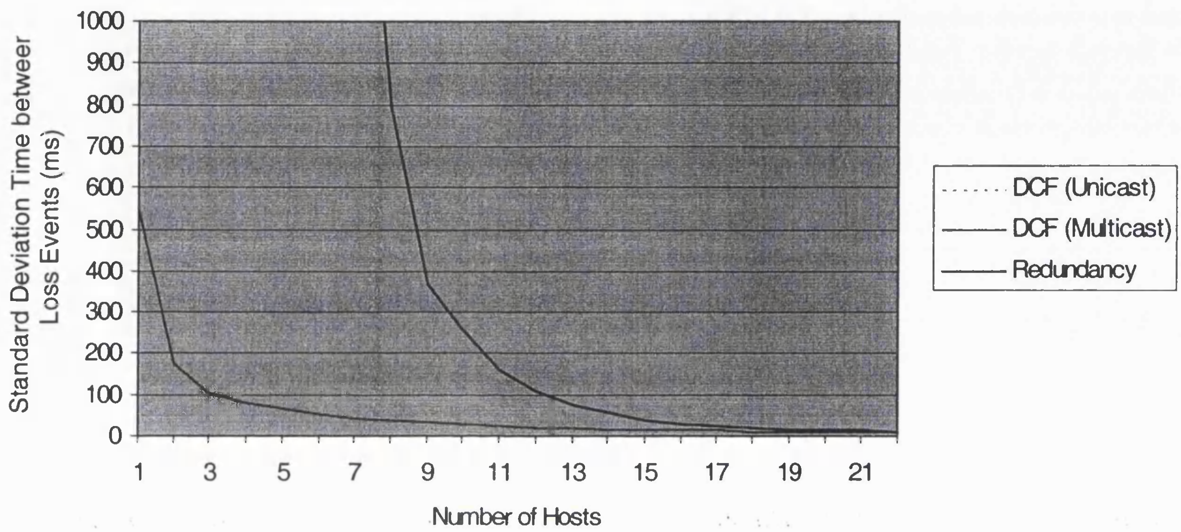
DCF unicast has more delay variation than DCF multicast due to lost frames being retransmitted. The effect should also be present for redundancy, since lost frames are also received in the next data frame. This can be seen in graph 6.2 where redundancy has more delay variance especially at low loads than DCF. At high loads the delay variance becomes similar to DCF multicast but still worse than DCF unicast.

Graph 6.3: Frame Loss against Offered Load



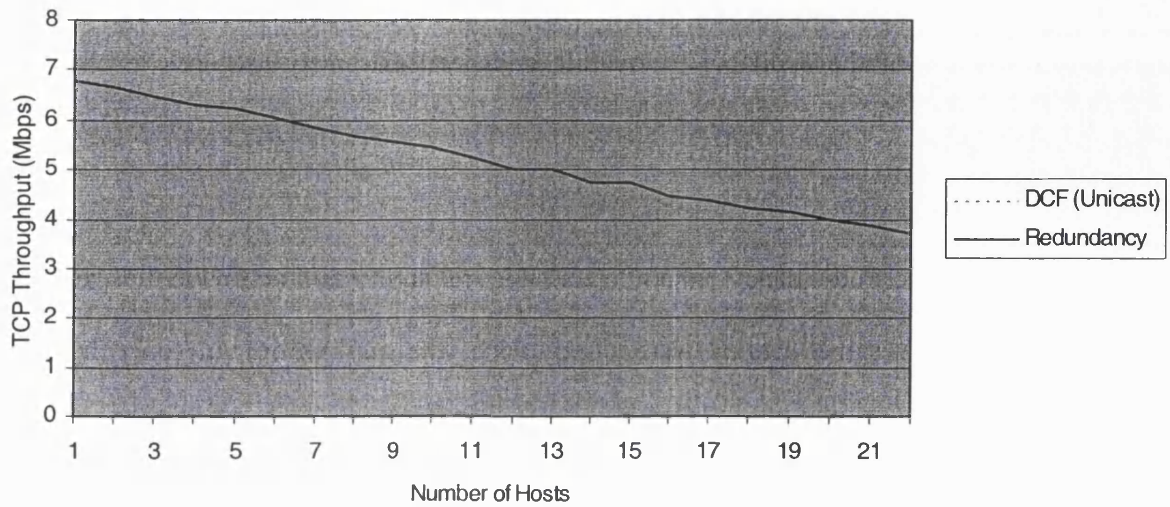
The frame loss graph 6.3 shows redundancy loss levels comparable with DCF unicast except at high loads where redundancy achieves a lower loss rate. Compared to DCF multicast redundancy performs between 5% and 15% better, the largest gains being at higher loads.

