PACKET VOICE/DATA IN LOCAL BROADCAST NETWORKS

ANDREW DAVID MALYAN

Doctor of Philosophy

ASTON UNIVERSITY

December 1990

PACKET VOICE/DATA IN LOCAL BROADCAST NETWORKS

BY
ANDREW DAVID MALYAN

A thesis submitted for the degree of Doctor of Philosophy

1990

SUMMARY

A local area network that can support both voice and data packets offers economic advantages due to the use of only a single network for both types of traffic, greater flexibility to changing user demands, and it also enables efficient use to be made of the transmission capacity. The latter aspect is very important in local broadcast networks where the capacity is a scarce resource, for example mobile radio. This research has examined two types of local broadcast network, these being the Ethernet-type bus local area network and a mobile radio network with a central base station.

With such contention networks, medium access control (MAC) protocols are required to gain access to the channel. MAC protocols must provide efficient scheduling on the channel between the distributed population of stations who want to transmit. No access scheme can exceed the performance of a single server queue, due to the spatial distribution of the stations. Stations cannot in general form a queue without using part of the channel capacity to exchange protocol information.

In this research, several medium access protocols have been examined and developed in order to increase the channel throughput compared to existing protocols. However, the established performance measures of average packet time delay and throughput cannot adequately characterise protocol performance for packet voice. Rather, the percentage of bits delivered within a given time bound becomes the relevant performance measure. Performance evaluation of the protocols has been examined using discrete event simulation and in some cases also by mathematical modelling.

All the protocols use either implicit or explicit reservation schemes, with their efficiency dependent on the fact that many voice packets are generated periodically within a talkspurt. Two of the protocols are based on the existing 'Reservation Virtual Time CSMA/CD' protocol, which forms a distributed queue through implicit reservations. This protocol has been improved firstly by utilising two channels, a packet transmission channel and a packet contention channel. Packet contention is then performed in parallel with a packet transmission to increase throughput. The second protocol uses variable length packets to reduce the contention time between transmissions on a single channel. A third protocol developed, is based on contention for explicit reservations. Once a station has achieved a reservation, it maintains this effective queue position for the remainder of the talkspurt and transmits after it has sensed the transmission from the preceeding station within the queue.

In the mobile radio environment, adaptions to the protocols were necessary in order that their operation was robust to signal fading. This was achieved through centralised control at a base station, unlike the local area network versions where the control was distributed at the stations. The results show an improvement in throughput compared to some previous protocols. Further work includes subjective testing to validate the protocols` effectiveness.

Key words:
Packet voice, Local area networks, Mobile radio networks, Random multiple-access protocols.

Results? Why man, I have gotton a lot of results. I know 50,000 things that won't work.

Thomas A. Edison

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank my supervisor Dr. R. L. Brewster for all the encouragement and constructive criticism he has given throughout the research. I am also grateful to all my friends in the Department of Electrical Engineering and Applied Physics for their helpful discussions.

I wish also to thank the U.K. Science and Engineering Research Council for their financial support.

Finally I would like to thank Frances Bishop, my parents and God for all their support and encouragement.

CONTENTS

ABBREVIATIONS

CSMA/CD	: Carrier Sense Multiple Access with Collision Detection
R-VT-CSMA/CD	: Reservation Virtual Time CSMA/CD
AR-VT-CSMA/CD	: Adaptive R-VT-CSMA/CD
DC-R-VT-CSMA/CD	: Dual Channel R-VT-CSMA/CD
Logical TDMA	: Logical Time Division Multiple Access
ICMA/CD	: Idle-signal Casting Multiple Access with Collision Detection
ATM	: Asynchronous Transfer Mode
ISDN	: Integrated Services Digital Network
LAN	: Local Area Network
MAC	: Medium Access Control
PABX	: Private Automatic Branch Exchange
PSDN	: Packet Switched Data Network
PSTN	: Public Switched Telephone Network
WAN	: Wide Area Network

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

a : Network propagation delay/packet transmission time ratio.
b : Length of packet (general) in bits.
b_d : Length of data packet in bits.
b_r : Length of reservation mini-packet packet in bits.
$E[B]$: Mean duration of a busy period in a cycle.
C : Channel capacity (bps).
C_r : Capacity of reservation channel in Logical TDMA.
Δ : Total end-to-end delay a voice packet experiences.
Δ_1 : Reservation delay for data packets.
Δ_2 : Queueing delay for data packets.
δ : Carrier detection time on signalling channel in DC-R-VT-CSMA/CD.
F : Frame length.
G : Packet arrival rate per T seconds.
γ : Talkspurt to total time ratio.
H : Packet header transmission time.
h : Packet header transmission time to packet transmission time ratio.
$E[I]$: Mean duration of an idle period in a cycle.
L : Limit on the number of voice packets that can be transmitted in Logical TDMA.
λ^{-1} : Mean talkspurt length.
λ_d : Mean data packet arrival rate.
λ_t : Mean arrival rate of new talkspurts per reservation slot in a frame.
μ^{-1} : Mean length of silence period.
N : Total number of voice stations.
N_f : Mean number of fades per second.
η : Virtual time clock update factor.
p : Probability of one transmission in a slot.
P_s : Probability of a successful packet transmission.
P_b : Equilibrium probability that the virtual time clock is backlogged.
P_c : Equilibrium probability that the virtual time clock is up to date.
P_e : Probability of a fade occuring during a packet header transmission.
P_{Loss} : Fraction of packets lost.
P_{max} : Maximum length voice packet.
P_{min} : Minimum length voice packet.
P_s : Shortest transmitted packet.
P_{01} : Probability that a station silent in the previous frame is in talkspurt this frame.
ρ : Threshold power level/average signal-to-noise power level.

ρ_d : Utilisation of data region within a frame.

ρ_v : Utilisation ofvoice region within a frame.

S : Channel throughput (general).

S_D : Data packet throughput in Logical TDMA.

S_d : Data reservation mini-packet throughput in Logical TDMA.

T : Packet transmission time (general).

T_d : Data packet transmission time.

T_{dr} : Data reservation mini-packet reservation time.

T_{max} : Maximum length time constraint.

T_{min} : Minimum length frame time.

T_{pre} : Preamble transmission time.

T_s : Transmission time of shortest packet.

T_v : Length of voice region in Logical TDMA.

T' :Transmission time of voice packet using DC-R-VT-CSMA/CD.

t_f : Fade interval (mean T_f).

t_i : Inter-fade interval (mean T_i).

t_n : Non-fade interval (mean T_n).

τ : Maximum end-to-end network propagation delay.

ϑ : Total transmission overhead.

φ : Fraction of total bandwidth assigned to message channel.

E[U] : Mean time during a cycle that a channel transmits packets without collisions.

V : Voice encoding rate (bps).

W : Total available bandwidth.

W_A : Acknowlegement channel bandwidth.

W_R : Reservation channel bandwidth.

υ : Number of reservation slots in a frame.

ζ : Wasted channel time due to collisions.

σ : Number of queued voice stations in Logical TDMA.

ς : Length of reservation mini-packet to length of data packet ratio.

CHAPTER 1

INTRODUCTION

1.1 Integrated Packet Networks

Current communication networks have been designed to support specific services. For example, the PSTN (Public Switched Telephone Network) for voice, and the X.25 PSDN (Packet Switched Data Network) and X.21 CSDN (Circuit Switched Data Network) for data. The parallel existence of these independent application oriented networks is inefficient, expensive, and inflexible for future communication requirements. What is needed are networks that provide basic communications capabilities in an application-independent fashion.

ISDN (Integrated Services Digital Network) has evolved to provide an integrated digital customer access between the customer premises and the ISDN, in order to support a multiplicity of services [20]. ISDN, however, is not really integrated as there are two different bearer services, circuit switching and packet switching, resulting in two overlay services. The problem with ISDN is the reliance on circuit switching, which requires that the available bandwidth be divided up into fixed size channels. The channel size is a permanent feature which does not provide the desired flexibility due to resource dedication. A fully integrated network requires the same switching fabric be used for handling all traffic types.

Future communications like Broadband ISDN will combine various types of traffic such as data, voice and video. This will be achieved by statistically multiplexing ATM (Asynchronous Transfer Mode) cells onto a single network in an application-independent fashion [19,97]. With a capacity range from a few bits per second (bps) up to 150Mbps, integration of this kind in wide area networks (WANs) will support a wide variety of services, whilst providing flexibility for future requirements.

Full integration requires that local networks must be capable of inter-netting with such WANs. In the office environment, voice and data integration can be realised through a PABX (private automatic branch exchange). However, the circuit switched technology employed may impose fundamental limitations for certain types of applications including [95]:

- Gateway interconnection with B-ISDN will be complex, due to the requirement of cell conversion.

- Instantaneous transmission bandwidth available is constrained to at most 64Kbps. This could be a limitation for many kinds of traffic such as high-resolution graphics and computer-to-computer communications.

- Protocol conversion support is generally minimal.

- Bursty traffic which does not possess long holding times is inefficient with capacity.

- Broadcast distribution and resource sharing are ill-suited for point-to-point circuit switched connections.

Alternatively a LAN (Local Area Network) could be used which is based upon packet switching. Throughout this thesis, the integration of voice and data packets has been considered on local broadcast networks. The advantages of using packet switching for an integrated services network is that it is adaptable to changing traffic. It provides the user with exactly the capacity required, unlike circuit switching. Thus, call blocking can be a function of the required average capacity rather than the peak capacity. An integrated voice and data packet communication system has several advantages over the use of separate circuit and packet switched networks. These include [11,45,115]:

- Potential cost savings through sharing of switching and transmission resources.

- If silence suppression is used, voice packets are generated only when the user is in a talkspurt, so there is no resource dedication to any speaker, unlike circuit switching. As a result, significant savings in channel capacity can be achieved because periods of silence are used to transmit packets from other sources.

- Only one of the voice conferees needs to use the channel capacity at any given time for voice conferencing.

- Packet internetworking techniques can be applied to provide intercommunication among voice users on different types of networks.

- Speech is a compressible source [31] that can be coded at rates ranging from 64Kbps down to 2.4Kbps. Packet switching can be used for variable-rate voice coders in order to adapt to network conditions.

The problem of realising this integration, however, is that voice and data have different traffic characteristics and require different types of service from a network. The above advantages are achieved in packet switching by buffering packets. Such buffering delays packets and hence introduces delay into the conversation. Thus, to realise an integrated packet network we must formulate an appropriate transmission protocol for handling both voice and data, bearing in mind the end-to-end user oriented performance requirements of both. These requirements for packet voice and data are now examined.

1.2 User Requirements For Packet Voice And Data

Depending upon the application, data packet delays of up to several seconds may be acceptable. However, the integrity of data is very important and thus any packet loss is usually unacceptable. Voice packets have quite different network requirements.

The main problem identified with packet voice is large end-to-end variable delays [15,38,81]. Such delays are a potential problem in any shared resource network that uses buffering to resolve contention. Thus, in order to have the advantages provided by an Integrated Packet Switched Network (IPSN), it is necessary to design voice protocols that limit the end-to-end and variable delay of voice packets to such an extent that it is not objectionable to listeners.

Thus the principle requirement for voice traffic is bounded delay. Voice packets that are not delivered to the receiver within some period, D_{max}, must be discarded, resulting in a percentage loss, P_{loss}, of voice packets.

1.2.1 Voice Packet Delay

The end-to-end delay a voice packet experiences in a broadcast network is composed of several parts.

- ***Packetisation delay***, F, is the time it takes to form a complete packet at the originating station from the serial output stream of a voice A/D converter. This delay is given by $F = b_v/R$, where b_v is the number of information bits/packet and R is the digitised voice encoding rate in bit/s. Since voice is only encoded in talkspurts, the packet arrival rate from a station is periodic, with inter-arrival time F.

In order to minimise both the packetisation delay at the transmitter and the perceptual effect of lost packets at the receiver, packets should be as short as possible. If the discarded packets are shorter than about 50ms, the loss can appear to the listener as background noise. Ideally, to reduce the packetisation delay, shorter packets would be used. However, in order to maintain high channel utilisation, the number of bits per packet should be as high as possible to reduce the effects of overhead in the packet header.

- ***Access delay***, t_a, is a variable delay experienced by packets whilst waiting to be transmitted on the broadcast network.

- ***Transmission delay***, T, is a function of the packet length and transmission capacity of the network. It is given by $T = (b_v+\vartheta)/C$, where ϑ is the number of overhead bits per frame and C is the network transmission rate.

- ***Propagation delay***, τ, the time it takes for the signals to propagate from the receiver to the destination.

Through experimentation, it has been shown that conversations are adversely affected by roundtrip delays of 600ms, though users are unaware of its effects until it is 1200ms [58]. These results however apply only for fixed end-to-end delay. It has been shown in [24] that packets which are delivered to the receiver in excess of 200ms on a terrestrial network are commercially acceptable. The CCITT [10] recommends that for a call to be acceptable, the mean end-to-end delay between two subscribers in an international call should not exceed 400ms, providing echo control devices such as echo suppressors and echo cancellers are used. Throughout this thesis, a limit of 300ms has been assumed.

Subject to this time constraint D_{max}, variability of delay is compensated for by adding a controlled variable delay to voice packets, depending on the network delay they have already experienced when they arrived at the receiver [3,85,104]. This is necessary to reach a compromise between delay and gaps in the output speech. A fixed buffering delay would suffice if network delays and delay deviation are small. If delays are expected to be large, or dispersions vary greatly with network load, it is more desirable to use an adaptive algorithm [25].

For example, packet voice communication within a LAN, for simplicity may discard packets if they have not been transmitted by one packetisation time since they were formed. Thus a delay buffering time of one packetisation interval at the destination would suffice to produce a serial output stream. For inter-LAN communication, the delay from the transmitter to the receiver may have a greater time constraint than one packetisation interval and so this method may be unacceptable due to the total end-to-end delay produced. The approach in [3] is to include a few bits of information in the packet header indicating at what time the packet was sent, called a *timestamp*. Each packet can then be buffered until its total delay is equal to a control time and then played out.

1.2.2 Voice Packet Loss

Due to the redundancy of speech, low values of packet loss (P_{loss}) are acceptable to listeners. Informal listening tests showed that levels of 5 to 10 percent packet loss is not objectionable, if the gaps are filled with repeated packets [49]. Formal listening tests in [39] have shown that packet loss of up to 1-2 percent is acceptable if the discarded packets are below about 50ms.

Various techniques exist for minimising the annoyance to the listener such as replaying the previous packet. If the discarded packets are larger than about 50ms, syllables may be lost, resulting in greater annoyance to the listener [14]. For a given loss level, the acceptability of the speech is inversely proportional to the mean packet length [40,49].

A number of strategies exist for the reconstruction of lost packets at the receiver, the simplest being filling the missing packets with silence or substitute the last correctly received packet. In [35], a substitute waveform based upon pitch detection was derived from the last correctly received packet so as to avoid phasing discontinuities. For low bit rate speech encoding, where prediction is already part of the decoding process, reconstruction of lost packets may be predicted on the basis of the last correctly received set of speech parameters. The problem of missing packets does not necessarily have to be dealt with at the receiver. The speech encoding mechanism could be used to reduce this annoyance to the listener.

Variable-rate and embedded-coding attempt to handle overload by reducing the coder bit rate rather than by discarding whole packets. One such method is if odd and even PCM samples are placed into different but consecutive packets at the transmitter [49]. If a packet then does not arrive at the receiver within the time-bound, a complete speech segment still exists, albeit with lower fidelity. Although, with this method, the packetisation delay is now doubled. However, with more advanced signal processing, larger percentages of speech packets can be lost and regenerated with no distinguishable effect on perceived quality [94].

1.3 Broadcast Networks

Broadcast networks allow stations to exchange messages over a shared broadcast channel. There are no intermediate switching nodes between stations, each station being attached to a transmitter/receiver that can communicate over this shared medium. The need for such networks arises whenever there is the requirement to share expensive resources and for communication among stations.

The shared channel cannot generally support several transmissions simultaneously. Thus broadcast networks require some form of cooperation between stations. *Medium Access Control* (MAC) protocols provide such a mechanism for sharing the channel under distributed control.

1.3.1 Medium Access Control Protocols

Communication within a broadcast network uses the physical and data link layers in the OSI reference model [70]. The physical layer performs the encoding/decoding of the signal, preamble generation/removal and bit transmission/reception. The data link layer controls the transmission of information over a link. This includes functions such as connection, initialisation, data formatting, address recognition, error control, flow control and connection termination. Within the data link layer, the IEEE 802 family [2,9,120] groups these functions into a sublayer called the logical link control layer. The data link layer must also manage access to the network. This is achieved by it having another sublayer called the MAC layer, as illustrated in figure 1.1.

In a broadcast network, only one station can successfully transmit on the shared medium at one time. A MAC protocol is an algorithm which separates ready packets of different stations until they are isolated enough to be transmitted without interference.

Access control techniques can be either synchronous or asynchronous. The two synchronous techniques are Time-Division Multiplexing (TDM) and Frequency-Division Multiplexing (FDM). These methods assign a fixed subset of the time-bandwidth space to each of several users. This is not optimal for broadcast networks because the needs of stations are generally unpredictable. For example, if one station has no data to transmit its time-bandwidth space is left idle, whereas a station with much data can only transmit in its own capacity allocation, and is unable to use any idle capacity. The capacity should be allocated to users in an asynchronous (dynamic) manner, based upon their requirements. To achieve this, packet communication is used, whereby part or all of the available resources are allocated to one user at a time for a short period of time. The asynchronous TDM techniques can be divided into contention and deterministic methods [103].

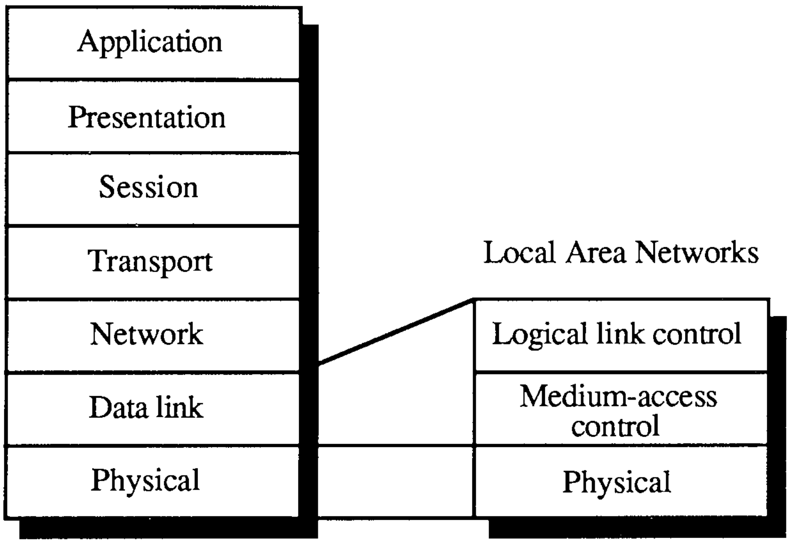


Figure 1.1 LAN standards apply to the data link and physical layers of the OSI reference model.

Stations using contention protocols have no definite or scheduled time when they should transmit. All stations contend for time on the network since there is no control mechanism for determining whose turn it is to transmit. This class of protocols include *Aloha* [109] and *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD) [46,111,122].

Deterministic access techniques can be either centralised such as *polling* [5,103], or distributed. Two commonly used distributed deterministic access schemes are *Token Passing* [5,47] and the *Slotted-Ring* [5,103] methods.

MAC protocols and their performance differ according to the environment and the system requirements to be satisfied. Broadcast networks come in many forms, each with differing performance characteristics because of the particular type of multiple access channel that is utilised. This research has examined the transmission of voice and data packets in two types of broadcast network; (i) contention type local area networks (LANs), and (ii) mobile radio networks with a central base station.

1.3.2 Local Area Networks

Local Area Networks (LANs) provide communication between stations in a small area, generally less than 10Km, such as a building. Depending upon its transmission medium, a LAN can achieve data rates between 100Kbps and 100Mbps whilst providing reliable data transmission. A LAN can be configured in three basic topologies, star, ring or bus/tree [51,103].

The bus topology is one of the most common topologies used with LANs. When a coaxial cable is used, the end-to-end propagation delay over the network is generally much smaller than a transmission time, thus providing useful feedback about the state of the channel to all stations in the network. The MAC protocol in the Ethernet [78,122] contention type LAN utilises such information in its CSMA/CD MAC protocol. CSMA/CD was developed for stations with bursty data and operates well, except at high load where instability occurs [111]. At heavy load, packet delay increases and throughput decreases due to the increased packet collision rate. A larger capacity on the bus can be achieved by using the Token Passing principle [47,51,103]. Here a station must wait to receive a token before transmitting. Once it has received the token and transmitted its packet, the station retransmits the token with the address of the next station within the network. This round-robin approach is capable of supporting a higher throughput than CSMA/CD due to the absence of collisions at high load, although it is a more complex algorithm.

1.3.3 Mobile Packet Radio Networks

In a mobile radio network, stations may be distributed over a geographic area of tens of kilometres. Propagation between the transmitting antenna and a mobile antenna occur over several paths such as the line-of-sight path and the path due to scattering caused by reflections from different structures. The received signal undergoes random variations in both amplitude and phase as a result of vehicle movement.

The topology in the network can change rapidly with mobile stations, and so the system connectivity is difficult to predict with the random variations in phase and amplitude. Due to the *hidden terminal* problem [107], a transmitting station may be unable to determine whether its packet was received correctly at the destination simply by monitoring channel activity.

The radio network has a similar topology to that of the bus/tree LAN, but has a lower data rate with highly variable error rates. Consequently, techniques such as token passing are not feasible. Contention protocols are used in this environment since stations can be added or deleted easily and most malfunctions due to fading etc. are localised to single stations and do not affect the whole network.

1.4 Problem Statement

Due to various functional and economic factors, it is advantageous to integrate packet voice and data on a single network. This is particularly true in mobile radio, where the growth in personal communications will require efficient transmission schemes in a limited bandwidth. Contention type (802.3) LANs such as Ethernet are relatively common and share a similar topology with a packet radio network. The objective of the research was to examine and design various MAC protocols suitable for integrated voice/data packet operation on:

i) IEEE 802.3 type LAN

Traditionally LANs have been used for data transfer. Consequently MAC protocols for such networks have been developed towards fulfilling data transmission requirements, and not those of time critical voice.

ii) Mobile radio with a central base station.

In this environment the topology of the network can change rapidly, resulting in random variations in both amplitude and phase. In such a situation, stations that are within range of their intended destinations may only hear a subset of the transmissions on the channel. Access schemes in this environment must allow the system to be robust and adapt itself to these changes.

The protocol operations should ideally be independent of the number of voice stations (unlike polling etc.) to provide greater flexibility in LANs and for the basic operations to be applicable for mobile radio networks, where the number of stations may vary dramatically. When designing a channel access protocol, it is necessary to define a set of criteria against which different protocols may be compared [105]:

- ***Throughput***, defined as the fraction of time useful traffic is transmitted on the channel.
- ***Delay characteristics*** in terms of mean and peak, and variability for voice packets.

- ***Flexibility*** to changing voice/data user requirements.
- ***Stability*** of throughput in the presence of channel congestion, at high input loads.
- ***Robustness*** of protocol operation in the event of station failures and channel errors.
- ***Implementation*** in terms of cost/complexity at terminals/base station.

MAC protocols must provide efficient scheduling of a broadcast communications channel that is shared by a distributed population of stations who want to transmit messages. No access scheme can exceed the performance of a single server queue, due to the spatial distribution of the stations. Stations cannot in general form a queue without using part of the channel to exchange protocol information.

The established performance measures of average packet time delay and throughput cannot adequately characterise protocol performance of packet voice. Rather, the percentage of bits delivered within a given time bound becomes the relevant performance measure. Subject to the time constraint on voice packets, data packets must also be allocated an acceptable user throughput.

1.5 Thesis Outline

The thesis is organised as follows. Chapter 2 reviews previous work on contention based MAC protocols for packet voice in LANs. Various techniques are examined for separating ready packets from different stations, until they are isolated enough to be transmitted without interference. This separation is generally achieved through either probability or through time, based upon packet arrivals.

Chapter 3 then examines in detail a fixed length voice packet LAN protocol that assigns implicit reservations to stations dependent upon their packet arrival times. The throughput of this protocol has been improved by adapting the collision backoff procedure. The time-bandwidth efficiency of the protocol is then derived through analytical and simulation modelling. The results of this analysis was the basis of a new protocol with a dual channel architecture. The data throughput of this protocol is then calculated analytically and simulated for various voice loads.

In chapter 4 a variable length packet LAN protocol has been designed based on the original protocol in chapter 3. Again it achieves a high throughput through implicit reservations and also by using longer packetisation intervals as the channel load increases in order to reduce the per-packet overhead of transmissions. The performance of this protocol is derived through simulation. A second protocol designed in chapter 4 also uses variable length packets. High channel efficiency is attained through explicit reservations from stations, which form a distributed global queue. Voice loss is examined both analytically and through simulation. A summary then discusses the relative advantages/disadvantages of the fixed and variable length voice packet protocols designed in chapters 3 and 4 for use in LANs.

Chapter 5 develops further the protocols designed in chapter 4, but for use in the mobile radio environment. The various characteristics of a radio channel are discussed, which affect the protocol designs. They are then compared both analytically and through simulation for various channel conditions.

Chapter 6 reviews the modelling process of the network protocols simulated in the research , along with the voice source conversation and mobile channel models.

Finally in chapter 7, the research is concluded by briefly comparing the various attributes of the developed protocols. Details of further recommended work are also discussed.

CHAPTER 2

REVIEW OF PREVIOUS WORK IN CONTENTION TYPE LANS

2.1 Characteristics/Requirements Of Packet Voice And Data

Contention type networks are not normally used in real-time environments due to their non-deterministic nature, there being no upper bound on the access time for a given node. Since they have traditionally been used for data transmission, the MAC protocols adopted have been tailored towards data characteristics. However, the traffic characteristics and transmission requirements for voice packets are fundamentally different from those for data, and thus these protocols may no longer be very efficient in an integrated voice/data environment. Table 2.1 summarises these differences, which were discussed in chapter 1.

	Data	Voice
Arrival pattern	Random.	Periodic during talkspurts.
Delay	Depending upon application, can be up to several seconds.	End-to-end delay should not exceed 300ms.
Integrity	No packet loss is acceptable.	Due to redundancy in speech, up to 2% packet loss can be acceptable.

Table 2.1 Comparison of packet voice and data.

Although the end-to-end delay has been rather arbitrarily stated as 300ms, a considerably smaller delay is necessary for LANs to allow them to operate as gateways between public and private networks. For example, if a person in one LAN was to communicate to another person in a LAN over a 270ms satellite link, the delay for each LAN should be kept below 15ms.

Transmitting digital speech/data over an integrated services packet network entails packetising the digital signal at the transmitter. For example, PCM speech is usually encoded into a 64Kbps bit stream consisting of 8000x8-bit speech samples per second, although more efficient encoding schemes are possible. Packets are then transported over the network and depacketised at the receiver. The end-to-end delay, Δ, that a packet experiences on a LAN was discussed in chapter 1 and is made up of four components given by:

$$\Delta = F + t_a + T + \tau \qquad ...(2.1)$$

where F is the packetisation delay commonly called the voice frame time, t_a is the channel access time, T is the packet transmission time and τ is the network propagation delay. The end-to-end delay, Δ, is variable due to the variability of the channel access time t_a. The speech packet process is illustrated in figure 2.1, where the receiver buffer attempts to reduce the subjective effects of this variability.

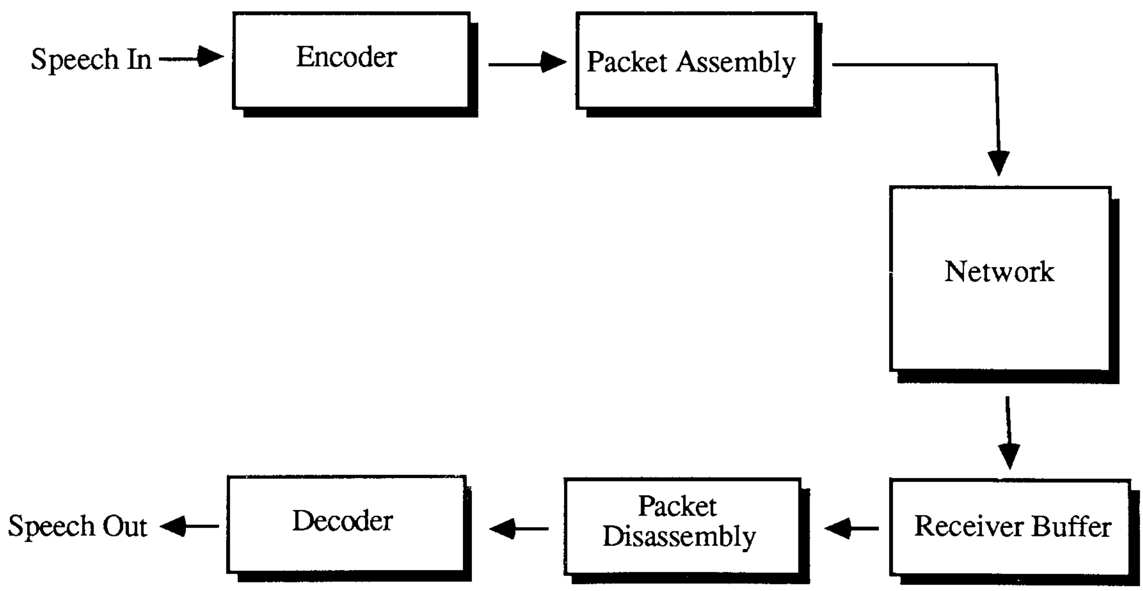


Figure 2.1 The voice packet process.

2.2 Boundary Techniques For Speech/Data Protocols

The two general approaches to designing an integrated voice/data network are to use fixed or movable boundaries. In these schemes the channel time is divided into frames of length equal to a voice packetisation interval. A protocol is classified as fixed boundary if the channel capacity is partitioned into two distinct regions, one for voice and the other for data. This fixed boundary approach becomes inefficient under unbalanced load conditions. In order to permit one traffic class to use the other region, a movable boundary [59,100] can be used.

Although the movable boundary technique is more flexible, the complications of this approach may be much higher. A simulation study in [54] examined the fixed and movable boundary techniques and recommends a hybrid technique depending on the mix of traffic loads. When there are enough voice and data channels, a fixed boundary technique should be used because of its lower operating costs. A protocol described in [124] increases the voice subframe whenever new voice stations collide. To guarantee a certain level of service to data stations, an upper limit is placed on the voice subframe. This attempt at balancing the loads and resources is achieved by pilot tones.

Some previous work is now reviewed which adopts contention based MAC protocols for data in LANs carrying an integrated voice/data load. The protocols are qualitatively examined, since it is difficult to compare many of the protocols numerically due to differently used simulation parameters. The objective is to minimise the contention time between voice stations in order that the voice packet requirements can be met and that more capacity may then be available for data stations. The protocol operations should ideally be independent of the number of voice stations (unlike polling etc.) to provide greater flexibility in LANs and for the basic operation to be applicable for local radio networks, where the number of stations may vary dramatically.

2.3 Probabilistic Partitioned Protocols

With any contention MAC protocol, channel time can be wasted due to collisions, this being a period of time during which two or more stations are transmitting simultaneously. To reduce the collision time, most contention type LANs use a form of Carrier Sense Multiple Access with Collision Detection (CSMA/CD) to access the medium. This MAC protocol is particularly suited to data, which has a bursty traffic pattern. It partitions stations with packets ready for transmission in a probabilistic manner. If the propagation delay between a sending and a receiving node is small compared to the packet transmission time, then when a station sends a packet all the other stations in the network are aware of it within a fraction of the packet transmission time. Observing the channel state thus provides valuable feedback as to whether another station is currently transmitting.

With CSMA/CD, when a station has a packet ready for transmission it monitors the channel to determine whether any transmission currently exists on the medium. If it is sensed busy, transmission is deferred according to a *persistence algorithm* [55]. Three types of persistence algorithm are possible:

(i) Nonpersistent. The station backs off a random amount of time and then senses the medium again.

(ii) 1-persistent. The station continues to sense the medium until it is idle and then transmits.

(iii) P-persistent. The station continues to sense the medium until it is idle and then transmits with probability p. Otherwise it backs off a fixed amount of time with probability (1-p) and repeats the procedure.

If the medium is sensed idle, the packet is transmitted immediately with nonpersistent and 1-persistent and with probability p for p-persistent CSMA/CD. After transmitting a packet, a station continues to monitor the medium to detect if a collision arises. If a collision occurs, all transmitting stations cease their transmissions and send a brief jamming signal to notify all stations of this collision. Stations then backoff a random amount of time before they attempt to transmit again. Collisions can only occur when two or more stations begin transmitting within a period of time not greater than the network propagation delay. Thus the CSMA/CD protocol is more effective for networks in which the packet transmission time is much longer than the propagation delay.

2.3.1 CSMA/CD And Packet Voice

An Ethernet CSMA/CD network is bit serial and has a 10Mbps capacity. If 64Kbps speech were used, complete utilisation of the capacity would result in the network supporting 195 two-person conversations, if each person spoke for an average of 40% of the time. Such efficiency, however, can never be achieved in practice. This is due to various reasons such as per-packet overhead in the packet header and trailer, network propagation time between stations, interframe spacing between transmissions and, when the network propagation delay is large, it is then predominantly due to collision jamming periods.

The simplest algorithm for transmitting voice would be to transmit fixed length packets asynchronously on the network. In this algorithm, each station will begin to transmit a new packet one frame time after beginning to attempt to transmit its previous packet. Thus, if two stations happen to collide with each other once, they will then continue to collide with each other every packet time there afterwards until one of them begins a silence, assuming silence suppression is being used.

An improvement to this algorithm deals with the problem of packet transmissions being delayed at the source until the medium has been successfully acquired. If, while the packet is waiting to be transmitted more speech samples are buffered, these can then be appended to the old packet before it is transmitted. This has the advantage that fewer packets are transmitted under heavy load. At light load this also produces a synchronisation effect due to the periodicity of voice packets in a talkspurt. If a station finishes transmitting its packet at time t_0, its next packet will be ready for transmission one frame time later at time t_0+F. Thus at light load, once a station has succeeded in transmitting a packet, it is unlikely to collide with another successful station who transmitted at time $t_{0'}$, since this station`s next packet will be ready at time $t_{0'}$+F. As the load increases, however, this synchronisation fails due to unsynchronised stations` colliding with synchronised stations and these stations retransmission attempts further colliding with other synchronised stations. Simulations on this adaptive algorithm performed in [18] are summarised in table 2.2.

	5.75 ms packets 5.75 ms receiver delay		50 ms packets 50 ms receiver delay	
Approx. maximum number of conversations for insignificant packet loss.	0% data load	5% data load	0% data load	5% data load
	60	30	150	125

Table 2.2 64Kbps voice communication on Ethernet using the adaptive CSMA/CD algorithm, with silence suppression.

Packets of size 5.75ms and 50ms were examined, with an additional artificial delay of 5.75ms and 50ms respectively at the receiver to reduce variability. Packets that were not received within these times, were considered lost.

With the adaptive algorithm, the results indicated that the network could support about 60 conversations with insignificant packet loss for 5.75ms packets. Beyond this value the distribution of packet loss increases rapidly with additional conversations. With the same packet size, this increase occurred at about 40 conversations for the fixed packet length algorithm. With 50ms packets and 50ms artificial delay at the receiver, the number of conversations the network can support is greater since the effective capacity has been increased by reducing the per-packet overhead and the variable delay threshold is larger. It can be seen from table 2.2 that, with the addition of data, the synchronisation breaks down noticeably due to the asynchronous nature of the data.

To overcome the effect of data packets colliding with voice packets, voice packets could have a preemption header of length at least twice the end-to-end propagation delay, τ. Thus, if a voice and data packet collide, the data station can sense a collision and cease its transmission before any useful voice information has been transmitted. Prioritised variants of networks such as Ethernet have appeared in literature to provide voice traffic with priority over data [37,74,124]. While priorities are useful when the high-priority class constitutes a small fraction of total traffic, in most practical cases voice may form the larger fraction. Thus some form of synchronisation is still required for voice stations.

A simulation in [91] examined the effect of different *backoff algorithms* used to reduce the occurrence of collisions. Ethernet uses a binary exponential backoff algorithm for data. This backoff is obtained from a uniform distribution by computing a time equal to the product of the slot time (τ) and two raised to the power of the number of collisions already experienced by the packet. The overall effect of the algorithm is that as congestion increases, stations tend to delay longer amounts of time. This backoff algorithm provides a degraded service, until the data load, which is bursty in nature has decreased.

Two of the algorithms considered in [91] did not discriminate between voice and data. They were the random algorithm, where the backoff was computed from a uniform distribution with a mean specified as a simulation parameter, and the binary exponential algorithm. Voice was then discriminated against data traffic in the random voice/binary exponential data algorithm and the voice biassed/binary exponential data algorithm. The random voice/binary exponential data algorithm applied a random backoff to voice packets with a mean specified by simulation and binary exponential backoff for data packets. Hence while congestion exists in this network, data packets may be delayed a large amount until congestion subsides, whilst voice packets are only delayed up to a certain limit due to their time-constraint. The voice biassed/binary exponential data algorithm used exponential backoff for both voice and data. However, since the voice is often more time-critical than data, the exponent of the mean value used for computing the retransmission time for data packets is twice that used for voice packets.

The random algorithm was found to be impractical for a voice/data load since, once congestion became significantly large, the fixed mean backoff time was insufficient to handle transmission requests. The binary exponential, voice biased/binary exponential and random voice/binary exponential data produced similar results for the percentage of voice packets delivered.

A CSMA/CD bus was examined in [21,22], where short packetisation intervals were used to reduce delay. To achieve the minimum length packet on the Ethernet [78,122], several of these packets were multiplexed to form a large maxi-packet. Reviews of the various IEEE medium-access control protocols for voice/data integration can be found in [34,98]. An experimental Ethernet system in [26] reported results similar to the simulations [21,22,33,34,92] and derived the following approximation for the maximum number of voice users that a CSMA/CD network can support:

The medium is organised into slots whose length is twice the end-to-end propagation delay plus the time it takes to transmit a jamming signal. Activity on the medium is modelled to alternate between contention and transmission intervals. It is assumed there are N active stations. To reduce the total collision time, stations transmit at the beginning of a given slot with probability P during a contention interval. The probability that any station acquires the network in a given slot, A, is the probability that only one station attempted to transmit in that slot. This is given by

$$A = NP\,(1-P)^{N-1} \qquad \text{...(2.2)}$$

The mean length of the contention interval, W, in slots is [103]:

$$E[W] = \sum_{i=1}^{\infty} i\, P_r \text{ [i slots in a row with a collision or no transmission followed by a slot with one transmission]}$$

$$= \sum_{i=1}^{\infty} i\,(1-A)^{i}\,A$$

$$= \frac{1-A}{A} \qquad \text{...(2.3)}$$

In heavy traffic, the acquisition probability is maximised when P=1/N. As $N \to \infty$, $A \to (1/e)$ and $E[W] \to e-1$. The maximum utilisation of the channel, U, is the length of a transmission interval in proportion to the length of the transmission interval and the average contention interval E[W].

$$U = \frac{T}{T + E[W]\tau'} \qquad ...(2.4)$$

where T is the time it takes to transmit a packet and τ' is the duration of a slot. Asymptotically, as $N \to \infty$, U approaches

$$\lim_{N \to \infty} U = U_{asym} = \frac{T}{T + (e-1)\tau'} \qquad ...(2.5)$$

If there is no silence detection, each active station transmits packets periodically at a rate 1/F. For N_{talk} stations, the total packet rate is $N_{talk}1/F$. The asymptotic bound on the packet service rate is $(U_{asym}C)/b$, where C is the network capacity and b is the total number of bits per packet. Thus, since b=CT

$$N_{talk}1/F = \frac{U_{asym}C}{b} = \frac{U_{asym}}{T} = \frac{1}{T + (e-1)\tau'}$$

Significant packet loss will result if N_{talk} exceeds the asymptotic packet service rate. Hence N_{talk} should be limited by N_{max}, where

$$N_{talk} \leq N_{max} = \frac{1}{(T + (e-1)\tau')1/F} \qquad ...(2.6)$$

Although a CSMA/CD system can loose packets when N_{talk} is less than N_{max}, an approximation of the fraction of packet loss is given by

$$P_{loss} = \frac{N_{talk} - N_{max}}{N_{talk}} \qquad ...(2.7)$$

If silence detection is now enforced, N_{talk} becomes a random variable distributed by the following density function:

$$P[N_{talk} = n] = \binom{N_{active}}{n} \gamma^{n} (1-\gamma)^{N_{active} - n} \qquad ...(2.8)$$

where, of the N_{active} voice stations, N_{talk} are in a talkspurt, generating voice packets periodically. γ is the probability of a station being in a talkspurt. Thus, over a long period of time, the estimate of the fraction of packet loss from (2.7) and (2.8) is given by

$$P_{loss} = \sum_{i=N_{max}+1}^{N_{active}} \frac{i - N_{max}}{i} P[N_{talk} = i] \qquad \text{...(2.9)}$$

The approximation of packet loss in equations (2.7) and (2.9) from [26] compare well with simulations in [91] and [18] respectively.

It is apparent from past work that the CSMA/CD protocol is capable of supporting voice traffic, provided the network does not become overloaded. Once past saturation point, packet delay, variance and loss increase rapidly. However, this saturation point is relatively low, since CSMA/CD does not take full advantage of the voice traffic characteristic that packets are generated periodically during talkspurts. Some protocols which attempt to raise this saturation point are now briefly examined.

2.3.2 Variations Of CSMA/CD Which Utilise Speech Periodicity

In a speech network, as opposed to a data network, the short-term history of the network behaviour can help in estimating the traffic. Since the length of the average talkspurt is much longer than the voice packetisation time, there will be a strong periodic component in the activity on the network. Such correlation of voice traffic is usually not available in a data packet network, since interpacket data arrival times are often independent random variables.

An analytical and simulation study in [50] describes a channel access strategy that allows a station`s probability to transmit in a slot when it has a ready packet to vary in time, depending upon channel activity. For a network to maximise its performance, each station with a packet to send should attempt a transmission in a contention slot, with probability 1/N, where N is the number of stations with packets awaiting transmission [78]. Since each station does not have this global information, it estimates this number based upon channel activity.

Four algorithms were examined which took advantage of the fact that the average talkspurt is much longer than the frame time. They estimated the traffic based upon the three possible outcomes of a contention period, these being a silent slot, a collision, or a successful transmission. Using one of these algorithms, a station either incremented its estimate of the number of contending stations by one, decremented it by one, or left it unchanged, depending upon the activity in the previous slot of length τ. The initial value of each algorithm at a station when it has a ready packet was 1, i.e. it will transmit in the first available slot. After each slot, every station with a ready packet adjusts its estimate of the number of contending stations according to the activity of the previous slot and will transmit in the next available slot with probability of 1/(estimated number of contending stations).

None of the proposed algorithms expected estimates equalled the actual number of stations over a range of values.The algorithm with the best property as the number of stations increased, however, decremented its estimate when a slot was sensed idle and incremented it when another station successfully transmitted or the station encountered a collision. Simulations of these algorithms illustrated that the number of contending stations can change significantly within a frame time and yet the periodicities over a large period of time change more slowly. Hence the short-term behaviour and not the long-term should be monitored for voice. Thus, when the next packet was generated, the algorithms resumed with the estimate they used for the last successful transmission. Data stations, however, should use the long-term observations to predict the number of ready terminals.

Stations which use the MAC protocol described in [123] also monitor the network activity in order to calculate backoff delays. When a packet is involved in a collision, the station concerned calculates the backoff delay for the packet based upon the network activity level and the access delay already experienced by the packet. It was suggested in the paper that the network activity level should be updated periodically, perhaps every 200ms. As the network activity increased, the backoff delay got larger. However, as discussed earlier in [50], within a frame time the number of contending stations can change dramatically, although over many frame times the periodicities of voice packet generation by stations evolves more slowly. Hence the network activity value that a station uses in [123] may not be a true representation of the actual number of waiting stations. Unlike Ethernet, this protocol is sensitive to packet delays. It also has the advantage that it resolves collisions by prioritised grouping in a dynamic sense through packet delay rather than physical grouping [82], which may bear no resemblance to the network dynamics. The disadvantage of the schemes in [50,82,123] is that they only take advantage of voice packet periodicity in a probabilistic manner and not in any deterministic sense.

2.3.3 Random/Controlled Access For Voice Packets

Superior performance can be attained by using reservations based upon the periodicity of voice sources. The protocol modelled in [13] distributively organises voice packets from different stations into *strings*. It takes advantage of voice packet periodicity by deterministically assigning which station is to transmit next. An active voice station remembers which voice station periodically transmits before it (its predecessor) in a string. After this transmission is sensed complete, it then transmits. In the case where a predecessor does not transmit due to malfunction, the successor station to it will finally transmit after a timeout period. Collisions between established active voice stations in a string do not occur, since each station has a different predecessor station. The voice strings protect themselves from data interference by maintaining the carrier on throughout the string transmission. To maintain the carrier, stations whose packets are adjacent in the strings use a handshaking procedure to pass channel control on to the next active voice station in the string. A station will maintain the carrier either until the succeeding station handshakes, or it times out. A new voice conversation attempts to join a string by contending with other new voice stations at the end of a string.

Consequently, data packets sense the channel busy for the duration of a voice string and defer transmission attempts until it is complete. The protocol was simulated for a voice and data load and compared to the CSMA/CD protocol. The string protocol could support more conversations and, for a fixed voice load, a higher data packet throughput than the CSMA/CD protocol. This improvement with voice strings is due to more efficient scheduling and so better utilised bandwidth, although there is additional channel overhead in the handshaking procedure.