Graph 6.4: Variations from the Mean Time between Loss Events against Offered Load



Variations in the loss distribution are shown in graph 6.4. Redundancy displays considerable variation under low load but stabilises to a steady variation as the load increases. Since the loss rates are generally low across the load range, these high variations at low loads should not cause a problem for realtime applications.

Graph 6.5: TCP Throughput against Offered Load



As stated earlier the main trade-off for low loss in redundancy schemes is throughput rather than delay. Increasing the amount of data being transmitted is expected to adversely affect throughput. Many realtime applications however have very low bandwidth requirements and, since the application is in control of the redundancy scheme, the application can decide whether the throughput/loss trade-off is viable. In low bandwidth situations redundancy may not be applicable, but over an 11Mbps wireless LAN, according to graph 6.3, the addition of redundant CBR data does not reduce TCP throughput when compared to the same throughput while the CBR data is carried by DCF unicast. The overhead of the retransmission process of DCF unicast therefore has a similar effect as simply increasing the payload size as redundancy does.

6.5 Conclusions

Application level redundancy offers a significant reduction in frame loss (approximately 15% at high loads), on an IEEE802.11 wireless LAN, at the expense of a small increase in average delay and delay variance. This trade-off would be beneficial for realtime applications that require a small proportion of the available bandwidth, and need extremely low loss levels to improve quality levels.

For TCP streams, redundancy is not a suitable reliability protocol due to the throughput reductions. If the same data is transmitted twice, the application will only have half the bandwidth. For low loss levels this decrease in throughput is not necessary, and low delay is not a requirement for such streams. In this case acknowledgements are preferable and retransmission only done on a reactive basis. For high loss levels the use of redundancy may be worth the trade-off in throughput, however.

It is therefore necessary for the link layer to turn on or off retransmission based on a "delay sensitive" flag in each packet indicating the applications preference. Although a delay flag exists in the IP header, it has been reinterpreted to represent differentiated service information and so could be overwritten or express a different meaning to the network. Application to network signalling has generally failed to be adopted by many link-layer technologies and could also be considered a layer violation. Requiring link-layers to understand higher-layer protocols does not cover all the possible higher-layer protocols that exist and cannot work with future revisions of higher-layer protocols. It is more practical that link-layers will continue to retransmit real-time multimedia data even if it is not necessary as the application is itself using redundancy to reduce loss.

Chapter 7 - Summary and conclusions

7.1 Introduction

This chapter concludes the thesis, compares the sub-hypotheses discussing the relative merits of the three approaches to improving the performance of realtime applications.

7.2 Comparative analysis of proposed techniques

To summarise the conclusions of each sub-hypothesis:

DCF/LA was found, under the tested conditions, to decrease delay and delay variation compared with DCF in both unicast and multicast modes. Loss for DCF/LA was also substantially better than DCF in the multicast mode but did not quite match DCF unicast except at high loads where DCF/LA had lower loss. DCF/LA had comparable TCP throughput as DCF unicast.

The acknowledged multicast protocol simulations showed an increase in delay and delay variation compared with DCF in both unicast and multicast modes. Loss for acknowledged multicast, however, was lower than DCF multicast while higher than DCF unicast, at low load. At high load, loss rates were much higher than DCF in both unicast and multicast modes. TCP throughput decreased slightly compared with DCF unicast especially in the mid-load range. A protocol to allow the link-layer to determine whether groups have members was proposed, but requires higher layer knowledge. The use of such a protocol was considered acceptable based on the consequences of a group with no members on the performance of the network. It was noted that higher layers should perform such filtering themselves.