Using this protocol, data stations can be guaranteed a certain amount of channel capacity by voice stations recording the current number of voice conversations. Thus, when the voice capacity allocation is used up, voice stations will be unable to gain access. This may be suitable if silence suppression is not used but, if packets are only transmitted in talkspurts, it is undesirable to prevent the continuation of a conversation once it has commenced. To overcome this, lower voice bit rates could be used during overload, although protocol complexity would increase significantly. In particular, when a station has to move several places up a string due to various predecessor stations terminating, the packetisation interval may not be completed in time for handshaking with its new predecessor and, when the bit rate is increased after an overload period, care would be necessary with the timeout intervals. The advantage of this string protocol described is that its operation is independent of the number of voice stations in the network, unlike similar reservation schemes described in [1,28,112]. Protocol independence of the number of stations is important to provide greater flexibility in LANs and is essential for local radio networks where the number of stations can change quickly. Other similar string protocols for voice/data integration are described in [4,27].

A scheme analysed in [12] uses an ordered list of established voice stations in talkspurt. Established voice stations transmit their packets without conflict from the start of each voice frame. The remaining time of the frame is used for stations with new talkspurts to contend and join the ordered list. They achieve this through CSMA/CD. If the number of stations on the ordered list exceeds the maximum number of permitted transmissions for each frame, a round-robin service discipline is used to distribute capacity fairly among stations. Stations at the end of the list who did not transmit in the previous frame therefore have a higher priority in the next frame. The advantage of this scheme over the previous string protocol described is that it is more manageable in terms of fairness when congestion results without necessarily blocking a call once it has commenced. Instead, the overall fidelity of the voice calls is reduced. Data can be accomodated by the voice transmissions utilising only a fraction of the frame time.

The simulated protocol described in [75] prevents collisions among established voice stations by allowing each station to acquire a movable time-division multiplexed slot. All the packets from data sources and the first packets from voice sources in talkspurt use the CSMA/CD protocol. After the first transmission, a voice source transmits all of the samples it has accumulated whenever it acquires the channel. A voice source schedules its next transmission attempt one packetisation interval after its last successful transmission. An established voice station listens before transmitting and defers transmission if a station is currently transmitting. Once the channel is acquired, the station does not listen while transmitting and never terminates transmission prematurely. This is due to a preemptive header on all packets from established voice stations. The preempt interval is long enough for a data station or a new voice station to detect a collision, stop transmitting and have the effects of the transmission removed from the channel before the established voice user begins transmitting useful data. The preemption header thus allows an existing voice station to effectively reserve a slot in each packetisation interval.

Established voice stations interact in a TDM manner. An established voice source acquires the channel and periodically transmits in a slot. It may however be delayed by a new voice source or data station transmission. When this occurs, the periodic slot scheduled for the established voice station is shifted slightly. Additional voice samples are transmitted in the delayed slot to compensate for the shift. This is achieved by transmitting the extra samples that accumulated whilst waiting for a transmission to finish in an overflow region of its packet. To maintain synchronisation, all voice stations transmit a carrier during the overflow period of their packet, i.e. an established voice station takes no more time to transmit when it has been delayed than when it acquired the channel immediately. The size of this overflow area in a packet is determined by the maximum delay an established voice station can experience. This can be due to either a new voice station or a data station transmitting. Hence, in order that the overflow area is not too large, since it may be wasted capacity, data packet lengths must be confined. In the protocol described, data packet lengths were limited to the length of a voice packet. This can obviously be inefficient for file transfer if small voice frame times are used. This protocol has the advantage that collisions between existing voice stations and data/new voice stations do not affect the transmission synchronisation of the voice stations, unlike the adaptive CSMA/CD protocol described earlier in section 2.3.1. However, the price for this synchronisation is overhead in terms of preemptive packet headers for established voice stations. Further studies on this protocol are described in [43,44].

The performance of Demand-orientated TDMA (DTDMA) is analysed in [67]. In this scheme, the time axis is divided into slots (of length equivalent to a transmission time) and the first bit in each slot is designated as a tag which is reserved for voice slot assignment. At call setup, a voice station transmits a carrier in the tag bit on the incoming slot without sending any packet. The tag is also sensed at the same time. If there is no collision, then the slot is dedicated as that station's voice slot for the remainder of the conversation. To achieve this, the station continues to transmit its carrier in the tag bit for every frame thereafter until the call terminates. If a collision does occur while attempting to acquire a slot, the voice station uses a retry procedure. If no free slots exist, the call is deleted from the system. Data packets are transmitted after the tag bit in the slots whose stations are not in talkspurt. The disadvantage with this protocol is that if the voice traffic is heavy and there is little data traffic, most of the tag bits will be marked by established voice stations, and so large amounts of capacity will be wasted due to non-established voice stations being unable to transmit in the silence periods of established conversations. Capacity is also wasted by the tag field, especially if the end-to-end propagation time is large. Another disadvantage with this protocol is that data packets must have lengths of less than one slot time.

2.3.4 Time Partitioned Speech Protocols

Time partitioned protocols specify a mapping from the time axis into subsets from the set of all possible waiting messages. The protocol simulated in [124] uses a dual channel architecture, where one channel is dedicated for contention resolution and the other for data transmission. On the contention channel, a prioritised 1-persistent algorithm is used, with slots of two times the end-to-end propagation delay (τ). At the beginning of a packet transmission on the data channel, all stations with a ready packet transmit a carrier in the first slot on the contention channel. If a collision occurs, all data stations backoff and any voice stations involved continue to resolve amongst themselves to decide who will transmit next on the data channel. They do this by transforming their packet waiting times into an m-bit binary word. If a packet waiting time is greater than can be represented by an m-bit word, it is represented as all 1`s. Each voice station maps its m-bit word (MSB first) onto the next m slots of the contention channel by transmitting a busy tone for '1' and not transmitting otherwise. If any of these m slots is detected busy by a voice station that does not transmit a busy tone in that slot, then this station defers its transmission attempt to the next start of a packet transmission on the data channel because there are other voice stations contending with longer packet waiting times. If there is still no resolution after the m-slot period, the contention is based on a random binary tree algorithm. The simulation results of this algorithm showed a dramatic improvement compared to the CSMA/CD protocol at the expense of requiring another channel.

The time partitioned protocol Virtual Time CSMA/CD (VT-CSMA/CD) [83,84] also achieves time separation through packet arrival times. The implementation of VT-CSMA/CD requires that each station be equipped with a real time clock and a virtual time clock. When a station`s virtual time clock equals the arrival time of its packet, it stops its virtual time clock and transmits the waiting packet. Upon sensing the transmission, all other stations also stop their virtual time clocks. When the transmission is complete, all stations simultaneously start their virtual time clocks again at a rate of η times that of the real time clocks, where $\eta>1$. When the virtual time clocks catch up with the real time clocks, they are slowed down to that of the real time clock rate. If a collision occurs, stations randomly reschedule their packet arrival times to future virtual times. A variation of this protocol called Reservation-VT-CSMA/CD (R-VT-CSMA/CD) [64] takes advantage of voice packet periodicity. In this protocol, when a voice source successfully transmits a packet, it transmits all its successive periodic packets in the talkspurt at the same time within the frame, thus effectively reserving the slot. Priority for voice over data is achieved by each station having two virtual time clocks, one for voice and the other for data. Only when the voice virtual time clock is up-to-date (i.e. there is no backlog of voice packets) can the data virtual time clock advance. In both the previous protocol and this one, m and η are design parameters. Both are fair, in that they clock packets out onto the channel in a FCFS manner. However, in the event of congestion, many packets would have an m-bit word full of 1`s in the previous protocol and this fairness may break down. In the virtual time clock approach however, stations will maintain their values and any lost packets occurring due to excessive waiting times will be fairly distributed.

The m-bit protocol will be inefficient when packet transmission times are short and the end-to-end propagation delay is large. With this scenario, R-VT-CSMA will be more advantageous since it takes advantage of voice packet periodicity to reduce the collision rate.

A related protocol described in [60,61], proposes the use of a time window mechanism to determine the enabled set of stations. This protocol maintains an interval of time or time window in the past and all stations which have a message that arrived during this time interval are enabled. This window size may be either fixed, as in [63], or dynamic, based upon packet delay [53]. However, unlike the virtual time clock protocol, these protocols do not take advantage of the voice packet periodicity.

2.4 Summary

In this chapter, previous research on MAC protocols for supporting packet voice in contention type LANs has been reviewed. The objective of the protocols was to minimise the contention time between stations in order that the user requirements for packet voice could be met. Since the length of the average talkspurt is much longer than the voice packetisation time, there is a strong periodic component in the activity on the network. To improve throughput, the MAC protocols utilised this characteristic. They broadly fall into two classes, those that transmit with a probability based on channel activity and those which use either explicit or implicit reservations.

In the first class, stations randomly transmit when they have a ready packet with a derived probability. The probability of transmission depended upon recent channel activity. In a speech network as opposed to a data network, the short-term history of the network behaviour can help in estimating the traffic. The disadvantage of such schemes is that they only take advantage of voice periodicity in a probabilistic manner and not in any deterministic sense.

The second class of protocols took advantage of periodic voice packets through either implicit or explicit reservations. These protocols have a superior performance due to a reduction in the collision rate and hence wasted channel time. However, the efficiency of the various reservation protocols depends upon the ease in which a station can make a reservation and the overhead necessary to maintain it.

CHAPTER 3

FIXED PACKET LENGTH VOICE MAC PROTOCOLS FOR LANS

3.1 Scheduling

A MAC protocol functions not only as an arbitrator between stations, but also as a distributed scheduler. This scheduling arranges the transmission order of waiting packets that are spread amongst the network`s stations. It has been shown that the scheduling function significantly effects the packet delay distribution [61]. For time constrained communication, since the distribution of packet delay is important, the scheduling mechanism should be based upon packet arrival times.

Previous research into MAC protocols examined in chapter 2 falls into two classes, those that transmitted with a probability based on network activity and those that used reservations, which generally provide a superior performance. The efficiency of the reservation protocols is dependent upon the way in which reservation is achieved and how it is maintained throughout the talkspurt/call duration. No protocol can arrange a perfect schedule of packet transmissions due to the spatial distribution of the stations, since stations cannot in general form a queue without using part of the channel to exchange protocol information. However, in the VT-CSMA protocol, in the limit as $a \rightarrow 0$, where a is the ratio of the end-to-end propagation delay to the packet transmission time, the overhead due to the spatial distribution of stations vanishes to produce an ideal M/G/1 queue [84]. This is not the case for the other CSMA protocols of section 2.3.

Section 3.2 examines in detail the R-VT-CSMA/CD protocol, that was briefly mentioned in chapter 2 and which is based upon implicit reservations. Its performance has been improved by introducing a new collision resolution mechanism and a voice bit loss, rather than packet loss approach. The time-bandwidth efficiency of the channel is then analysed. Based on this analysis, in section 3.3 a new protocol is proposed which uses the same basic implicit reservation scheme, but reduces the wasted channel capacity between transmissions. The dual channel architecture proposed, improves the throughput on the channel significantly compared to the R-VT-CSMA/CD protocol by allowing packet transmissions to occur in parallel with packet contention.

3.2 The Reservation Virtual Time CSMA/CD Protocol

The time partitioned Virtual Time CSMA protocol achieves separation of ready packets based upon their arrival times. In the synchronous version, the channel time is divided into slots, where one slot is the end-to-end propagation delay on the network, τ. Only at the beginning of each slot can transmissions commence. It can operate with or without collision detection. VT-CSMA requires that each station has two clocks, a real time clock and a virtual time clock. In the synchronous scheme, the real time clock advances in discrete steps at the rate of 1 slot/slot. The advancement of the virtual time clock however, is a function of the channel activity.

Whenever a packet is generated at a station, the current real time is stored in an arrival time register at the station. Initially both clocks start off together, advancing in discrete steps at the rate of 1 slot/slot. However, when a station`s virtual time clock exceeds a packet arrival time in its arrival time register, it stops its virtual time clock and transmits the waiting packet at the beginning of the slot. When all the other stations sense the transmission, they also stop their virtual time clocks. When all stations sense the transmission is complete, they start their virtual time clocks again from where they had stopped. The virtual time clocks advance simultaneously in discontinuous steps at the integer rate of η slots/slot, where $\eta>1$. This process continues until the virtual time clocks catch up with the real time clocks, at which time the virtual time clocks are slowed down to that of the real time clock rate of 1 slot/slot.

It can be seen that packets are transmitted in a First-Come First-Served (FCFS) manner which is independent of the channel activity when a packet arrives, unlike the other CSMA protocols. In the event of a collision, transmission attempts are rescheduled to a future virtual time. A stability study in [76] showed that under heavy traffic, most transmissions in VT-CSMA are resolved through packet arrival times, whereas for other CSMA protocols, most packets are separated through the retransmission algorithm. This feature of VT-CSMA is very desirable for packet voice due to the importance of the packet delay distribution.

If a packet arrival time is separated by at least $\eta\tau$ from any other stations`s packet arrival time, the packet transmission will be successful. During a talkspurt, a voice station will generate packets periodically. If a station can remember the virtual time at which it transmitted the packet in the frame, and transmits its next packet in the same position, the transmission will most likely be successful since all the other established talkspurts will have their own reservation slots and so be interleaved on the virtual time axis. The $\eta\tau$ can then be regarded as an effective reservation slot. This idea was realised in the Reservation Virtual Time CSMA/CD (R-VT-CSMA/CD) protocol [64] by each station having a buffer which stores its reservation virtual time in the frame. For example, if the first transmission results in a collision, the transmission attempt is randomly rescheduled to a future virtual time and this random offset from the arrival time stored in the reservation buffer. If the next transmission attempt also results in a collision, a new random offset is added to the value already in the reservation buffer. This process continues until the packet has been successfully transmitted. When the station`s next packet arrives, its arrival time will be increased by the value in the reservation buffer to form a new pseudo arrival time. The station attempts to maintain this effective reservation slot for the remainder of the talkspurt.

In the R-VT-CSMA/CD protocol, a packet is discarded and hence lost if it has not been transmitted before the station`s next packet arrives. The voice packet access delay is thus bound by the length of a voice frame. Hence the value in the reservation register is a value between 0 and the frame length, F.

Typically, the transmission time of a packet is larger than $\eta\tau$ and so it can be seen that the maximum possible number of packet transmissions in a frame time is less than the effective number of reservation slots within the frame. Hence new talkspurts will quickly find a reservation slot. The protocol described in [75] required that reservation times be separated by packet transmission times and so, under heavy load, a voice station in this scenario will not easily find a free reservation slot.

Obviously collisions can still occur with established voice stations in R-VT-CSMA/CD when one or more new talkspurts select the same reservation slot. These new voice stations then randomly reschedule their arrival times to another virtual time within the frame. The protocol described in [64] does not specify what procedure the established voice station then takes. The station could either discard the packet and maintain the same reservation slot for its next packet or reschedule the packet to a new future virtual time. Figures 3.1a and 3.1b define the R-VT-CSMA/CD protocol at a particular station, where established stations randomly reschedule collided packets.

Simulation results have shown that the version which randomly reschedules the packets of established stations to future virtual times, can support a greater number of voice conversations than the case where they are discarded. An *avalanche* effect, whereby retransmissions from established stations start colliding with other established voice stations, does not occur until the load is heavy, since the ratio of the number of reservation slots to packet transmission times within a frame is generally large.

During the research another mechanism was examined whereby if an established voice station`s packet is involved in a collision, that station persists transmitting a jamming signal until the one or more new stations have realised there was a collision and have backed off. The established station then retransmits its packet. This is desirable since it no longer needs to find a new reservation slot, with the possibility of interfering with another station. This appeared to be satisfactory since each established voice station effectively holds its own $\eta\tau$ virtual time window, for each frame in the talkspurt.

```
while in talkspurt do
    begin
        while packetisation interval incomplete for current packet do
            assemble packet;
        commence packetisation interval for next packet;
        add arrival time offset;
        repeat
            begin
                wait until virtual time clock ≥ arrival time;          {fig. 3.1b}
                transmit packet;
                if collision then
                    transmit jamming signal;
                    compute backoff;
                else                                                   {successful transmission}
                    complete successful transmission;
            end
        until successful transmission
        calculate arrival time offset;
    end
```

Figure 3.1a. Pseudo code describing the R-VT-CSMA/CD protocol, where established voice stations reschedule collided packets.

```
begin
    while virtual time < arrival time do
        begin
            while channel is busy do
                begin
                    stop virtual time clock;
                    wait τ seconds;
                end
            virtual time = min{virtual time + ητ, real time};
            if virtual time < arrival time then
                wait τ seconds;
            if following packet has arrived then
                discard original packet;
        end
end
```

Figure 3.1b. *Wait until virtual time clock ≥ arrival time* pseudo code.

However, on closer inspection, this is not always the case. Figure 3.2 illustrates the scenario where in the first frame stations 1 and 2 arrive with new talkspurts. The virtual time clocks are advancing at the same rate as the real time clocks when the arrival time of station 1's packet is exceeded. This is then transmitted at the beginning of the next slot, thus preventing any further virtual time clock update at the stations. Once the transmission is complete, the virtual time clocks at the stations update in unison at the rate of $\eta\tau$ slots/slot. During this first virtual time advancement of $\eta\tau$, station 2's packet arrival time is exceeded and so is transmitted at the beginning of the next slot. Hence both stations 1 and 2 were successful in their first attempt; they will now transmit at these virtual time positions in the next frame.

At the beginning of the next frame, station 3 commences a talkspurt. The virtual time clocks pass station 3's arrival time first while advancing at rate unity and so station 3 then transmits. Once the transmission is complete, the virtual time clocks advance at the rate of $\eta\tau$ slots/slot. This time, the arrival times of the second packets from the established stations 1 and 2 are both encompassed by the time window. Since they have both successfully transmitted in the previous frame at the same virtual time position, they believe the collision involves a new voice station. Obviously if they both persist, much channel time will be wasted, since this condition will exist until one of the stations ceased being in talkspurt.

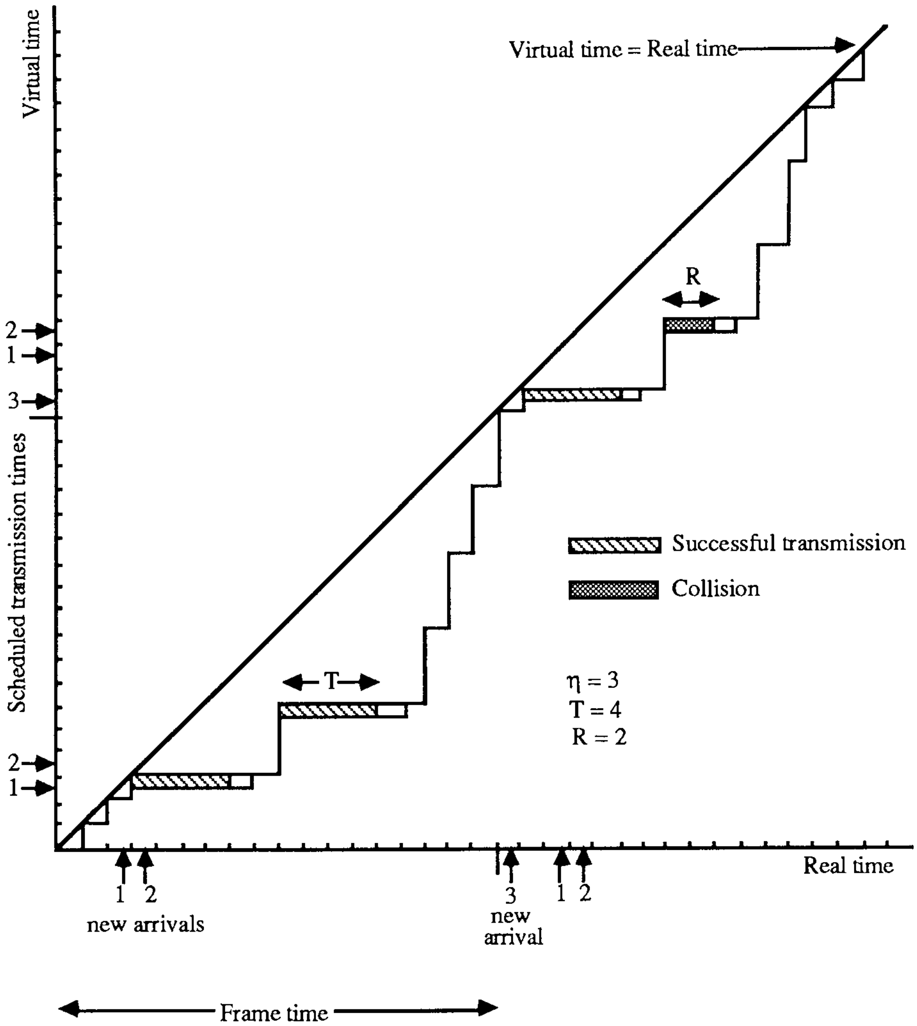


Figure 3.2. Collisions can occur between established voice stations in R-VT-CSMA/CD.

The policy decided upon was called established station preemption. Here, if an established station`s transmission is involved in a collision, it persists in transmitting its jamming signal for a fixed time period longer than the jamming period of a new station and then attempts its transmission again. Any new stations involved in the collision randomly reschedule their packet arrival times. If the established station`s transmission continues to be involved in a collision after the jamming signal, it knows that this involves another established station. Both established stations would then backoff and attempt to find a new reservation slot for their respective packets.

Priority for voice over data is achieved in R-VT-CSMA/CD by all stations having two virtual time clocks, one for voice and one for data. Only when the voice virtual time clocks are up-to-date, i.e. there is no backlog of waiting voice packets can the data virtual time clocks advance.

3.2.1 Ethernet-Type Compatibility of R-VT-CSMA/CD

In order for voice stations to have priority over data stations in R-VT-CSMA/CD, data stations must not interrupt the voice virtual time clock update, since there may be voice packets waiting to be transmitted. To achieve this a carrier can be maintained during the virtual time clock update.The carrier could be transmitted by the last transmitting voice station. When another station`s packet arrival time is then exceeded, this station could transmit a carrier, thus relinquishing control of the channel from the previous station. This collision handshaking procedure would be adequate if no collisions ever occurred. For example, if a station was transmitting a carrier after its transmission and it then sensed a collision handshake, it then removes its carrier. But it may have been two or more stations that handshaked and so a collision will result. This collision time will increase the virtual time lag of the real time clock further and so virtual time clock update is still required. However, which of the stations involved in the collision should resume the carrier?

To overcome this problem, a centrally located virtual time controller could have the responsibility of always maintaining the carrier after either a successful transmission or a collision, until the virtual time clocks have caught up with the real time clocks. Consequently, data stations on the channel will sense the channel busy during each voice transmission and clock update and defer depending upon the CSMA protocol. At the beginning of each data packet transmission, the voice virtual time clocks stop updating at unity. Once the transmission is complete, the voice virtual time will now be lagging behind the real time. The controller could transmit its carrier after every data packet transmission to update the voice virtual time. A less complex solution is for the controller to update the virtual time clocks after the voice virtual time lag exceeds a certain limit. Once the update is complete, data stations are free to transmit until this limit is again exceeded. A schematic and timing diagram of this idea are illustrated in figures 3.3 and 3.4.