For the redundancy experiments, delay and delay variation were higher than DCF multicast and unicast. Loss was comparable to DCF unicast (and better at very high loads). TCP throughput remained the same compared with DCF unicast. Since application redundancy removed the need for any link-layer unicast acknowledgement, it seemed favourable that the application be able to signal the link-layer to turn off this reliability mechanism when redundancy is being used. It was noted that this type of signal had been specified and later dropped through lack of use in the past, and so the use of this indication was not pursued.

In comparison, DCF/LA is the preferred protocol for improving realtime performance. DCF/LA produced better loss results than acknowledged multicast over most of the load range, within the tested environment, while providing delay figures lower than DCF. Also, acknowledged multicast showed a reduction in throughput that wasn't present in DCF. The use of DCF/LA removes the need to do link-layer error control. This means that applications don't have to use link-layer error control if it is not a suitable trade-off for them (for example applications that would prefer to use redundancy), and a special technique for multicast acknowledgements is not required (and also layer violations needed to maintain group membership are not needed).

There is still scope for using redundancy for applications which prefer to trade-off some delay budget for reduced loss. This could be combined with DCF/LA with no unnecessary retransmission being carried out by the MAC layer.

7.3 Implications for Typical Scenarios

From the point of view of realtime applications, the network conditions have now changed. By studying particular applications, the applicability of each of the three sub-hypotheses can be examined.

Conference calls fall into the category of an interactive realtime multimedia application. This implies that delay is bounded in order to maintain interactivity. For such an application DCF/LA with its reduction in multicast delay and loss would definitely improve quality. Application level redundancy would be a way to reduce loss further and since the bandwidth used by voice is low compared to general bandwidth availability this may be a way to improve user perception. Redundancy also increases delay, though, which is tightly bounded for interactive applications and may not be a wise trade-off with loss.

In the case of live broadcast video the class of realtime application is non-interactive so delay not so important because a few seconds of media can be stored at the receiver. Loss, however, is important to this particular application since lost frames affect the smoothness of video. For such an application DCF/LA would reduce the multicast loss and improve the video quality. Redundancy may also be beneficial in this case to reduce loss further but the use of redundancy with a high bandwidth application like live video may not be beneficial unless the wireless network has sufficient available capacity.

7.4 Examination of the hypothesis

The hypothesis stated that:

"It is claimed that the performance of realtime multimedia applications on an ad-hoc IEEE802.11b wireless network can be improved by a cross-stack approach to reducing packet loss and delay. The cross-stack approach focuses on improvements at the MAC sub-layer, the data-link and application layers. Delay, delay variance, and loss are used to measure realtime application performance."

In the tested scenarios, the DCF/LA protocol reduces delay whilst maintaining loss levels (reducing loss at high loads) in comparison to the DCF protocol, and is applicable to unicast and multicast traffic. Although not conclusive, this evidence tends to suggest that the DCF/LA protocol could be more generally applicable and is deserving of further study. In the multicast simulations, redundancy also reduces loss but increases delay and so improves performance only for realtime multimedia applications that value loss reduction in preference to an increase in delay. The multicast acknowledgement experiments increased delay and decreased loss at low loads but rapidly increased loss at high loads making its use questionable in a wireless bandwidth limited environment.

7.5 Conclusions

Based on the evidence produced by the simulations, DCF/LA improves realtime performance primarily from a delay perspective, but also from a loss perspective at high load levels. The redundancy simulations also showed reduced loss at high load levels, but increased delay and delay variation. In the multicast simulations, however, DCF/LA and redundancy reduce loss figures across all load levels with DCF/LA also showing decreased delay metrics.

Table of Abbreviations

AP. Access Point

BEB. Binary Exponential Backoff

BLAM. Binary Logarithmic Arbitration Method

CBR. Constant Bit Rate

CSMA/CA. Carrier Sense Multiple Access / Collision Avoidance

CSMA/CD. Carrier Sense Multiple Access / Collision Detection

CTS. Clear To Send

DCF/LA. Distributed Coordination Function / Logarithmic Arbitration

DFWMAC. Distributed Foundation Wireless MAC

FAMA. Floor Acquisition Multiple Access

FEC. Forward Error Correction

FTP. File Transfer Protocol

GSM. Groupe Spéciale Mobile

IEEE. Institute of Electronic and Electrical Engineers

IFS. Inter-Frame Space

IGMP. Internet Group Management Protocol

IP. Internet Protocol

LAN. Local Area Network

MACA. Multiple Access with Collision Avoidance

MACAW. Multiple Access with Collision Avoidance Window

MAC. Medium Access Control

MAP. Medium Access Protocol

Mbps. Mega Bits Per Second

MILD. Multiplicative Increase Linear Decrease

NACK. Negative ACK

NAV. Network Avoidance Vector

PCF. Point Coordination Function

RTP. Realtime Transport Protocol

RTS. Request To Send

SIFS. Short Interframe Space

SFP. SuperFrame Period

STA. Station

TCP. Transmission Control Protocol

UDP. Uniform Datagram Protocol

References

[Halsall 95] Halsall, Fred. "Data Communications, Computer Networks and Open Systems", 4th Edition. Adison Wesley, 1995.

[Tanenbaum 96] Tanenbaum, Andrew. "Computer Networks", 3rd Edition. Prentice Hall 1996.

[Edelson 72] B. Edelson, A. Werth, "SPADE System Progress and Application", COMSAT Technical Review, Vol. 2, No. I, Spring 1972

[Humblet 92] P.A. Humblet, R. Ramaswami, and K.N. Sivarajan. "An Efficient Communication Protocol for High-Speed Packet-Switched Multichannel Networks," Proc. SIGCOMM '92 Conf., ACM, pp 2-13, 1992.

[Palmer 90] L.C. Palmer, L.W. White, "Demand Assignment in the ACTS LBR System," IEEE Trans. on Commun., Vol. 38, pp 684 - 692, May, 1990

[Roberts 72] L. Roberts, "Extensions of Packet Communication Technology to a Hand Held Personal Terminal," Proc. Spring Joint Computer Conference, AFIPS, pp 295-298, 1972

[Crowther 73] W. Crowther, R. Rettberg, D. Walden, S. Ornstein, F. Heart, "A System for Broadcast Communication: Reservation-ALOHA," Proc. Sixth Hawii Int. Conf. System Sci., pp. 371-374, 1973

[Roberts 73] L. Roberts, "Dynamic Allocation of Satellite Capacity through Packet Reservation," Proc. NCC, AFIPS, pp. 711-716, 1973

[Binder 75] R. Binder, "A Dynamic Packet Switching System for Satellite Broadcast Channels," Proc. Int'l. Conf. on Commun., pp.41-1 to 41-5a, 1975

[Capetanakis 79] J.I. Capetanakis, "Tree Algorithms for Packet Broadcast Channels," IEEE Trans. on Information Theory, Vol. IT-25, pp. 505-515, Sept. 1979

[Metcalfe 76] R. Metcalfe, D.R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," Commun. of the ACM, Vol. 19, pp. 395-404, July, 1976

[802.3 85] "Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications" ANSI/IEEE Std 802.3-1985

[Tokoro 77] M. Tokoro, K. Tamaru, "Acknowledging Ethernet", IEEE Compcon 1977, pp. 320-325

[Almes 79] G.T. Almes, E.D. Lazowska, "The Bahaviour of Ethernet-Like Computer Communications Networks," ACM SIGCOMM '79

[802.5 85] "Token-Ring Access Method and Physical Layer Specifications," IEEE Std 802.5-1985

[802.4 85] "Token-Passing Bus Access Method," ANSI/IEEE Std 802.4-1985

[Venkatramani 95] C. Venkatramani, T. Chiueh, "Design, Implementation and Evaluation of a Software-Based Realtime Ethernet Protocol," Proc. ACM SIGCOMM '95, pp. 27-37

[ANSI 87] "Fiber Distributed Data Interface MAC", ANSI Std. X3.139-1987

[Karn 90] P. Karn, "MACA - A new Channel Access Method for Packet Radio", ARRL/CRRL Amateur Radio 9th Computer Networking Conference, Sept. 22, 1990

[Bharghavan 94] V. Bharghavan, A. Demers, S. Shenker, L. Zhang, "MACAW: A media access protocol for wireless LANs", SIGCOMM 94. http://beta.xerox.com/pub/net-research/macaw-cr.ps