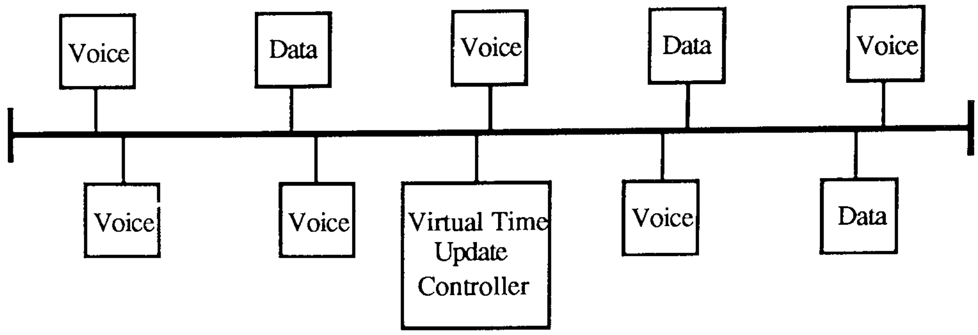


Figure 3.3. Schematic of Ethernet-type compatible R-VT-CSMA/CD

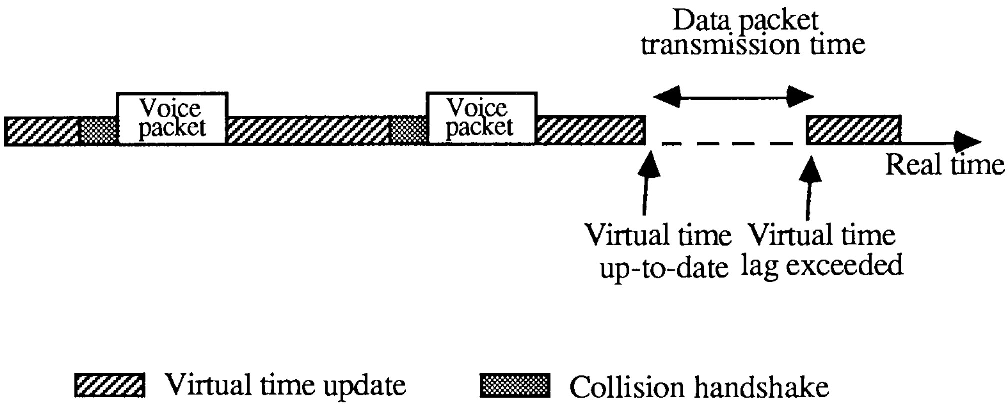


Figure 3.4. Timing diagram of Ethernet-type compatible R-VT-CSMA/CD. Virtual time update with the carrier is achieved through the controller.

3.2.2 Throughput Approximation Of The R-VT-CSMA/CD Protocol

The following analysis estimates the maximum number of voice stations a network can support without adverse speech quality when using the synchronous R-VT-CSMA/CD protocol with established user preemption. Each conversation on the network consists of two voice stations paired together alternating between talkspurt and silent periods. It is assumed that when one voice station is talking, the other is silent. A total of N voice stations on the network, i.e. N/2 conversations have been considered alternating between talkspurt and silent periods. This can be statistically modelled as a Markov chain, as shown in figure 3.5, where λ^{-1} and μ^{-1} are the mean lengths of the talkspurt and silence periods respectively [7].

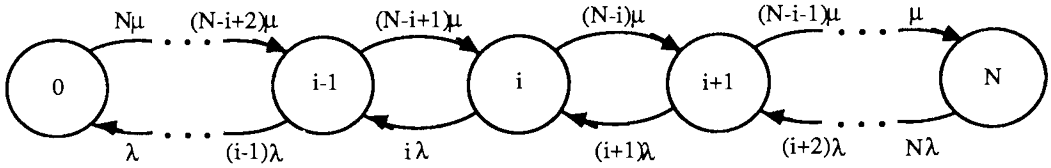


Figure 3.5. Talkspurt-Silence Markov chain

The model state is the number of voice stations that are in talkspurt. It is a birth-death process, whereby the distribution of K, the number of voice stations in talkspurt, is given by:

$$P_k = P_0 \prod_{j=0}^{k-1} \frac{(N-j)\mu}{(j+1)\lambda}$$

$$= P_0 \left(\frac{\mu}{\lambda}\right)^k \frac{N!}{k!\,(N-k)!}$$

where P_0 is the probability of being in state 0 of the Markov chain, i.e. no voice station is in talkspurt. This is equal to

$$P_0 = \left(\frac{\lambda}{\mu+\lambda}\right)^N$$

$$\therefore \quad P_k = \left(\frac{\lambda}{\mu+\lambda}\right)^N \left(\frac{\mu}{\lambda}\right)^k \frac{N!}{k!\,(N-k)!} \qquad \text{...(3.1)}$$

Let the frame be of length F seconds and out of the N voice stations on the network, K of them are in talkspurt in the current frame. It is assumed that voice stations only change between talkspurt to silence or vice versa at the beginning of a voice frame. Using equation (3.1), the mean number of voice stations in talkspurt is given by:

$$E(K) = \sum_{k=1}^{N} P\{K=k\}k$$

$$= \frac{\lambda^{-1} N}{\lambda^{-1} + \mu^{-1}} = \frac{\mu}{\mu+\lambda}.N \qquad \text{...(3.2)}$$

Also, let J be the number of stations which were silent in the previous frame but which are in talkspurt in the current frame. The probability that J=j is

$$P\{J=j\} = \binom{N}{j}\left(\frac{\mu^{-1} P_{01}}{\lambda^{-1}+\mu^{-1}}\right)^{j}\left(1-\frac{\mu^{-1} P_{01}}{\lambda^{-1}+\mu^{-1}}\right)^{N-j} \qquad ...(3.3)$$

where P_{01} is the probability that a station silent in the previous frame is in a talkspurt in the current frame. Assuming that the silent periods are exponential, we have [38]

$$P_{01} = 1 - e^{-\mu F} \qquad ...(3.4)$$

Thus the average value of J in (3.3), E(J), can be written as

$$E(J) = \sum_{j=1}^{N} P\{J=j\}j = \frac{\mu^{-1} P_{01}}{\lambda^{-1}+\mu^{-1}} .N$$

$$= \frac{\lambda P_{01}}{\mu+\lambda} .N \qquad ...(3.5)$$

Since the ratio of the number of reservation slots to maximum number of possible packet transmissions in a frame is large, it is assumed that stations which were in talkspurt in the previous frame found a reservation slot and so are established. Therefore, from (3.2) and (3.5), the number of established stations is

$$E(C) = E(K) - E(J) = N.\left(\frac{\mu - \lambda P_{01}}{\mu+\lambda}\right) \qquad ...(3.6)$$

If a packet transmission time is T slots in length, where a slot is equal to the end-to-end delay, τ, then, whenever the virtual time clocks are running at rate η, the backlog is decreasing at a rate of η-1. This can be seen from figure 3.6. Therefore, the total amount of time that the k stations transmit their packets and update the virtual time clocks is given by:

$$\text{Total time for k transmissions and updates} = kT + kT/(\eta-1)$$

$$= kT(\eta/\eta-1) \qquad ...(3.7)$$

In order to calculate the maximum number of stations a network can support without significantly affecting voice quality, it is assumed that significant packet loss does not occur if the following inequality holds:

$$E(K).T.\left(\frac{\eta}{\eta-1}\right) \quad + \quad E(J).\text{(Average contention time for each new talkspurt)} \quad \leq \quad F \qquad ...(3.8)$$

where E(K) and E(J) are given by (3.2) and (3.5) respectively. Qualitatively (3.8) states that the total amount of time used for packet transmissions in a frame, plus the total amount of contention time for new talkspurts generated in the frame, must not be greater than the frame length.

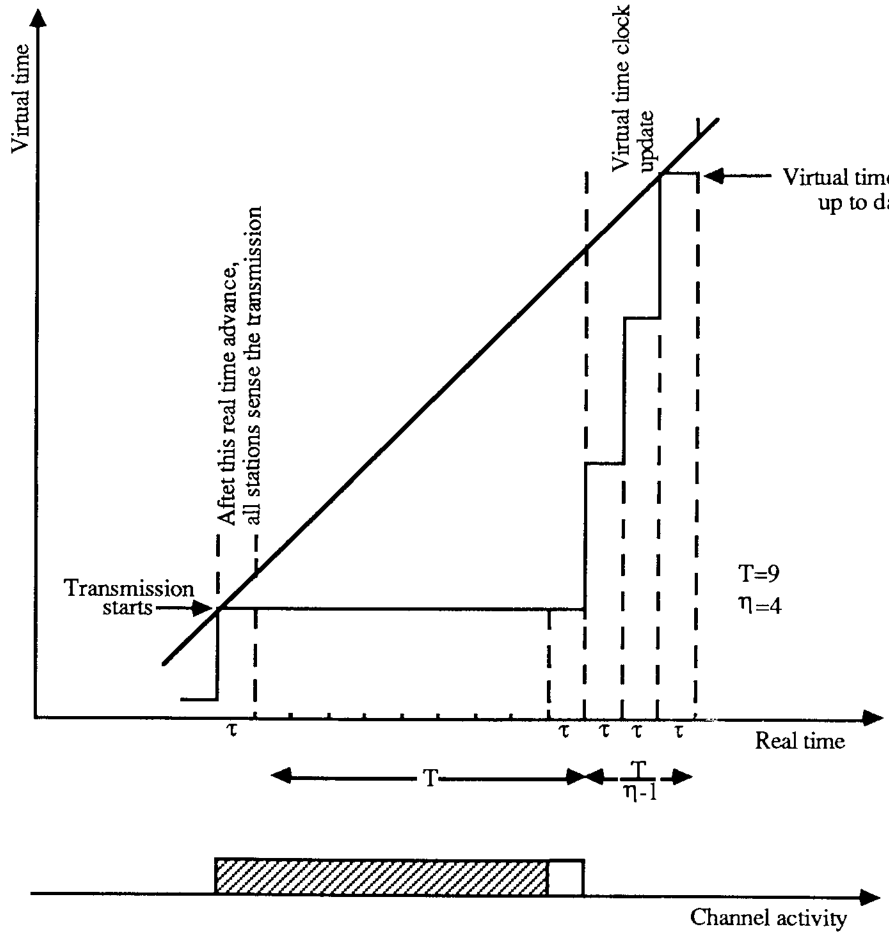


Figure 3.6. Virtual time clock update for synchronous VT-CSMA.

It now remains to calculate the amount of contention time a new talkspurt requires to find a reservation slot. To calculate the capacity of supportable stations, the virtual time clock will either have ceased for a transmission, or it will be advancing at rate η. However, the virtual time lag behind the real time clock must be stable, e.g. if at the end of each frame it just manages to catch up before another transmission commences. The number of reservation slots in the frame is then $\upsilon = F/\eta\tau$. Hence the probability that an existing station is in a slot is $E(C)/\upsilon$, when $E(C) \leq \upsilon$.

When an established station transmits, it knows it will never collide with another established station, except in the case illustrated in section 3.2. Since in the majority of cases collisions with established stations involves only new stations, the scenario where two established stations collide is ignored. The established station preemption method is assumed in the event of collisions. When a new talkspurt station transmits, it is either successful with probability P_s, or it is involved in a collision with probability $1-P_s$. The talkspurt arrivals can be approximated by a Poisson process [52].

Let $\lambda_t = J/\upsilon$ = arrival rate of new talkspurts per reservation slot in a frame.

When a collision occurs, new stations randomly reschedule their retransmissions. It is still assumed that the arrival process is Poisson with mean λ_t. With a slotted model, the probability that the transmission from a new station in talkspurt is involved in a collision is approximated by:

$$1 - P_s = \frac{C}{\upsilon} + \left(1 - e^{-\lambda_t \eta\tau}\right) - \frac{C}{\upsilon}\left(1 - e^{-\lambda_t \eta\tau}\right) \quad ...(3.9)$$

The first term on the RHS of (3.9) is the probability that the new station will collide with an existing station, the second term is if it collides with one or more other new stations and the third term being the mutual probability. Hence the total amount of time wasted by a new voice station in trying to find a reservation slot is given by:

$$E\{\text{Contention time for one station}\} = \sum_{i=0}^{\infty} i\zeta P_s (1-P_s)^i$$

where ζ is the wasted channel time (jamming signal and propagation delay) when a collision occurs. Hence the contention time necessary for all J new voice stations is

$$E\{\text{Contention time} \mid J=j\} = j\sum_{i=0}^{\infty} i\zeta P_s (1-P_s)^i \quad ...(3.10)$$

This can be rewritten as

$$E\{\text{Contention time} \mid J=j\} = \frac{j\zeta(1-P_s)}{P_s}$$

Hence the total time for contention resolution, including virtual time clock update is given by

$$E\{\text{Contention time}\} = \left(\frac{\eta}{\eta-1}\right)\sum_{j=1}^{N} j\zeta\,\frac{(1-P_s)}{P_s}\,P\{J=j\}$$

$$= \left(\frac{\eta}{\eta-1}\right)\zeta\,\frac{(1-P_s)}{P_s}\,N.P_{01}\,\frac{\lambda}{\lambda+\mu} \qquad ...(3.11) \quad \text{Using (3.5)}$$

Therefore, using (3.8), to avoid significant packet loss with the R-VT-CSMA/CD protocol, the following equation must be satisfied:

$$E(K)\,T\left(\frac{\eta}{\eta-1}\right) + E(J)\,\zeta\,\frac{(1-P_s)}{P_s}\left(\frac{\eta}{\eta-1}\right) \leq F$$

which can be reduced to

$$N\left(\frac{\eta}{\eta-1}\right)\left(\frac{1}{\lambda+\mu}\right)\left(\mu T + \lambda P_{01}\,\zeta\,\frac{(1-P_s)}{P_s}\right) \leq F \qquad ...(3.12)$$

where $1-P_s$ is given in (3.9). The percentage of lost voice can now be approximated using (2.7) and (2.8), where N_{max}, the maximum number of supportable voice stations is calculated from (3.12).

3.2.3 Simulation Results Of The R-VT-CSMA/CD Protocol

The synchronous R-VT-CSMA/CD protocol was simulated on an error-free 10Mbps LAN with an end-to-end propagation delay of 10μs. A voice frame time of 7.5ms was used i.e. the packetisation delay and access delay will be bound by 15ms. If the voice connection has a maximum delay of 300ms, then 270ms is available for the interconnection of LANs. Two versions of the protocol were examined:

(i) All voice and data stations were assumed to have virtual time clocks in order that data stations only transmitted on the channel when the voice virtual time clocks were up-to-date with the real time clocks. This is referred to as the standard protocol.

(ii) Ethernet-type compatible protocol. In this case all data stations were assumed to access the channel using the CSMA/CD protocol. Voice stations maintained their priority over data stations by transmitting a carrier during the virtual time clock update, as described earlier in section 3.2.1.

Both versions updated their virtual time clocks at the end of a data packet transmission. The value of virtual time clock update speed, η, is a function of arrival rate and transmission time etc. [76]. For each protocol, the optimal value was determined through simulation for 2% voice loss. Further details of the network parameters and simulation program are given in chapter 6.

Initially only a voice load was considered. Figures 3.7 and 3.8 illustrate the percentage of packets lost with no established station preemption, i.e. whenever an established station is involved in a collision it reschedules its transmission time. Established stations maintain their reservation position in the frame until either they are involved in a collision, or the talkspurt finishes. Figure 3.7 shows the case where uniform collision backoffs are used. The upper limit of the uniform distribution was dependent upon the amount of time a packet had left before the stations next packet arrived to discard it. A small offset was added to this maximum to prevent a packet that is near to its time constraint from retransmitting in every possible slot.

Figure 3.8 uses a binary exponential backoff with a maximum upper limit of one frame time. Each time a collision occurred, the reschedule time was taken from a uniform distribution between 0 and τ multiplied by 2 to the power of the number of previous collisions the packet has experienced. It can be seen that the protocol with the binary exponential backoff can support a greater number of conversations compared to the uniform backoff. With the uniform backoff, if a packet is involved in a collision and is rescheduled to a time near the upper limit of the uniform distribution, it is then highly likely that this packet, if involved in another collision, will be discarded as a result of exceeding the time constraint. A similar result has since been discovered in [73].

Figures 3.9 and 3.10 illustrate the packet delay for the uniform and exponential backoffs respectively. Both protocols maintain the packet access delay within one frame time i.e. packetisation delay + access delay $\leq$ 2 frame times. However, the delay standard deviation is larger in the uniform backoff due to the large reschedule times a packet can experience. As the voice load increases, stations must wait longer for the virtual time clocks to exceed the packet arrival times. Thus the packet access delay tends towards one frame time and so the deviation in delay decreases. For both protocols, the Ethernet-type compatible has an inferior performance due to the wasted capacity in the collision handshaking phase.

Figures 3.11 and 3.12 show the R-VT-CSMA/CD protocol with the established station preemption mechanism. The number of supportable conversations has increased since established stations no longer have to find another reservation slot when they collide with new stations and so then possibly interfere with other established stations. It can be seen from figure 3.11 that the approximation for the maximum supportable number of conversations from (3.12) is a little optimistic. This is predominantly due to three reasons:

(i) Not all stations that contended for a reservation slot in the previous frame were successful, i.e. the actual number of new contending stations is larger than estimated.

(ii) Packets can still be lost even when the inequality in (3.12) is satisfied, due to packet retransmissions exceeding the time constraint.

(iii) The throughput analysis ignored the fact that established stations could collide. This would have a two-fold detrimental effect in that channel time would be wasted in the extra jamming signal and then the rescheduled packets would have to use time contending for a new reservation slot.

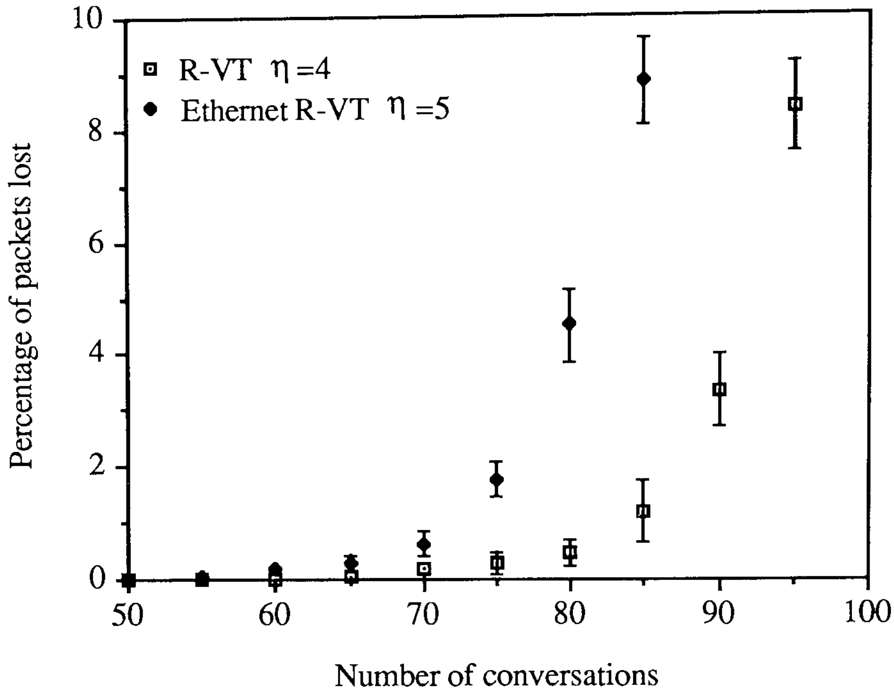


Figure 3.7 Percentage of packets lost using R-VT-CSMA/CD with a uniform collision backoff.

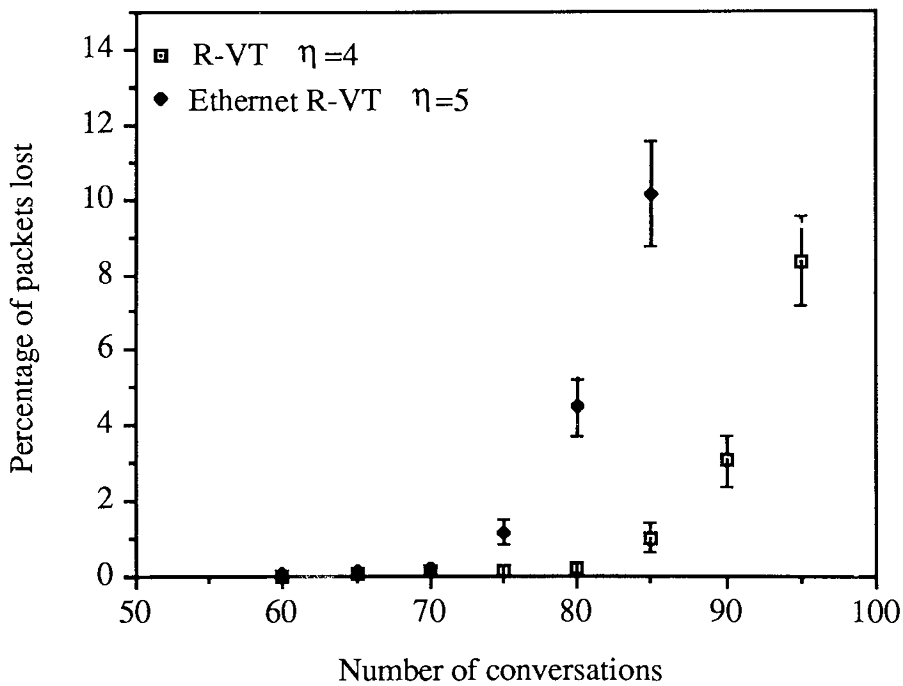


Figure 3.8 Percentage of packets lost using R-VT-CSMA/CD with a binary exponential collision backoff.