[802.11 96] Wireless Medium Access Control and Physical Layer WG, IEEE Draft Standard P802.11, "Wireless LAN," IEEE Stds. Dept. D3, Jan. 1996

[Tobagi 75] F. Tobagi, L. Kleinrock, "Packet Switching in Radio Channels: Part I - CSMA modes and their throughput Delay Characteristics, Part II - The Hidden Terminal Problem in CSMA and Busy-Tone Solution", IEEE Trans. on Comms., Dec. 1975, Vol. 23, No. 12, pp. 1400-1433

[Tuch 93] B. Tuch, "Development of WaveLAN, an ISM Wireless LAN", AT&T Technical

Journal, Vol. 72, No. 4, July/Aug. 1993, pp. 27-37

[Weinmiller 96] J. Weinmiller, H. Woesner, A. Wolisz, "Analyzing and Improving the IEEE 802.11-MAC Protocol for Wireless LANs," Proc. MASCOTS '96, San Jose, CA, Feb. 1996, pp. 200-6

[ETSI 95] ETSI: Radio Equipment and Systems (RES); "High Performance Radio Local Area Network; Functional Specification", draft standard, version 1.1, 1995

[Crow 97] Crow, B. et al. "IEEE 802.11 Wireless Local Area Networks," IEEE Communications Magazine, Sept. 1997

[Crow 97 (2)] Crow, B. et al. "Investigation of the IEEE 802.11 Medium Access Control (MAC) Sublayer Functions." IEEE INFOCOM 1997

[Molle 94] Molle, M. L. "A New Binary Logarithmic Arbitration Method for Ethernet", Computer Systems Research Institute Technical Report, April 1994, University of Toronto, Canada

[Brady 71] Brady, P. T. "Effects of Tranmission Delay on Conversational Behaviour on Echo-Free Telephone Circuits." Bell System Technical Journal, pp115-134, January 1971

[Hardman 95] Hardman, V., Sasse, M.A., Handley, M., Watson, A., "Reliable Audio for Use over the Internet", Proc. INET'95.

[Bolot 95] Bolot J. Crepin H, Vega Garcia A. "Analysis of Audio Packet Loss on the Internet", Proceedings NOSSDAV 95, pp 163- 174, Durham, NH, April 95.

[Schulzrinne 96] H. Schulzrinne, S. Casner, R. Frederick, V. "RTP: A Transport Protocol for Real-Time Applications", www.ietf.org/rfc/rfc1889.txt, 1996

[Deering 89] S. Deering "Host Extensions for IP Multicasting", www.ietf.org/rfc/rfc1112.txt, 1989

[Abramson 70] N. Abramson "The ALOHA System---Another Alternative for Computer Communications", proceedings of the Fall 1970.

[Fullmer 95] C. L. Fullmer, J. J. Garcia-Luna-Aceves "Floor Acquisition Multiple Access (FAMA) for Packet-Radio Networks", proceedings of SIGCOMM '95.

[Garcia-Luna-Aceves 95] C. L. Fullmer, J. J. Garcia-Luna-Aceves "FAMA-PJ: A Channel Access Protocol for Wireless LANs", proceedings of MOBICOM '95.

[Garcés 96] R. Garcés, J. J. Garcia-Luna-Aceves "Floor Acquisition Multiple Access with Collision Resolution", proceedings of MOBICOM '96.

[MANET 02] "Mobile Ad-hoc Network Homepage", http://www.ietf.org/html.charters/manet-charter.html

[Gilbert 60] "Capacity of a burst-noise channel", Bell Systems Technical Journal 39, pp. 1253-1256, 1960.