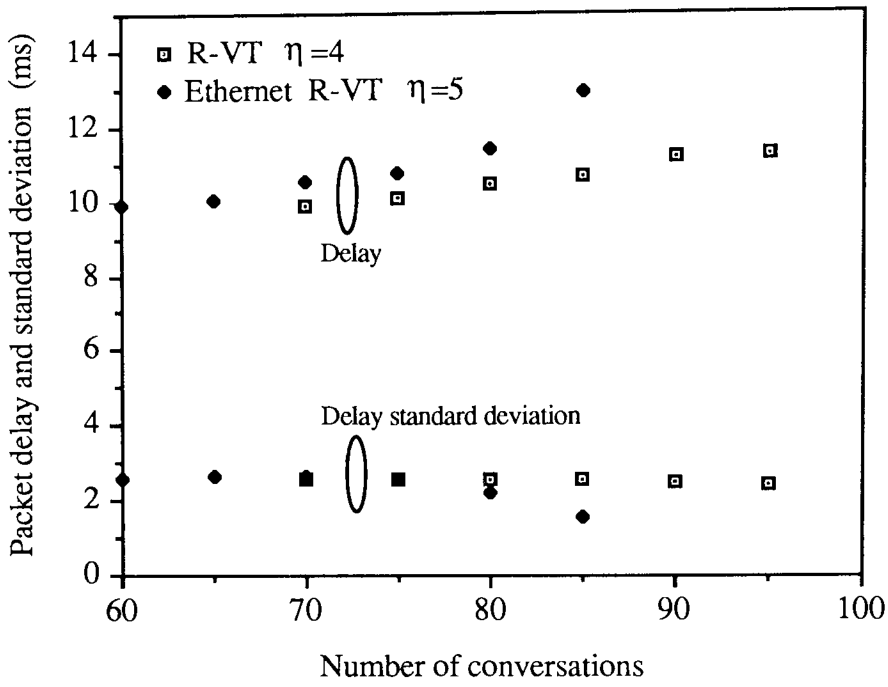


Figure 3.9 Packet delay and delay standard deviation using R-VT-CSMA/CD with a uniform collision backoff

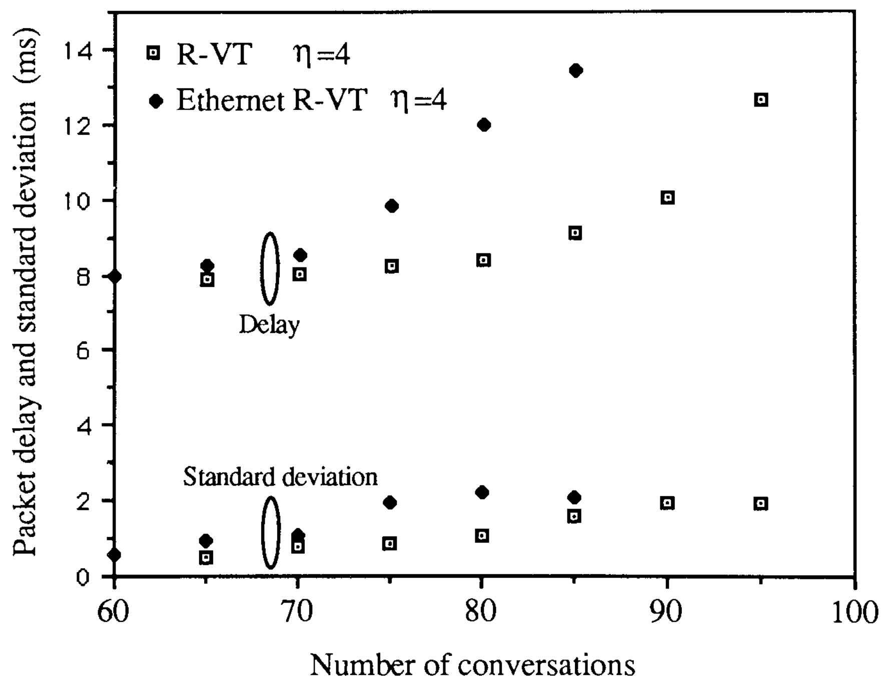


Figure 3.10 Packet delay and delay standard deviation using R-VT-CSMA/CD with a binary exponential collision backoff.

In the R-VT-CSMA/CD protocol, a packet is discarded if its access delay exceeds a frame time. However some of the bits at the end of the packet will not yet be one frame time old and so they need not be discarded. For example, if packet 1 from a station is ready for transmission at real time R, then packet 2 will be ready for transmission at real time R+F. Two possible scenarios exist at the transmission time for packet 1:

(i) If packet 1 is successfully transmitted at real time R+t, where t<F, no bits from packet 1 will be lost and the station maintains the same reservation slot in the following frame for packet 2.

(ii) If packet 1 is successfully transmitted at real time R+t', where t'>F, after real time R+F the oldest voice bits from packet 1 are replaced by the oldest voice bits from packet 2. In this condition, packet 2 is also ready for transmission, since its packetisation interval is complete. Packet 2 can then be *piggybacked* onto the transmission of packet 1. The stations next packet will be ready for transmission at real time R+t'+2T+F, where T is the transmission time of a packet.

The advantage of this bit loss scheme is that if a complete packet has to wait longer than one frame to access the channel, only the bits older than two frame periods and not the complete packet are discarded. Figures 3.13 and 3.14 illustrate the R-VT-CSMA/CD protocol with established station preemption and bit loss. A comparison between the packet loss and bit loss approaches are given in table 3.1. The N_{max} value is the number of conversations the network can support with packet/bit loss of 2%. It can be seen that the bit loss protocol can support a greater number of conversations whilst also having superior clip statistics, since only bits which are older than two frame times are discarded.

	N_{max}	Clip Length (ms)		
		Mean	Std dev	Max
Packet loss	96	14.5	19.9	195
Bit loss	101	3.9	12.1	181

Table 3.1 Clip statistics for R-VT-CSMA/CD with 2% packet and bit loss.

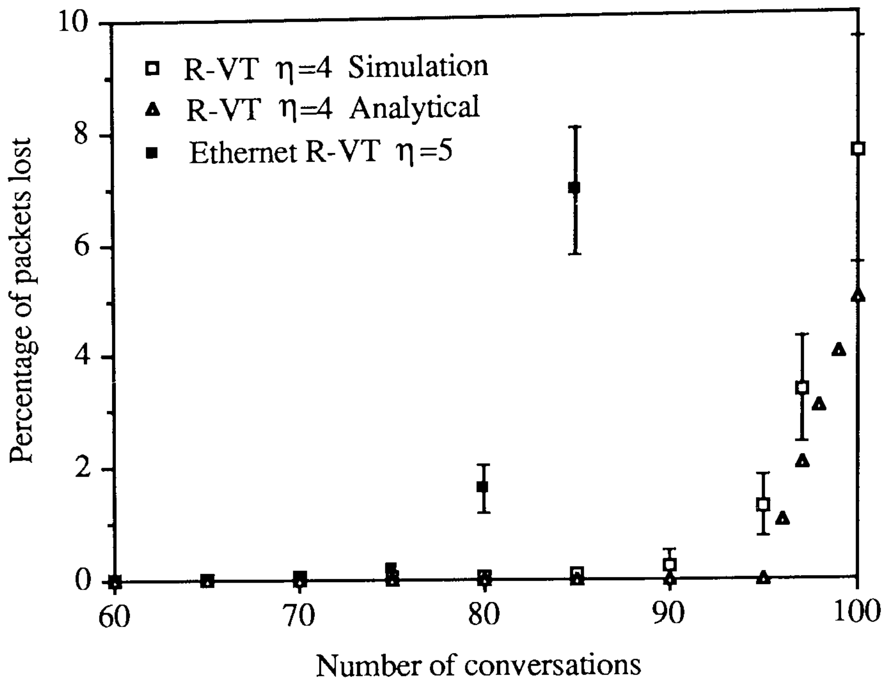


Figure 3.11 Percentage of packets lost using R-VT-CSMA/CD with a binary exponential collision backoff and established user preemption.

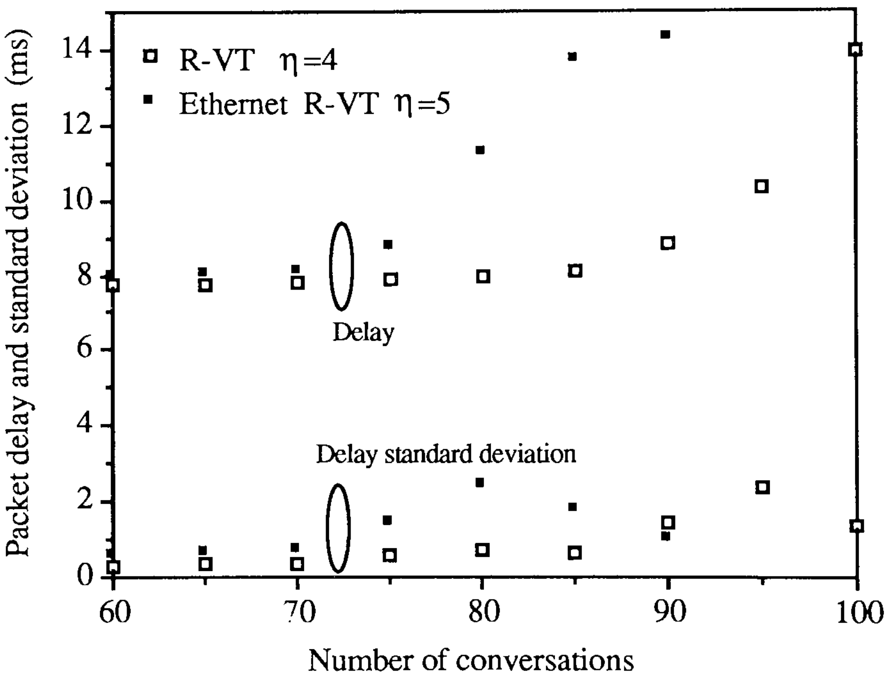


Figure 3.12 Packet delay and delay standard deviation using R-VT-CSMA/CD with binary exponential backoff and established user preemption.

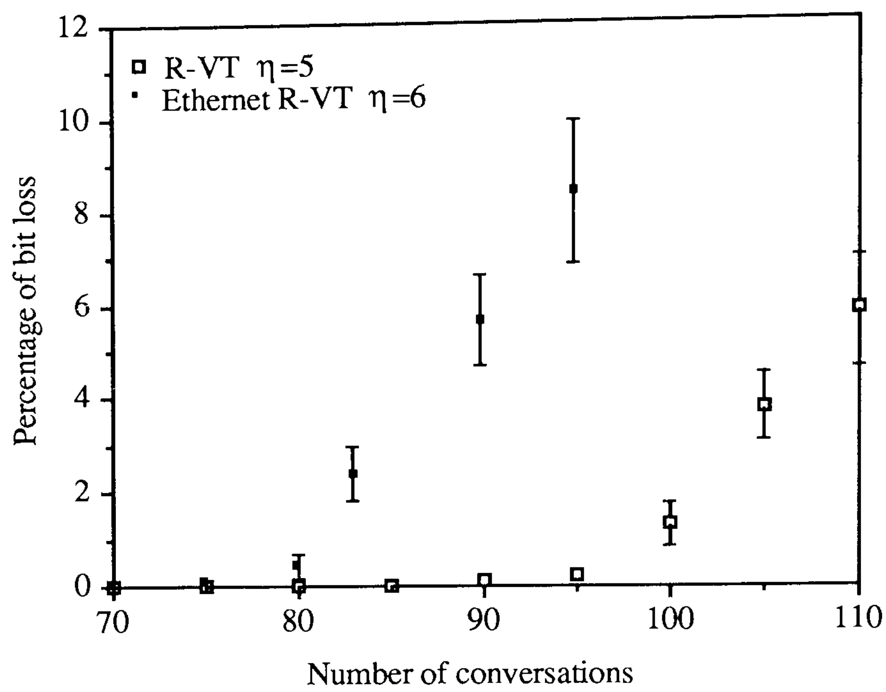


Figure 3.13 Percentage of bit loss using bit loss R-VT-CSMA/CD with binary exponential backoff and established user preemption.

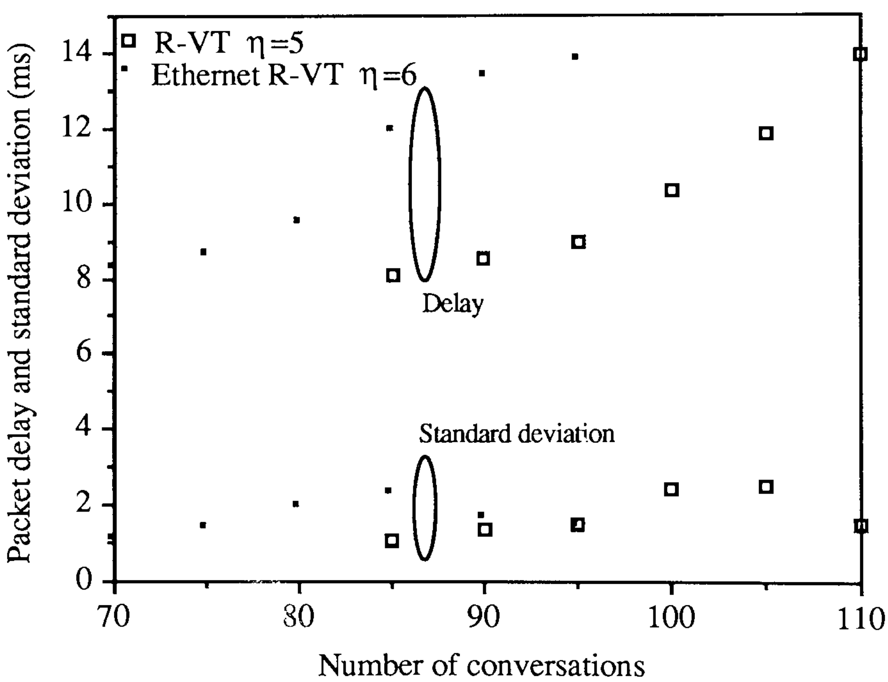


Figure 3.14 Packet delay and delay standard deviation using bit loss R-VT-CSMA/CD with binary exponential backoff with established user preemption

Figure 3.15 illustrates the bit loss R-VT-CSMA/CD protocol with a data load. The data packet delay using VT-CSMA/CD is illustrated at various data loads with a constant voice load of 40 conversations. The data load consisted of two types:

(i) Data packets with a fixed length of 1000 bits.

(ii) A bimodal data load where 90% of the packets had a length of 8000 bits and 10% of packets had a length of 800 bits.

Voice packets have priority over the data packets since the data virtual time clocks could only update when the voice virtual time clocks were up-to-date. It can be seen from the graph that the data packet delay is stable up to 31% and 33% for bimodal and uniform data loads respectively.

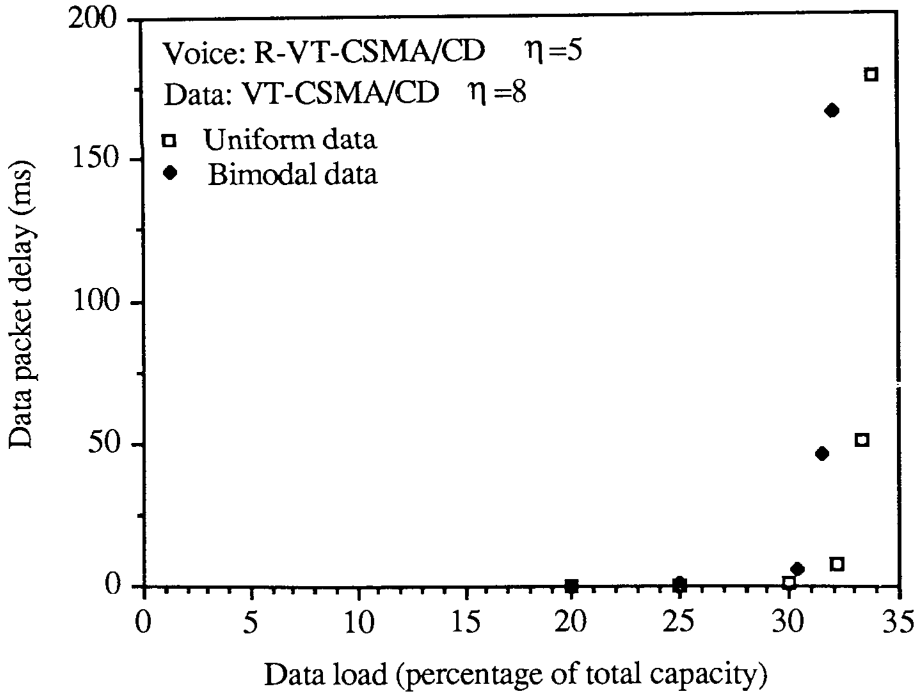


Figure 3.15 Data packet delay at various data loads with a constant voice load of 40 conversations.

3.3 R-VT-CSMA/CD With Split-Channel Reservation

It was assumed in the derivation of (3.12) that all voice stations which transmitted in the previous frame had successfully acquired a reservation slot for the current frame. This assumption depends on the number of reservation slots in a frame i.e. it is dependent upon η. For the previous results, the optimal value of η was found through simulation. Although (3.12) cannot be used to calculate the value of η then, it does however show that when the assumption is true, very little channel time is wasted due to contention. The time quantity on the LHS of (3.12) is predominantly due to the effect of packet transmissions and virtual time clock update. The average contention time per frame is small since the arrival rate of new talkspurts per station is low, and the reservation slot to transmission time ratio within a frame is high. Thus, the capacity of the network could be increased if the virtual time clock update intervals were reduced, whilst still maintaining low contention periods.

In the R-VT-CSMA/CD protocol, the common channel resource wasted between transmissions is the product of the channel capacity and the time it takes for the virtual time clocks to pass the next arrival time. The virtual time clock update could be speeded up by increasing the value of η. However, η is a function of packet arrival rate and transmission time etc.[76] and so increasing η beyond some value will bring instability into the system. This would result in an increase in the collision jamming times, since there would be less reservation slots of length $\eta\tau$ per frame. Hence the virtual time clock update period is usually dictated by the end-to-end propagation delay, τ, and cannot be reduced. However, the virtual time clock update time-bandwidth product can be reduced by using a separate narrowband channel for contention to take place while a current packet transmission is using a wideband message channel. The advantage of this split-channel reservation or Dual Channel R-VT-CSMA/CD (DC-R-VT-CSMA/CD) is that contention resolution is performed *a priori* and thus can significantly improve the channel efficiency.

The timing diagram for this protocol is illustrated in figure 3.16. Again the synchronised version is described. A packet transmission cycle on the message channel is defined as a packet transmission period followed by a virtual time clock update period. If the contention for the next packet transmission has been resolved on the signalling channel before the end of the packet transmission on the message channel, then the virtual time clock update period is zero in length. If this occurs, then the station whose packet arrival time has been passed by the virtual time clock transmits a busy-tone signal on the signalling channel to prevent any further virtual time clock update from stations. When the current packet transmission is complete, the selected station transmits its packet on the message channel and withdraws its busy-tone signal from the signalling channel so that stations can once more update their virtual time clocks on the signalling channel and repeat the process. If the virtual time update is not complete after a packet transmission, further virtual time update takes place either on the signalling or the message channel.

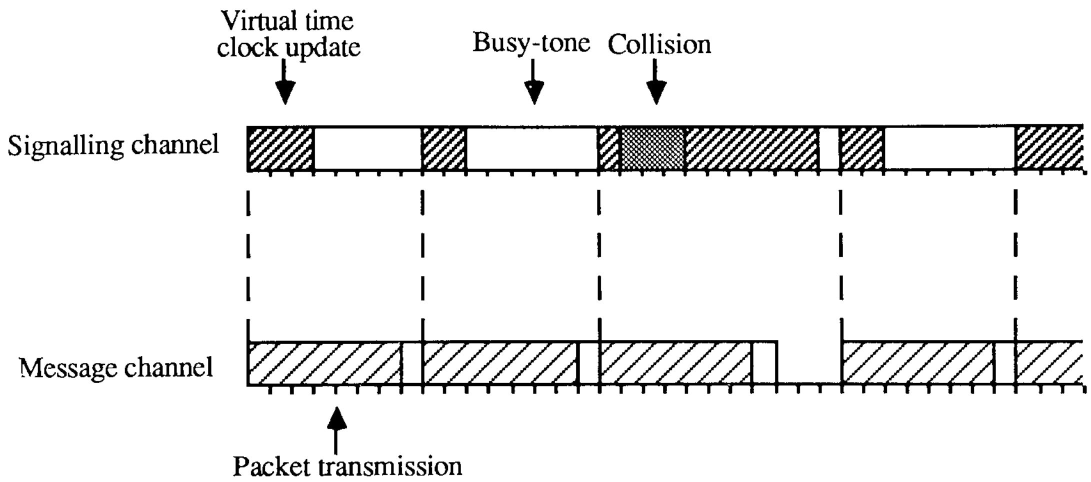


Figure 3.16 Timing diagram for R-VT-CSMA/CD with split-channel reservation

The dual channel could be implemented either by frequency-division multiplexing a single channel into two channels in a broadband scheme, or a separate line could be used for the signalling channel for a baseband system as illustrated in figure 3.17. For Ethernet-type compatibility, data stations would only have access to the message channel and voice stations would collision handshake on this channel between voice transmissions as described in section 3.2.1. Figure 3.18 illustrates the relationship between virtual time and channel activity for the DC-R-VT-CSMA/CD protocol.

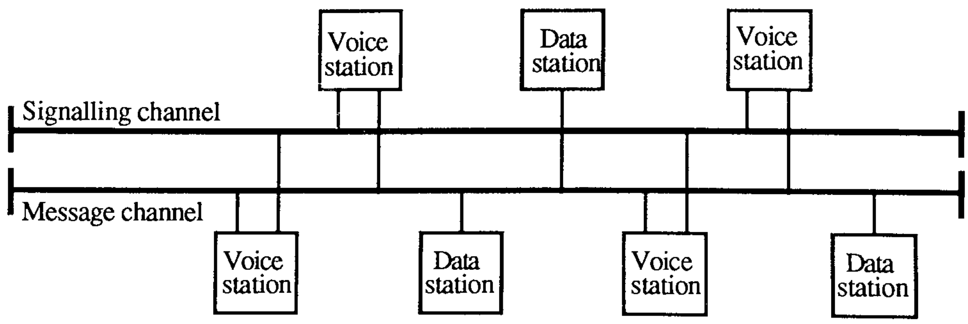
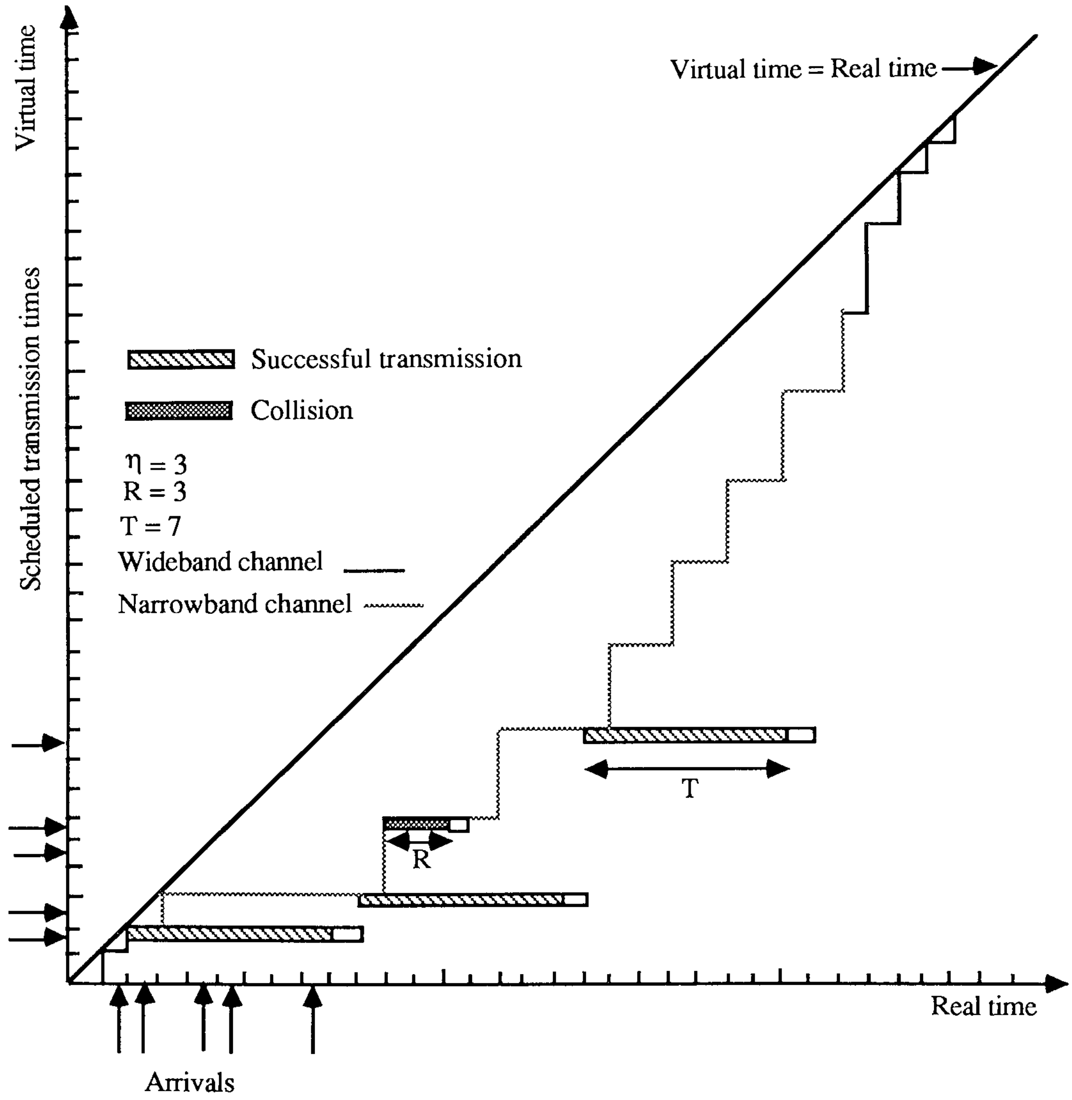


Figure 3.17 DC-R-VT-CSMA/CD system architecture.



In this synchronous example, the carrier detection time equals one minislot, this being equal to the end-to-end propagation delay.

Figure 3.18 Real and virtual time clock mappings in DC-R-VT-CSMA/CD.

A description of the DC-R-VT-CSMA/CD protocol is illustrated in figure 3.19a. Figure 3.19b expands the 'Wait until virtual time clock ≥ packet arrival time' statement of figure 3.19a. The carrier detection time is assumed to be zero on the wideband message channel and δ on the narrowband signalling channel.

```
while in talkspurt do
    begin
        while packetisation interval incomplete for current packet do
            assemble packet;
        commence packetisation interval for next packet;
        add arrival time offset;
        repeat
            begin
                wait until virtual time clock ≥ arrival time;            {fig. 3.19b}
                transmit packet;
                withdraw busy-tone signal from signalling channel;
                commence virtual time update on the signalling channel;
            end
        until successful transmission
        calculate arrival time offset;
    end
```

Figure 3.19a. Pseudo code describing the DC-R-VT-CSMA/CD protocol, where established voice stations reschedule collided packets.

```
while virtual time < arrival time do
    begin
        while signalling channel is busy do
            begin
                stop virtual time clock;
                wait τ+δ seconds;
            end
        virtual time = min{virtual time + ητ, real time};
        if virtual time < arrival time then
            wait τ+δ seconds;
        else
            begin
                transmit busy-tone on signalling channel;
                if a collision is detected then
                    begin
                        transmit jamming-tone on signalling channel;
                        compute backoff for new arrival time;
                    end
                else
                    begin
                        while message channel is busy do
                            continue to transmit busy-tone on signalling channel;
                    end
            end
        if following packet has arrived then
            discard original packet;
    end
```

Figure 3.19b *Wait until virtual time clock ≥ arrival time* pseudo code using the DC-R-VT-CSMA/CD protocol.

3.3.1 Throughput Of VT-CSMA/CD With Split-Channel Reservation

In this section the analytical throughputs of data packets are compared when using VT-CSMA/CD and DC-VT-CSMA/CD. The VT-CSMA algorithm has two modes of operation. It is backlogged when the virtual time clock runs at rate η, and caught up when it runs at the real time clock rate. Hence to derive the throughput it is necessary to find the conditional throughput equations for each mode separately and then average them in proportion to the time spent in each mode. To determine the throughput of DC-VT-CSMA/CD, the following notation is used:

W: Total available bandwidth.

φ: Fraction of W assigned to the message channel.

b: Number of bits per packet.

T: Transmission time of a packet in VT-CSMA
= b/W seconds.

T': Transmission time of a packet in DC-VT-CSMA
= $b/(\varphi W)$ seconds.

G: Packet arrival rate per T seconds.

τ: End-to-end propagation delay.

δ: Carrier detection time on the narrowband signalling channel. The modulation rate is taken as 1 bit/Hz.
$\delta = 1/(2(1-\varphi)W)$ seconds.

P_b: Equilibrium probability that the virtual time clock is backlogged.

P_c: Equilibrium probability that the virtual time clock has caught up with the real time clock.

The throughput assumptions are as follows:

- The number of users is infinite and the arrival process from this infinite population is Poisson distributed with arrival rate G per packet transmission time (T); the process includes both new and previously collided packets.
- Carrier sensing on the wideband message channel takes negligible time.
- The time axis is slotted, where the slot size is equivalent to twice the end-to-end delay (plus the carrier detection time with the dual channel protocol).

- The channel is noiseless so that failure of transmission is due to collisions only.

- The overlap of any fraction of two packets results in destructive interference so that both packets must be retransmitted. The ideal case is considered, where the collision detection and abort operations take only one slot time. This also applies to the busy-tones on the signalling channel in the dual channel protocol.

A transmission period followed by an idle period constitutes a cycle. With the infinite population assumption, all cycles are statistically identical [55]. From the renewal theory, the throughput for the VT-CSMA/CD protocol can be expressed as:

$$S_{sc} = \frac{T}{(T+2\tau) + 2\tau E[I]} \qquad \ldots(3.13)$$

where E[I] is the expected length of the idle period in slots, following a transmission.

$$P\{I=0\} = P\{\text{One arrival in a slot}\}$$
$$= arGe^{-arG} = p$$

where $r \in \{1,\eta\}$ and a is the ratio of the slot size to the transmission time, $=2\tau/T$. A continuation of this argument gives

$$P\{I=i\} = (1-p)^i p$$

and with a mean value given by

$$E[I] = \frac{1-p}{p} = \frac{1 - arGe^{-arG}}{arGe^{-arG}} \qquad \ldots(3.14)$$

Using equations (3.13) and (3.14), the throughput of the single channel VT-CSMA/CD, S_{SC}, can be expressed as:

$$S_{sc} = \frac{rG\, e^{-arG}}{1 + rG\, e^{-arG}} \qquad \ldots(3.15)$$

Because of the Poisson traffic assumption, the number of transmissions in each slot is Poisson distributed. The mean number in the distribution must be either aG or $a\eta G$, with equilibrium probabilities P_c and P_b respectively. If η is large enough for the backlog to remain finite with probability one, then the throughput may be determined by equating the average duration of a cycle with the average advance of the virtual time clock per cycle. Using $P_b=1-P_c$, then from [84]

$$P_c = \frac{\min\{0, E[L|\eta G/T] - \eta\tau\}}{E[L|\eta G/T] - \eta\tau - E[L|G/T] + \tau} \qquad ...(3.16)$$

where E[L] is the expected duration of a cycle. For the DC-VT-CSMA/CD protocol, the throughput can be derived in similar manner. The throughput is given by:

$$S_{DC} = \frac{T'}{(T' + 2\tau + \delta) + (2\tau + \delta)E[I]} \qquad ...(3.17)$$

where E[I] is the expected duration of the idle period in slots.

$$P\{I=i\} = (1-p)^{T'/(2\tau+\delta)} (1-p)^i p$$

$$\text{where } p = a'rG e^{-a'rG}$$

a' is the ratio of the slot time to the packet transmission time, $= (2\tau+\delta)/T'$.

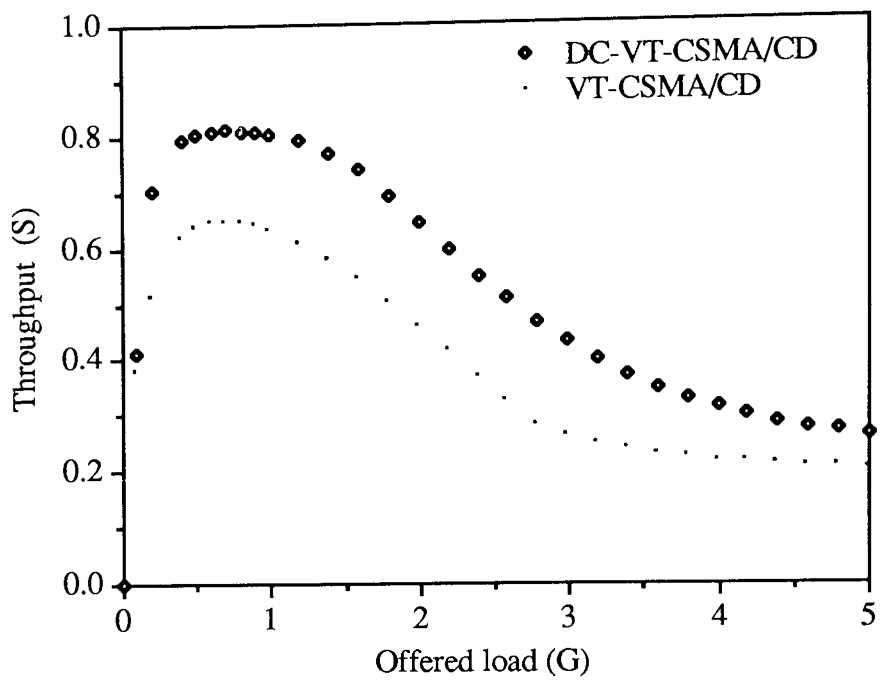
The mean value of the idle period is thus,

$$E[I] = \frac{\left(1 - a'rG e^{-a'rG}\right)^{\frac{1+a'}{a'}}}{a'rG e^{-a'rG}} \qquad ...(3.18)$$

Combining equations (3.17) and (3.18), the throughput for DC-VT-CSMA/CD is given by:

$$S_{DC} = \frac{rG e^{-a'rG}}{(1+a')\, rG e^{-a'rG} + \left(1 - a'rG e^{-a'rG}\right)^{\left(\frac{1+a'}{a'}\right)}} \qquad ...(3.19)$$

The DC-VT-CSMA/CD throughput can be evaluated as before by equating the average duration of a cycle from (3.19) with the average advance of the virtual time clock per cycle. Data throughput for the VT-CSMA/CD and DC-VT-CSMA/CD protocols is illustrated in figure 3.20. This figure demonstrates the advantage of using a small fraction of the available bandwidth for contending the next packet transmission in parallel with the current packet transmission.



Total channel capacity =10Mbps, signalling channel =100Kbps, end-to-end propagation delay = 10μs, 1000 bit packets and η=7.

Figure 3.20 Throughput comparison of VT-CSMA/CD and DC-VT-CSMA/CD.

3.3.2 Simulation Results For The DC-R-VT-CSMA/CD Protocol

From the 10Mbps of available capacity, a signalling channel of 100Kbps was used. Assuming the modulation rate was 1bit/Hz, then the carrier detection time, δ, on this narrowband channel is 5μs, using the equation $\delta = 1/2(1-\varphi)W$. The number of bits which can be transmitted on the wideband channel per slot is $W\varphi\tau$ .With a 10μs end-to-end delay, 99 bits can thus be transmitted in each slot. Using a frame time of 7.5ms and 120 bit header, the number of bits per packet is 600. To utilise the synchronous channel efficiently, an integer number of slots should be transmitted. It is required that the total LAN delay does not exceed 15ms and so the frame time has now been reduced so that an integer 6 slots may be used. The DC-R-VT-CSMA/CD protocol hence used a frame time of 7.375ms. For the Ethernet-type network, it was assumed that a separate cable would be used for the signalling channel and so the frame time here is still 7.5ms.

Figures 3.21 and 3.22 illustrate the synchronous dual channel protocol with exponential backoff and exponential backoff with established station preemption respectively. Again the optimum value of η was found through simulation for a 2% voice loss. As expected, more conversations can be supported with established station preemption. Also shown is the perfect scheduler M/D/1 queue. This assumes that the total voice packet arrival times are exponentially distributed, the service (transmission) time is constant and there is only one server. This is the scenario where the propagation time is zero, i.e. the system is no longer distributed. The total time a packet accesses the network is equal to its waiting time and its transmission time. For an M/D/1 queue this is given by [56]:

$$D = \frac{\rho T}{2(1-\rho)} + T \qquad \text{...(3.20)}$$

where T is the packet transmission time and $\rho=\lambda T$ where λ is the mean packet arrival rate. Limiting the access delay, D, to one frame time, the maximum number of possible conversations that can be supported without voice loss can be calculated using (3.20) with (2.7) and (2.8).

The performance of the Ethernet-type protocol is inferior due to the wasted capacity in the collision handshaking procedure. Figures 3.23 and 3.24 illustrate the packet delays for the two protocol versions. As the total delay of successful packets tends towards two frame times, the standard deviation of the delay tends to zero.

Figures 3.21 to 3.24 illustrates the case where whole packets were lost if the access time exceeded one frame time. Figures 3.25 and 3.26 show the voice loss and packet delay when only the bits whose access time exceeds one frame time are discarded. With bit loss, 4 more conversations (8 voice stations) can be supported. The clip statistics are superior since the minimum clip is no longer one frame which is probably more desirable to a listener. These are illustrated in table 3.2.

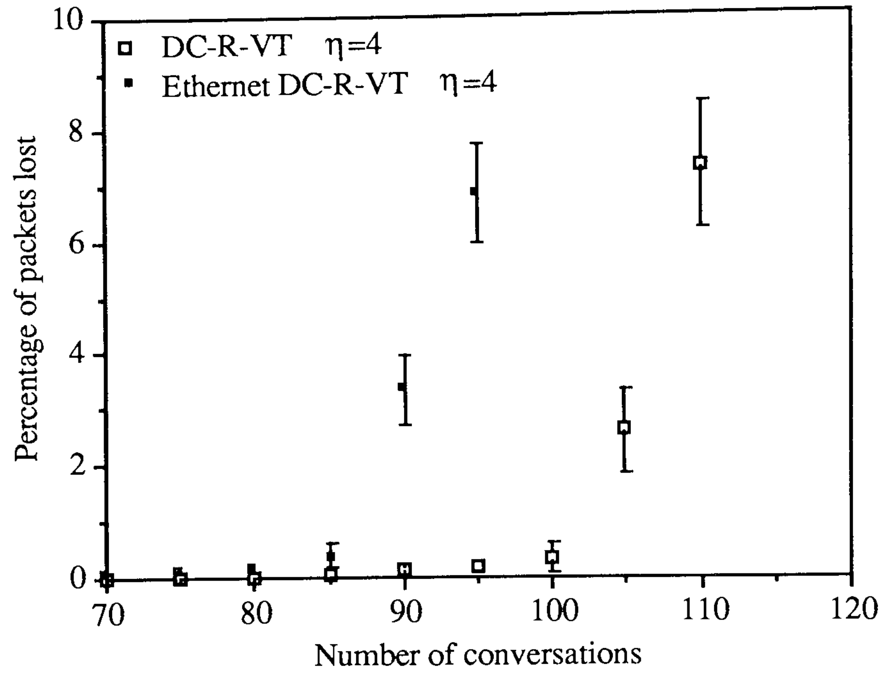


Figure 3.21 Percentage of voice packets lost using DC-R-VT-CSMA/CD with a binary exponential collision backoff.

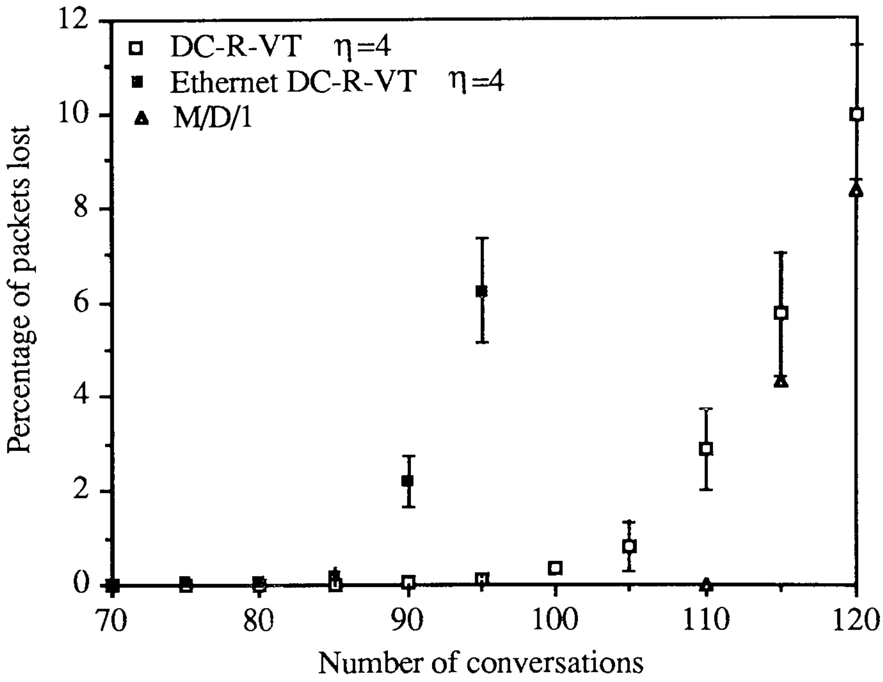


Figure 3.22 Percentage of voice packets lost using DC-R-VT-CSMA/CD with a binary exponential collision backoff and established station preemption.

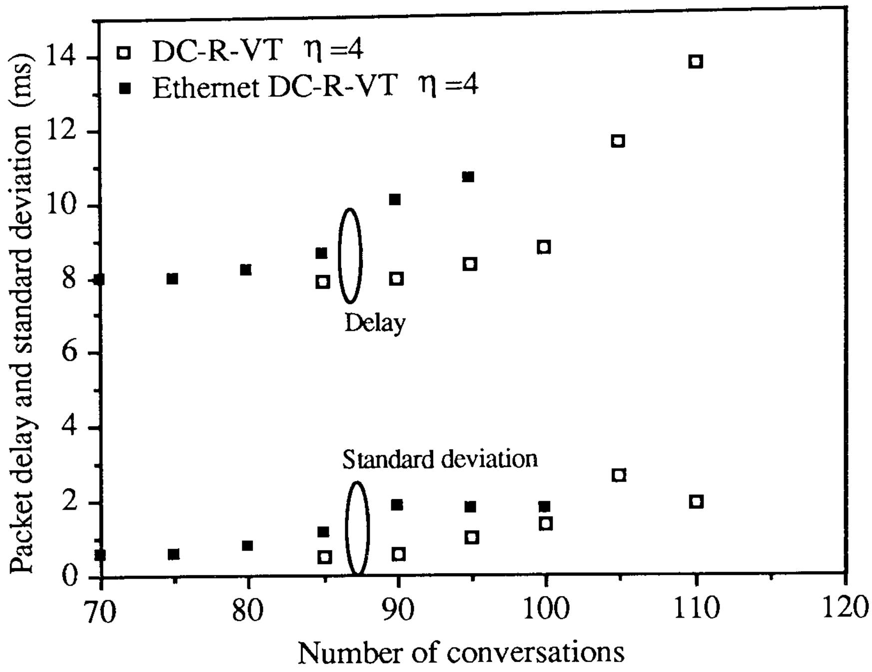


Figure 3.23 Voice packet delay and delay standard deviation using DC-R-VT-CSMA/CD with binary exponential collision backoff.

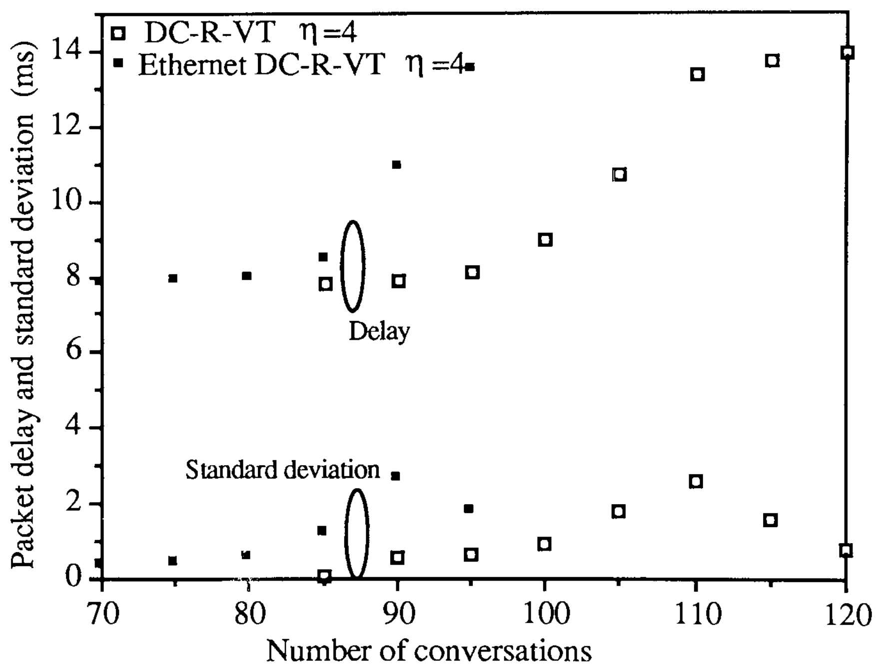


Figure 3.24 Voice packet delay and delay standard deviation using DC-R-VT-CSMA/CD with binary exponential collision backoff and established station preemption.