Bibliography

D. Bantz, F. Bauchot, "Wireless LAN Design Alternatives," IEEE Network, Apr. 1994, Vol. 8, No. 2, pp. 43-53

I. Bar-David, R. Krishnamoorthy, "Barker Code Position Modulation for High-Rate Communication in the ISM Bands," Bell Labs Technical Journal, Autumn 1996, Vol. 1, No. 2, pp. 21-40

F.J. Bauchot, K.S. Natarajan, "Wireless LAN MAC Protocol: 2nd Update", 93/62

V. Bharghavan, A. Demers, S. Shenker, L. Zhang, "MACAW: A media access protocol for wireless LANs", SIGCOMM 94. http://beta.xerox.com/pub/net-research/macaw-cr.ps

P. Brady, "A model for generating on-off speech patterns in two-way conversation," Bell Sys. Tech. J., Sept. 1969, Vol. 48, No. 7, pp. 2445-72

J. Browne, "Top Products of 1996," Microwaves & RF, 1996, Vol. 35, No. 13, p. 189

A. Champness, "Understanding IEEE 802.11", Communications News, 1997, Vol. 34, No. 8, pp.25-26.

M. Chelouche, S. Hethuin, L. Ramel, "Digital Wireless Broadband Corporate and Private Networks: RNET concepts and applications", IEEE Communications Magazine, 1997, Vol. 35, No. 1, p. 42

K.C. Chen, "Medium Access Control of Wireless LANs for Mobile Computing," IEEE Network, Sept. 1994, Vol. 8, No. 5, pp.50-63

K.C. Chen, C.H. Lee, "Group Randomly Addressed Polling for Wireless Data Networks", Proc. IEEE ICC, 1994

K.C. Chen, C.H. Lee, "RAP - A Novel MAC Protocol for Wireless Data Networks", Proc. IEEE GLOBECOM '93

H.S. Chhaya, S. Gupta, "Performance of Asynchronous Data Transfer Methods in the IEEE 802.11 MAC Protocol," IEEE Personal Communications, Oct. 1996, Vol. 3, No. 5, pp.8-15

H.S. Chhaya, S. Gupta, "Throughput and Fairness Properties of Asynchronous Data Transfer Methods in the IEEE 802.11 MAC Protocol," PIMRC '95, 1995, pp. 613-17

B.P. Crow, I. Widjaja, J.G. Kim, P.T. Sakai, "IEEE 802.11 Wireless Local Area Networks", IEEE Communications Magazine, Sept. 1997, Vol. 35, No. 9, pp. 116-126

B. Crow et al., "Investigation of the IEEE 802.11 Medium Access Control (MAC) Sublayer Functions," Proc. INFOCOM 97, Kobe, Japan, Apr. 1997, pp. 126-33

S.G. Glisic, R. Rao, L. Milstein, "The Effect of Imperfect Carrier Sensing on Non-Persistent

CSMA", Proc. ICC '90, paper 332.4

S.G. Glisic, "persistent CSMA in radio channel with imperfect carrier sensing", IEEE Transactions on Communications, March 1991, Vol. COM-39, pp. 458-464

S.G. Glisic et al., "Channel State PDF and Throughput Analysis for Multichannel CSMA Systems with Imperfect Carrier Sensing", ICCS/ISITA '92, Singapore, pp. 681-685

C.C. Huang, R. Khayata, "Delay Spread and Channel Dynamics Measurements at ISM Bands", Proc. IEEE ICC '92

A. Kamerman, "Throughput Density Constraints for Wireless LANs Based on DSSS," Proc. IEEE Fourth Int. Symp. on Spread Spectrum Techniques and Applications (ISSSTA '96), Mainz, Germany, Sept. 1996, pp. 1344-1350

A. Kamerman, "Spread Spectrum Schemes for Microwave-frequency WLANs", Microwave Journal, Feb. 1997, Vol. 40, No. 2, pp. 80-90

A. Kamerman, L. Monteban, "WaveLAN-II: A High-Performance Wireless LAN for the Unlicenced Band", Bell Labs Technical Journal, Summer 1997

A. Kamerman, "Spread-Spectrum techniques drive WLAN performance", Microwaves & RF, 1996, Vol. 35, No. 9, p. 109

C. Kantarjiev, A. Demers, R. Frederick, R. Krivacic, "Experiences with X in a Wireless Environment", Proceedings of the USENIX Mobile & Location-Independent Computing Symposium, 1993