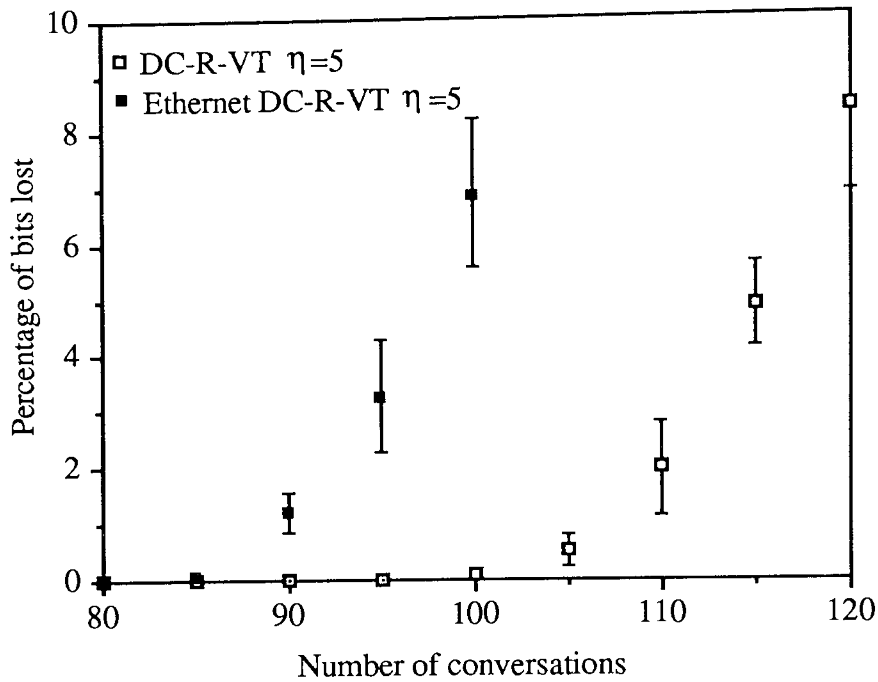


Figure 3.25 Percentage of voice bits lost using DC-RVT-CSMA/CD with a binary exponential backoff and established station preemption.

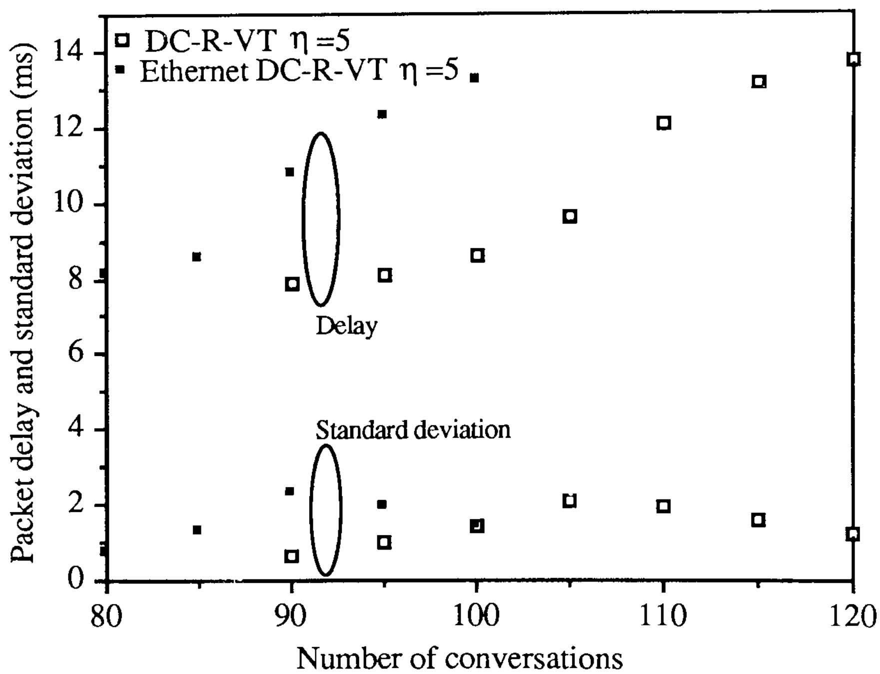


Figure 3.26 Packet delay and delay standard deviation using DC-R-VT-CSMA/CD with binary exponential backoff and established station preemption.

	N_{max}	Clip Length (ms) Mean	Std dev	Max
Packet loss	106	22.5	24.3	307
Bit loss	110	1.9	2.9	134

Table 3.2 Clip statistics for DC-R-VT-CSMA/CD with 2% packet and bit loss.

When data traffic is applied, data packets can only be transmitted once the voice virtual time clocks are up-to-date.The dual channel protocol updates the voice virtual time clock to contend for the next transmission in parallel with the current transmission. This should provide more channel transmission time for data packets than the R-VT-CSMA/CD protocol. Figure 3.27 illustrates the amount of time the voice virtual time clock is up-to-date with the real time clock for both protocols. It is seen that the dual channel protocol would provide data traffic with significantly more channel transmission time.

Figure 3.28 shows data packet delay using VT-CSMA/CD at various data loads with a constant voice load. Approximately 42% uniform data load could be supported before the data packet delay becomes unstable. For the same input loads, figure 3.15 illustrated that the R-VT-CSMA/CD protocol could support only 33%.

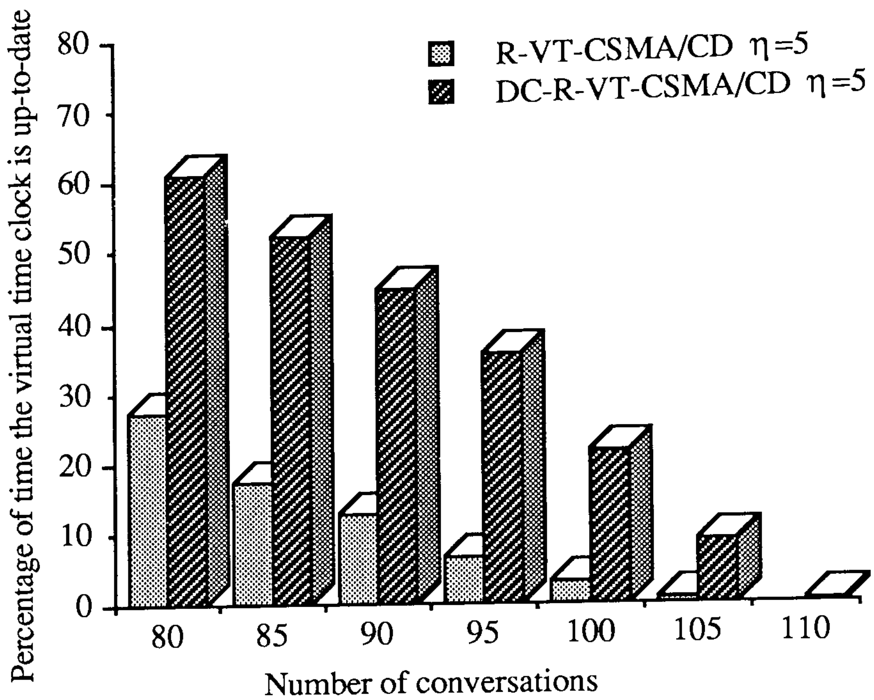


Figure 3.27 A comparison between R-VT-CSMA/CD and DC-R-VT-CSMA/CD of the amount of time the virtual time clock is up-to-date with the real time clock.

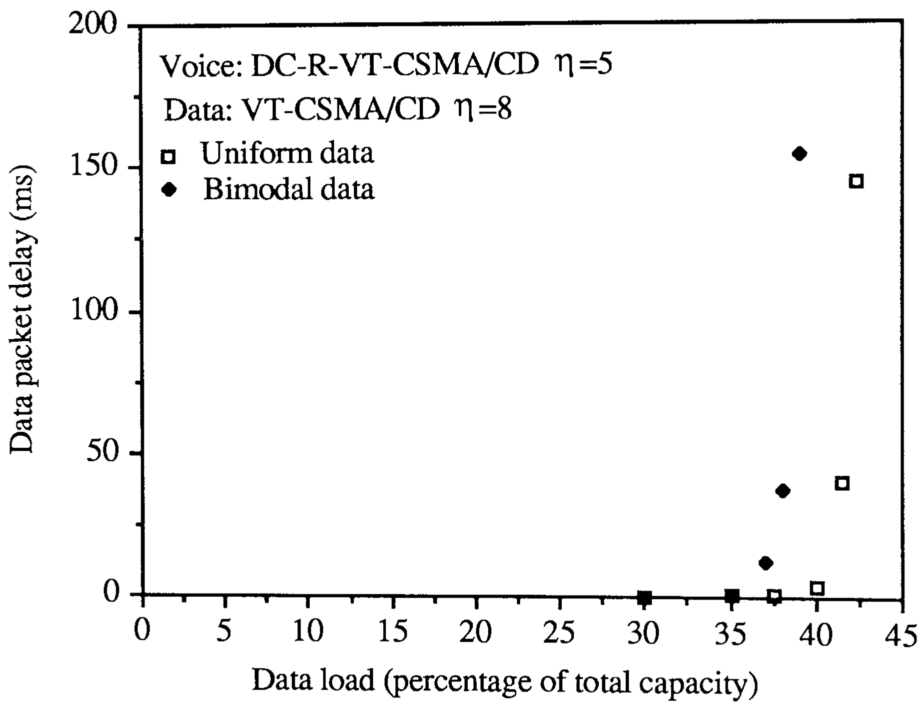


Figure 3.28 Data packet delay at various data loads and for a constant voice load of 40 conversations.

3.4 Summary

In this chapter, the R-VT-CSMA/CD protocol was examined. An improvement in performance over the original protocol was obtained through a different collision resolution mechanism, and through the use of voice bit loss rather then voice packet loss. These improvements supported a greater number of conversations and gave more desirable voice clip statistics than the original protocol.

Based on the analysis of the number of supportable conversations for the protocol, a new protocol called DC-R-VT-CSMA/CD was developed which was found to have a superior performance. It was shown in section 3.2.2 with equation (3.12) that at the optimum value of η for the R-VT-CSMA/CD protocol, the majority of wasted channel time was due to the virtual time clock update and not contention time for new stations to find a reservation slot. To reduce the wasted product of channel bandwidth and time to update the virtual time clocks, the channel was split in two; a narrowband channel for contention and a wideband channel for transmissions. The advantage of this dual channel architecture is that the virtual time clock update is performed in parallel with a packet transmission. It can be seen from figure 3.29 that this dual channel protocol can support about 9 conversations (18 voice stations) more than that of the R-VT-CSMA/CD protocol, assuming 2% bit loss is acceptable.

Also shown is the nonpersistent CSMA/CD protocol with a frame size of 7.5ms and fixed length packets with bit loss. It can be seen that both the R-VT and DC-R-VT-CSMA/CD protocols can support significantly more conversations on the 10Mbps LAN than that of the CSMA/CD protocol.

The successful packet delays of the three protocols are illustrated in figure 3.30. All the protocols limit their packet delays by two frame times, otherwise the packets are lost. The delay consists of one frame time for the packetisation interval and a maximum of one frame time for channel access. As the load increases, the delays of both the R-VT and DC-R-VT-CSMA/CD protocols approach the two frame time limit. The variance in the delay of successful packets thus tends towards zero. However, with the CSMA/CD protocol, as the load increases, the number of collisions also increases. For virtual time protocols, bit loss is predominantly due the virtual time clock lag exceeding one frame time, but bit loss in CSMA/CD is due to collided packets being rescheduled to a future time where their access times are greater than one frame time. It is for this reason that the mean CSMA/CD packet delay is still relatively low when bit loss starts to become significant.

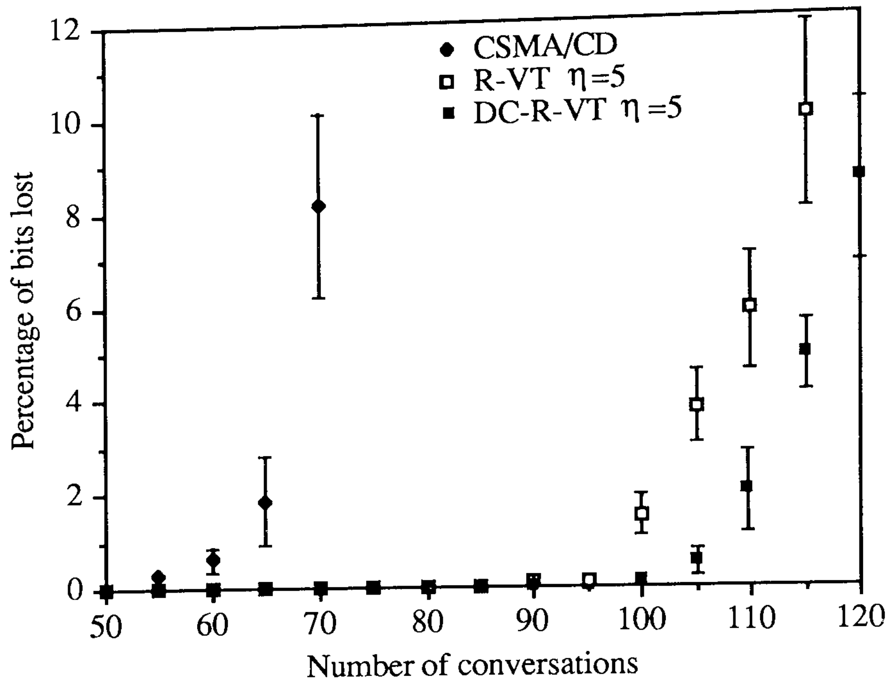


Figure 3.29 A comparison of the percentage of bits lost for the CSMA/CD, R-VT-CSMA/CD and DC-R-VT-CSMA/CD protocols.

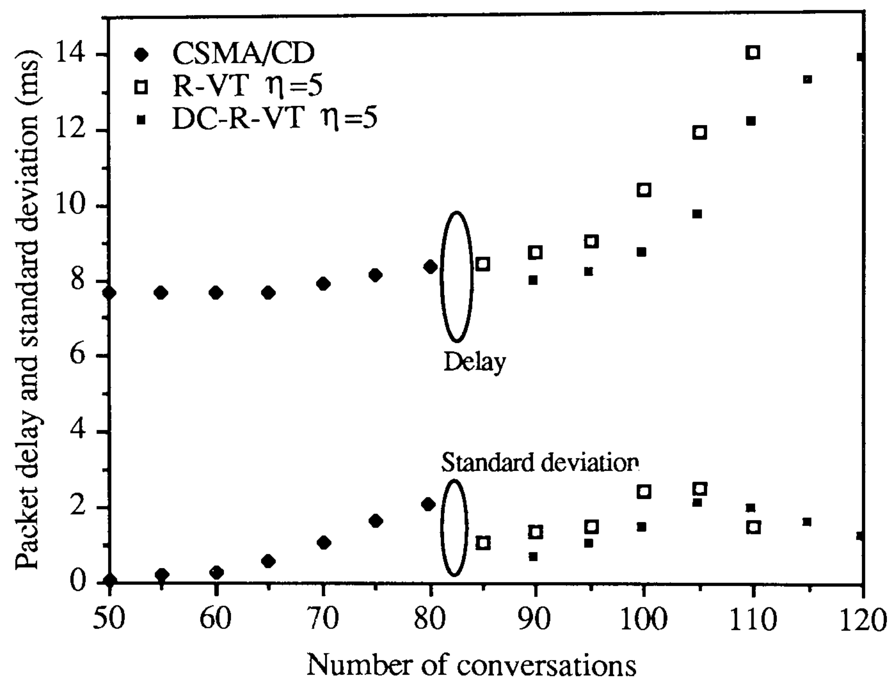


Figure 3.30 A comparison of the packet delay and standard deviation for the CSMA/CD, R-VT-CSMA/CD and DC-R-VT-CSMA/CD protocols.

This fact is illustrated in the clip statistics for the CSMA/CD protocol. The mean clip and its standard deviation are significantly higher than that of the virtual time protocols due to increased number of random reschedules at high load. The bit loss clip statistics are illustrated in table 3.3.

	N_{max}	Clip Length (ms) Mean	Std dev	Max
CSMA/CD	66	27.1	36.8	213
R-VT-CSMA/CD	101	3.9	12.1	181
DC-R-VT-CSMA/CD	110	1.9	2.9	134

Table 3.3 Clip statistics for CSMA/CD, R-VT-CSMA/CD and DC-R-VT-CSMA/CD with 2% bit loss.

A frame time of 7.5ms was chosen so that for inter-LAN communications with a maximum allowable delay of 300ms, 270ms can be allocated to the network linking the LANs. However, if such a link is not used i.e. intra-LAN voice communication, larger frame times could be used. The advantage of this is a reduction in per-packet overhead in terms of header, propagation time and interframe spacing etc. The improved performance with this reduced overhead is illustrated in figure 3.31 for the virtual time protocols.

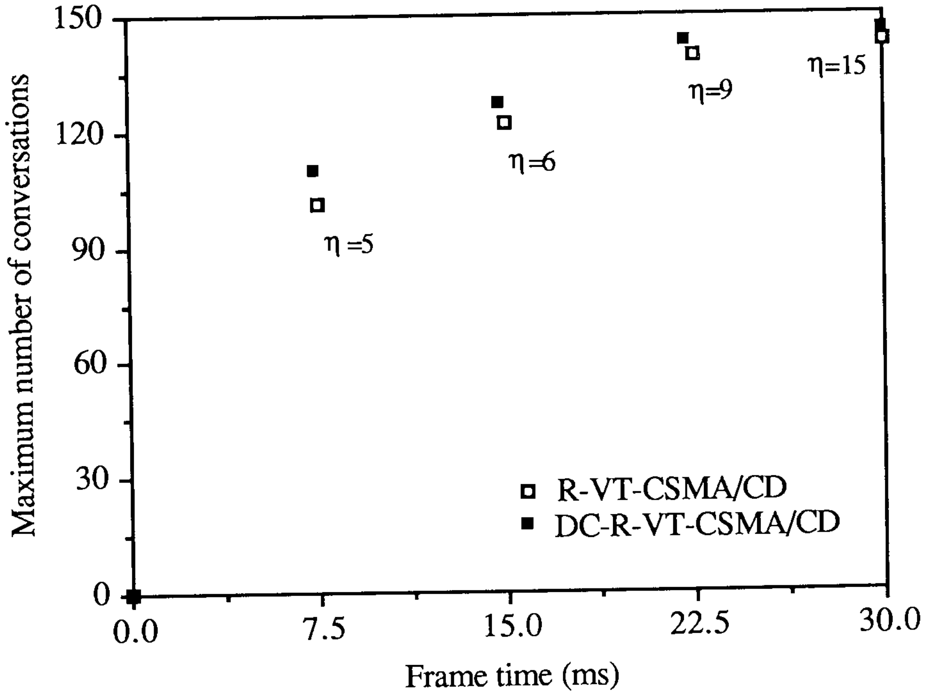


Figure 3.31 Number of conversations for 2% bit loss using R-VT-CSMA/CD and DC-R-VT-CSMA/CD at various frame times.

CHAPTER 4

VARIABLE LENGTH PACKET VOICE MAC PROTOCOLS FOR LANS

4.1 Variable Length Packets

In this chapter, two protocols using variable length packets are developed. The first protocol, called Adaptive R-VT-CSMA, again uses the implicit virtual time reservation mechanism of chapter 3, but uses short packets at low network load to reduce delay and longer packets at higher network loads to increase efficiency. The use of variable length packets is shown to effect the position of reservation slots, and hence the virtual time update speed.

The second protocol arranges active voice stations into a distributed global queue using explicit reservations to determine the transmission order. If the transmission times of the stations in the queue exceed the allocated total transmission time within a frame, the packets are either shortened, or scheduled to the front of the queue in the next frame in order that voice loss is distributed fairly among stations.

4.2 The Adaptive R-VT-CSMA/CD Protocol

In the R-VT-CSMA/CD protocol described previously, η determined the input traffic rate *clocked out* onto the channel. If the time window $\eta\tau$ is too small, packet loss occurs because of active voice stations having to wait too long to transmit their packets. If η is too large at high load, packet loss occurs because of collisions, since the time window $\eta\tau$ is encompassing too many packet arrivals. 1-persistent CSMA behaviour can be obtained by choosing $\eta=\infty$. With the R-VT-CSMA/CD and dual channel protocols, the minimum separation on the virtual time axis for successful transmissions between established stations is $\eta\tau$. $\eta\tau$ is typically several times smaller than a packet transmission time, so the ratio of effective reservation slots to maximum number of users in a frame is large. Thus it takes several slots after a transmission for the virtual time clock to catch up with the real time clock. Voice and data throughput could be increased if this voice virtual time clock update period was reduced.

Larger voice virtual time clock speeds could be used if established stations are separated on the virtual time axis by more than $\eta\tau$. This separation on the virtual time axis can be increased by using variable length packets [34]. Consider if each voice station has a buffer of length P_{max} in which the voice samples are placed. Assume that when the length of the packet in the buffer reaches a minimum P_{min}, the packet is ready for transmission. The real time at which this occurs is stored in the packet arrival time buffer. The packet length continues to grow inside the buffer. When the voice virtual time clock passes the value in the arrival time buffer, the whole contents of the buffer is transmitted. If, due to traffic intensity, the packet length reaches P_{max} before it is transmitted, the oldest sample is discarded when a new one is generated.

While the packet is being transmitted at the channel transmission speed C bits/s, it continues to grow at a rate of V bits/s, where V is the voice encoding rate. The length of the shortest packet P_s is given by

$$P_s = P_{min} + (P_s / C).V$$

thus,
$$P_s = \frac{P_{min}}{1 - V/C} \qquad \text{...(4.1)}$$

With the variable length packets described, assume an existing station transmits its packet at a real time R (this occurs when the virtual time clock passes its arrival time - the value of the real time clock at which the packet buffer was of length P_{min}). The station`s next packet then will be ready for transmission at the real time $R+T+P_{min}/V$, where T is the transmission time of its current packet which is equal to P/C, where P is the length of the current packet.

The real time, R+T, at which the transmission finished is unique to that particular station. Thus the arrival time of the station`s next packet at the real time $R+T+P_{min}/V$ will be separated on the virtual time axis from all other existing users by at least the amount T, which is in the range $[P_s/C + \vartheta/C, P_{max}/C + \vartheta/C]$, where ϑ is the frame overhead. Hence, as long as $\eta < T_s/\tau$, where T_s is the packet transmission time of the shortest packet including overhead, established users will never collide. In R-VT-CSMA/CD however, $\eta\tau$ is typically several times smaller than T_s. It can be seen then, that this variable length packet protocol has the potential to reduce the virtual time clock update duration compared to the previous protocols of chapter 3, through the use of larger virtual time clock speeds, η.

If an existing user collides with new users, it continues to transmit its collision jamming signal since it knows only new users are involved. When all new users have backed off, the existing user then transmits its packet.

A correlation factor CF can be defined as the expected number of talkspurts in the current frame which still persist in the next frame at equilibrium, divided by the expected number of talkspurts in the current frame at equilibrium [123]. Under equilibrium conditions, the expected number of talkspurts which will not persist is the sum of the expected new arrivals and the expected departures of talkspurts in a frame. Thus

$$CF = \frac{\left(\frac{\mu_t}{\lambda}\right) - 2\mu_t F}{\left(\frac{\mu_t}{\lambda}\right)} \qquad \ldots(4.2)$$

where $1/\mu_t$ is the mean total talkspurt arrival rate. Since the mean talkspurt length $(1/\lambda)$ is generally much greater than twice the frame time (2F), it is seen that the CF is close to 1.

Hence, in order to reduce the virtual time update period, larger virtual time clock speeds, η, could be used with variable length packets, since collisions between existing users never occur if $\eta < T_s/\tau$. At high load, packet lengths will be greater than P_s, and so packet separation times on the virtual time axis will be greater than T_s. Hence, due to the high correlation factor, the ratio of the number of effective reservation slots to new stations in talkspurt should still be adequate, if $\eta<T_s/\tau$.