S. Kempainen, "IEEE 802.11 Protocol Firmware and Device Drivers Now Available", EDN, 1998, Vol. 43, No. 1, p. 11

J.H. Kim, J.K. Lee, S.H. Cho, "Performance Analysis of Wireless MAC Protocols with Rayleigh fading, Log-normal Shadowing and Capture Effects,", IEICE Transactions on Communications, 1998, Vol. E81B, No. 1, pp. 73-81

C.Y. Ko, K.C. Chen, C.C. Lu, "Performance of Slotted ALOHA in Multiple Joint Cell of Wireless Networks", Proc. Int'l Symposium on Personal, Indoor and Mobile Radio Communications", Boston, 1992

F. Kuperus, J. Arnbak, "Packet Radio in a Rayleigh Channel", Electronic Letters, 1982, Vol. 18, pp.506-507

J.F. Kurose, M. Schwartz, Y. Yemeni, "Multiple-Access Protocols and Time-Constrained Communications", Computing Surveys, Mar. 1984, Vol. 16, No. 1

S.S. Lam, "Packet Broadcast Networks - A performance Analysis of the R-ALOHA Protocol", IEEE Trans. on Comms, July 1980, Vol. 29, No. 7, pp. 596-603

R.O. Lamaire, A. Krishna, H. Ahmadi, "Analysis of a Wireless MAC Protocol with Client-Server Traffic", Proc INFOCOM '93, Mar./Apr. 1993

R.O. Lamaire, A. Krishna, P. Bhagwat, J. Panian, "Wireless LANs and Mobile Networking: Standards and Future Directions", IEEE Communications Magazine, Aug. 1996, Vol. 34, No. 8, pp. 86-94

C.T. Lau, C. Leung, "Capture Models for Mobile Packet Radio Networks", IEEE Trans. on Comms, May 1992, Vol. COM-40, pp. 917-925

W. Leland, et al., "On the self-similar nature of Ethernet traffic (extended version)", IEEE Transactions on Networking, Feb. 1994, Vol. 2, No. 1

W. Leland, D. Wilson, "High time-resolution measurement and analysis of LAN traffic: implications for LAN interconnections", INFOCOM 91, ftp://ftp.bellcore.com/pub/wel/Leland+Wilson_INFOCOM_91.ps.Z

S. Meditch, C.-T.A. Lea, "Stability and Optimisation of the CSMA and CSMA/CD Channel", IEEE Trans. on Comms, June 1983, Vol. COM-31, No. 6, pp. 763-774

P. Morel, J.D. Decotignie, "Integration of Wireless Mobile Nodes in MAP/MMS", Control

Engineering Practice, 1996, Vol. 4, No. 6, pp. 825-829

K. Pahlavan, T.H. Probert, M.E. Chase, "Trends in Local Wireless Networks," IEEE Communications Magazine, Mar. 1995, Vol. 33, No. 3, pp. 88-95

K. Pahlavan, A. Zahedi, P. Krishnamurthy, "Wideband Local Access: Wireless LAN and Wireless ATM", IEEE Communications Magazine, 1997, Vol. 35, No. 11, pp. 34-40

S. Saunders, "Wireless LANs: Closer to cutting the cord", Data Communications, March 21, 1993

S. Shenker, "Some Conjectures on the Behaviour of Acknowedgment-Based Transmission Control of Random Access Communication Channels", Proceedings of ACM Sigmetrics, 1987

F.A. Tobagi, V.B. Hunt, "Performance Analysis of CSMA/CD", Computer Networks, Oct.-Nov. 1980, Vol. 4, pp. 245-259

B. Tuch, "Development of WaveLAN, an ISM Wireless LAN", AT&T Technical Journal, Vol. 72, No. 4, July/Aug. 1993, pp. 27-37

I. Vukovic, K. Vastola, "Throughput Analysis of Asynchronous CSMA Protocols on Star-Like LAN Topology", Proc. IEEE ICC '92

K. Zhang, K. Pahlavan, "CSMA Local Radio Networks with BPSK Modulation in Rayleigh Fading Channels", Electronic Letters, 1990, Vol. 26, pp. 1655-1656

Z. Zhang, A. Acampora, "Performance of a Modified Polling Strategy for Broadband Wireless LANs in a Harsh Fading Environment", Proc. IEEE GLOBECOM '91, pp. 1141-1146