This Adaptive R-VT-CSMA/CD (AR-VT-CSMA/CD) protocol achieves a unique reservation time on the virtual time axis for existing users, which depends upon traffic intensity.The channel activity using the AR-VT-CSMA/CD protocol is illustrated in figure 4.1. Initially station 1 packet 1 (S1 P1) successfully transmits when the virtual time clock passes the real time at which the station had assembled $T_{min}.V$ bits, where T_{min} is the minimum packetisation interval, P_{min}/V. After the transmission is complete and all the bits have been transmitted, S1 P2 is ready T_{min} seconds later. However S1 P2 collides with S2 P1. S1 P2 continues to transmit immediately after the jamming signal since it is an established station, while S2 P1 randomly reschedules since it is new. It reschedules to a virtual time which is less than $\eta\tau$ from the S1 P2 scheduled time (i.e. could cause a collision in the R-VT-CSMA/CD protocol in subsequent frames). However, with AR-VT-CSMA/CD this does not occur. It can be seen that the scheduled transmission times for S1 P3 and S2 P2 are greater than the minimum transmission time,T_s. The AR-VT-CSMA/CD protocol is described in figures 4.2a and 4.2b.

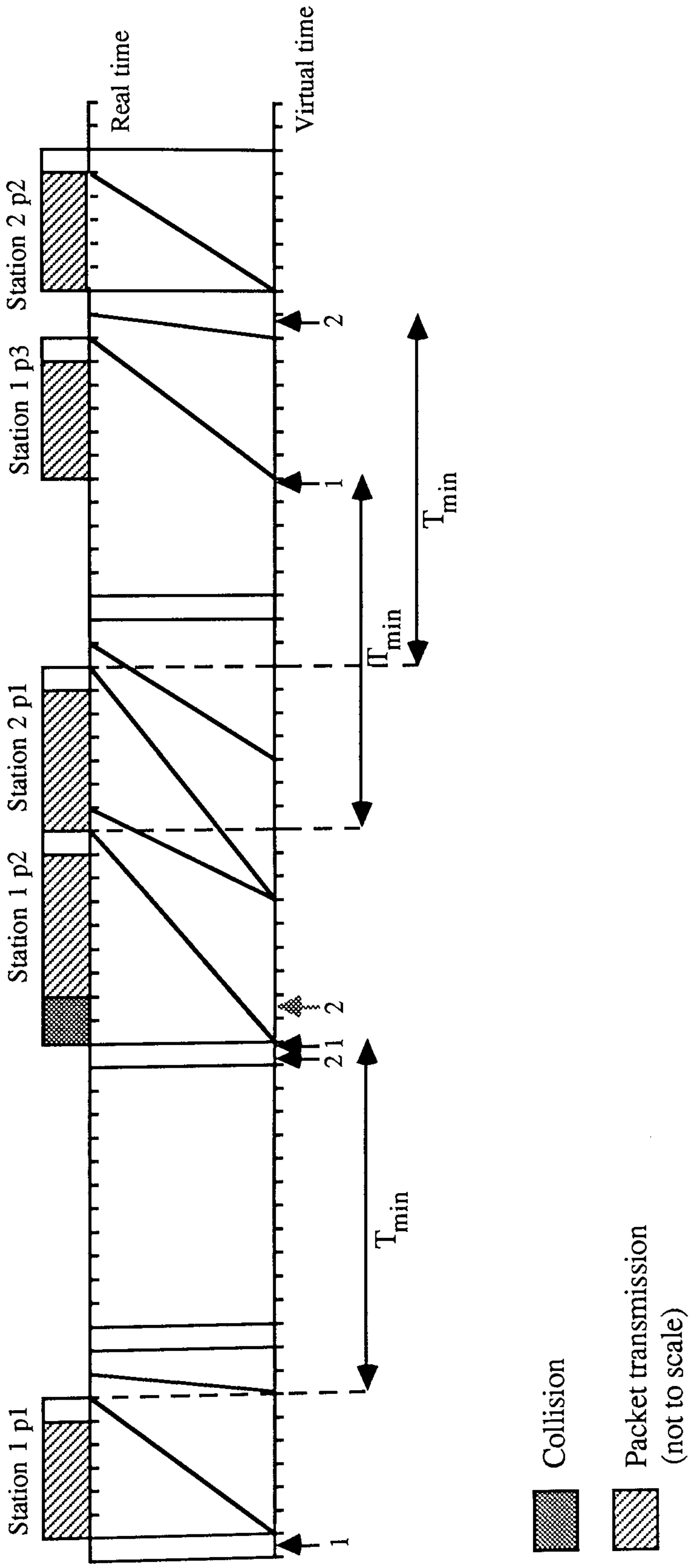


Figure 4.1 Channel activity using the AR-VT-CSMA/CD protocol.

```
while in talkspurt do
  begin
    while packetisation interval < T_min do
      assemble packet;
      repeat
        begin
          wait until virtual time clock ≥ arrival time;                {fig. 4.2b}
          transmit packet (all voice bits up to P_max);
          if collision then
            begin
              if established station then
                transmit jamming signal;
                retransmit packet (all voice bits up to P_max);
                complete successful transmission;
              else
                transmit jamming signal;
                compute backoff;
            end
          else                                           {successful transmission}
            complete successful transmission;
        end
      until successful transmission
  end
```

Figure 4.2a. Pseudo code describing the AR-VT-CSMA/CD protocol

```
while virtual time < arrival time do
  begin
    while channel is busy do
      begin
        stop virtual time clock;
        wait τ seconds;
      end
    virtual time = min{virtual time+ητ, real time};
    if virtual time < arrival time then
      wait τ seconds;
    if voice buffer contents exceeds P_max then
      discard excess voice bits;
  end                                          {packet ready for transmission}
```

Figure 4.2b. *Wait until virtual time clock≥arrival time* pseudo code.

Thus the AR-VT-CSMA/CD protocol achieves a unique reservation time on the virtual time axis for existing users, which depends upon traffic intensity. It can improve throughput by using a larger virtual time clock speed than R-VT-CSMA/CD, since the established user reservation times are now separated by a packet transmission time.

4.2.1 Simulation Results Of The AR-VT-CSMA/CD Protocol

The synchronous AR-VT-CSMA/CD protocol was simulated on an error-free LAN. The LAN had a capacity of 10Mbit/s with an end-to-end delay of 10μs. The speech was encoded at 64Kbit/s. Further network and simulation details can be found in chapter 6.

Again there is the requirement that the packetisation delay plus access delay must not exceed 15ms. This gives the constraint that T_{max}=15ms. Figures 4.3 and 4.4 illustrate the percentage of voice bits lost when using T_{min}=7.5ms and T_{min}=10ms respectively. Various values of η have been used, all of which satisfy the requirement that $\eta<T_s/\tau$, where T_s is the minimum packet transmission time. As the value of T_{min} increases, the amount of overhead per packet decreases and so the channel can support a greater number of conversations.

Figures 4.5 and 4.6 show the case where T_{min}=12.5ms and T_{min}=14.9ms respectively. At the various values of η, T_{min}=12.5ms can support more conversations than T_{min}=7.5ms or 10ms. However, when T_{min}=14.9ms, although the per-packet overhead is small, the number of supportable conversations decreases, since if a voice station does not transmit its packet within 0.1ms of it being ready, bits will be lost. To achieve little voice loss, new stations must be able to find a reservation slot quickly and also the virtual time clock lag must be small. Table 4.1 illustrates the number of supportable conversations (assuming 2% bit loss is acceptable) for the various values of T_{min}.

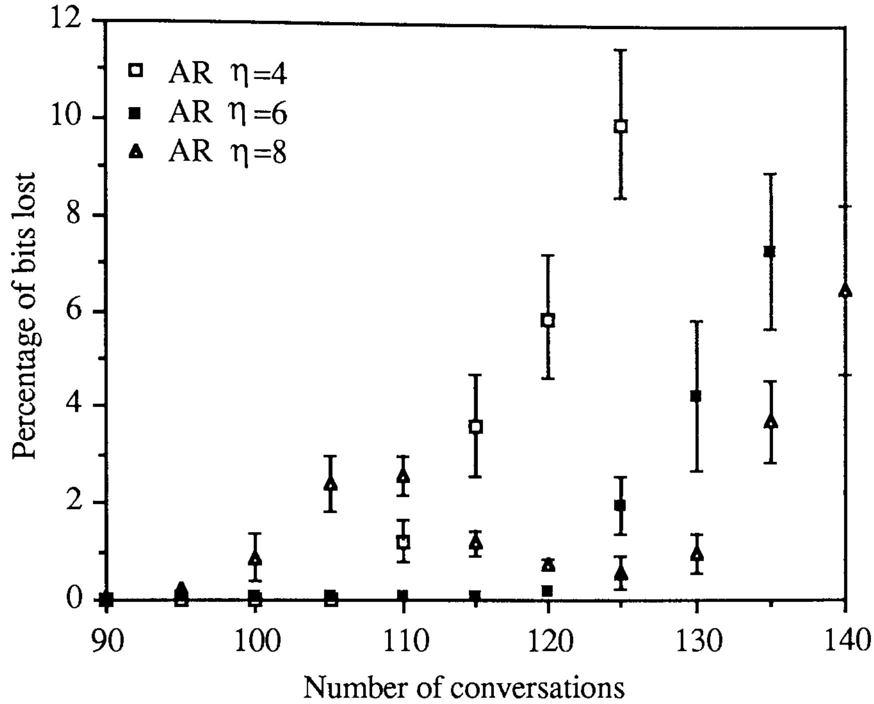


Figure 4.3 Percentage of bits lost using AR-VT-CSMA/CD with Tmin=7.5ms and Tmax=15ms.

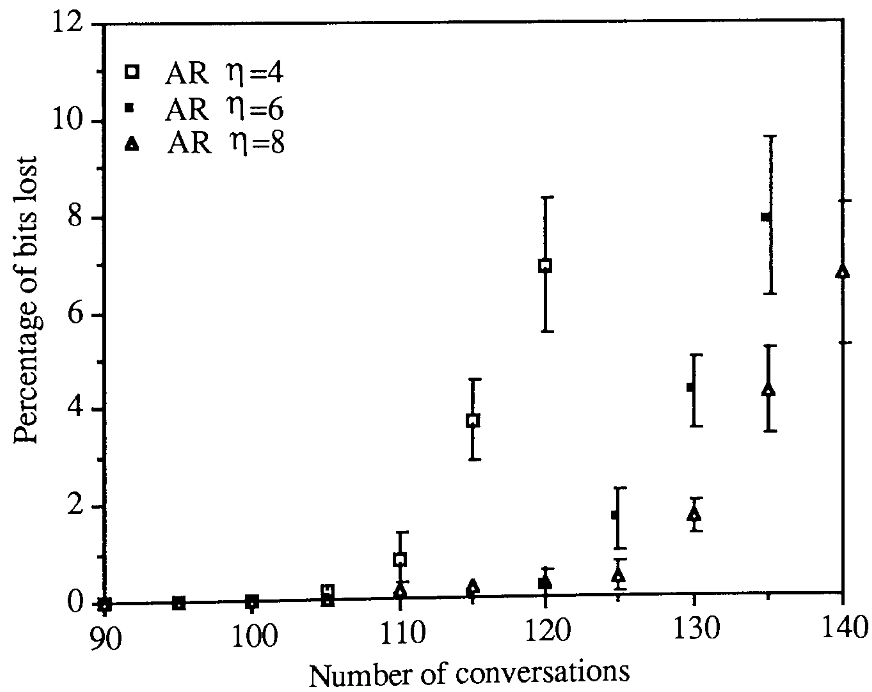


Figure 4.4 Percentage of voice bits lost using AR-VT-CSMA/CD with Tmin=10ms and Tmax=15ms.

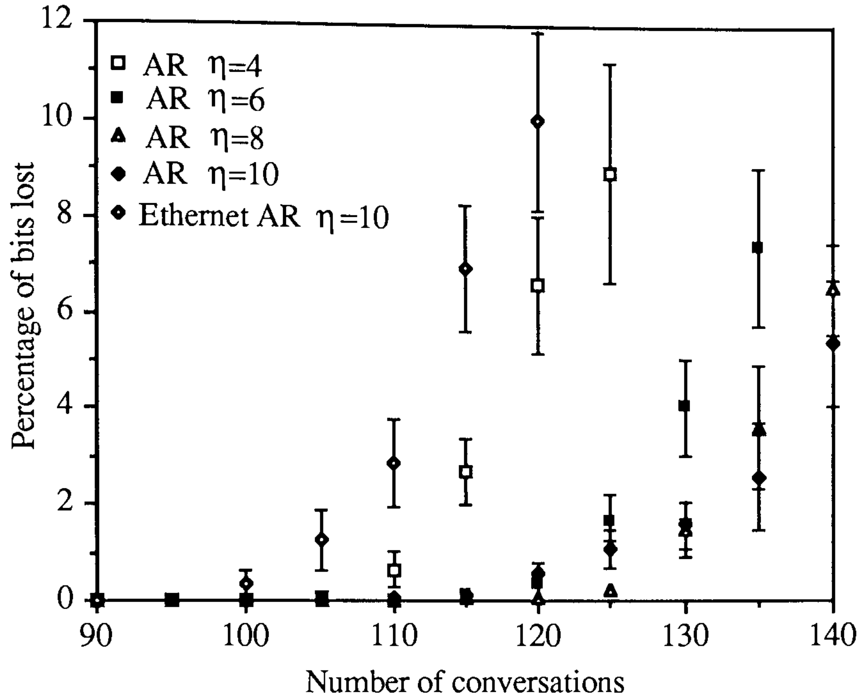


Figure 4.5 Percentage of voice bits lost using AR-VT-CSMA/CD with Tmin=12.5ms and Tmax=15ms.

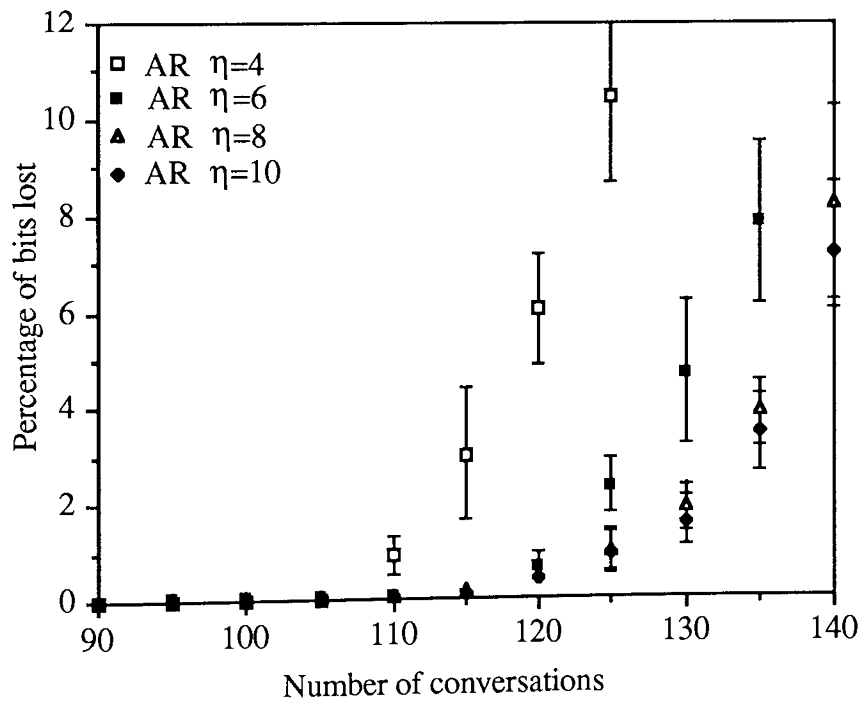


Figure 4.6 Percentage of voice bits lost using AR-VT-CSMA/CD with Tmin=14.9ms and Tmax=15ms.

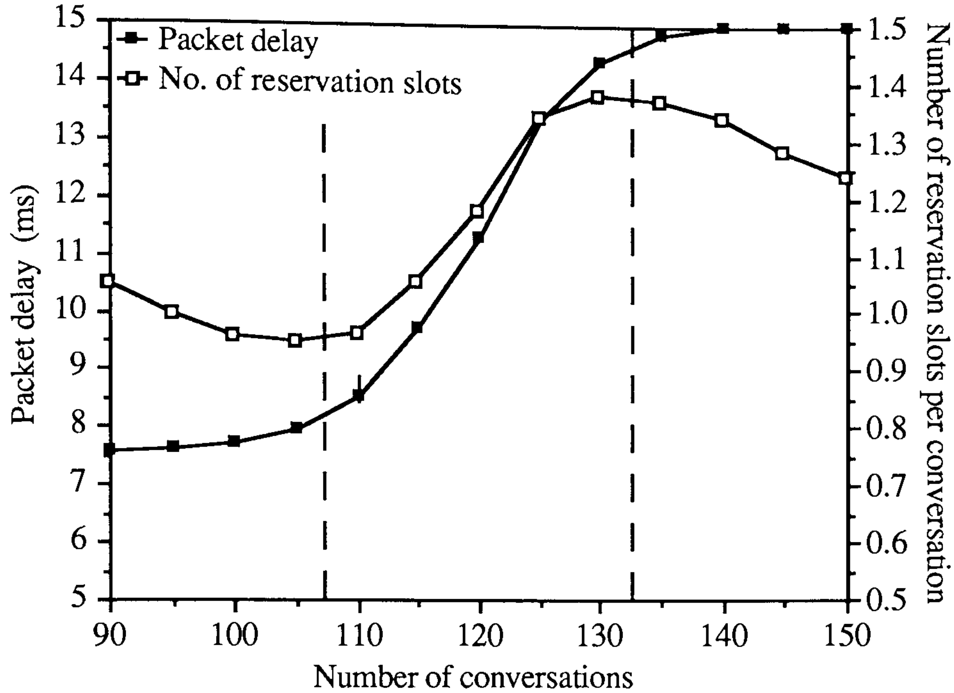
T_{min}	N_{max} $\eta=4$	$\eta=6$	$\eta=8$	$\eta=10$
7.5ms	112	125	132	_
10ms	113	127	132	_
12.5ms	114	128	133	134
14.9ms	111	124	130	131

Table 4.1 Maximum number of supportable conversations using AR-VT-CSMA/CD at various values of T_{min} and with T_{max} =15ms.

It can be seen from table 4.1 that the number of supportable conversations increases as T_{min} increases due to lower per-packet overhead, and then decreases at T_{min}=14.9ms due to the strict access time of 0.1ms imposed for no bit loss.

The percentage of bits lost for T_{min}=7.5ms and η=8 oscillates as the voice load increases. This oscillation is due to the variation of the effective number of reservation slots as the load increases. At low load, stations do not have to wait too long before the virtual time clock pass their arrival times. This then implies that stations are often separated on the virtual time access by the minimum packet transmission time. As the load increases, packet delay also increases. Since stations transmit all the voice bits they have acquired, the packet delay is the effective packetisation delay. As this packetisation delay increases, so also does the number of reservation slots, $\eta\tau$, within the packetisation interval, up to a maximum of $T_{max}/\eta\tau$. The more reservation slots there are, the easier it is for new stations to find a reservation slot and so reduce the number of collisions.

Figure 4.7 illustrates the number of reservation slots available as the packet delay increases with voice load. With η=8 and τ=10μs, a reservation slot equals one transmission time at low load. However at higher loads, this reservation time interval becomes a smaller fraction of the packet transmission time i.e. more reservation slots for each conversation. At high loads the packet delay is 15ms and so the number of reservation slots available in this packetisation interval of 15ms is now maximum. Thus at this point, as the number of conversations increases, the number of reservation slots per conversation now decreases.



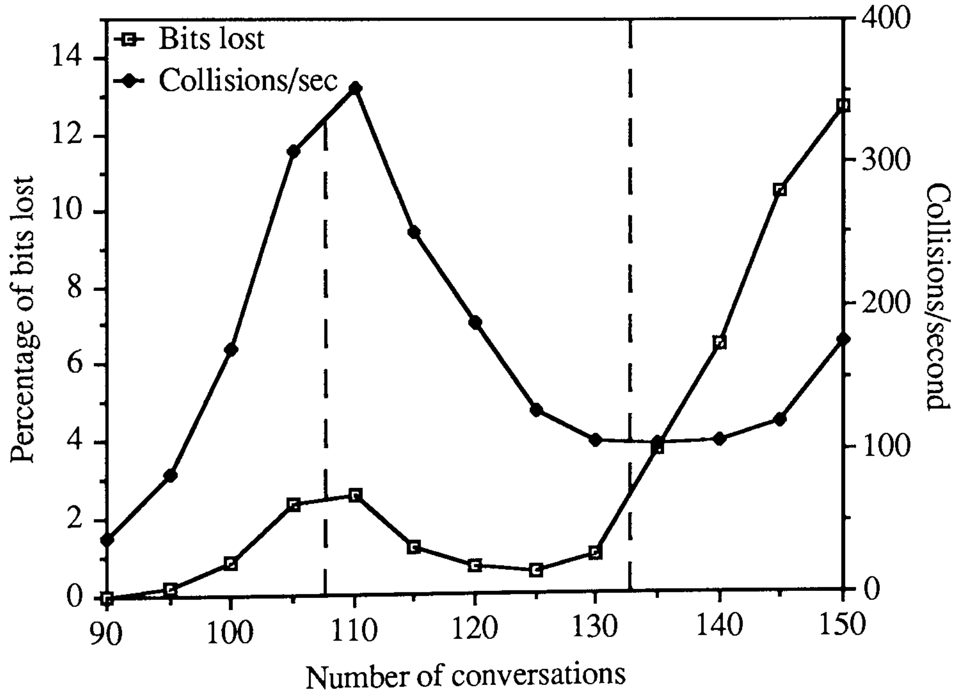


Figure 4.7 The performance of AR-VT-CSMA/CD when η=8 and Tmin=7.5ms and Tmax=15ms.

It can be seen from figure 4.7 that when the number of reservation slots per conversation within a frame is a minimum, the number of collisions per second is a maximum. This high collision rate causes packets to be rescheduled to future virtual times and so increases the percentage of lost voice bits due to stations having to wait too long. When the number of reservation slots per conversation is a maximum, the collision rate is a minimum. In this state however, voice bits are lost due to stations having to wait too long to transmit their packets - not due to retransmissions, but due to the virtual time clock lag increasing since the $\eta\tau$ can only be up to the minimum packet transmission time. This oscillation effect is not so pronounced for lower values of η, because in this case there are many more reservation slots available. For large η values at higher values of T_{min}, the effect again is not so evident since there is less variation in the packet delay as the load increases and hence the number of reservation slots available does not vary so significantly.

Table 4.2 illustrates the effect of η on the voice clip statistics. As η increases, the number of available reservation slots decreases and so new talkspurts find it increasingly difficult to find a free slot. This can be seen especially by the increase in the maximum clip statistic. Hence, although larger values of η can support more conversations, the distribution of maximum clip statistics may subjectively be unacceptable to listeners. The Ethernet-type protocol is again inferior to the standard protocol due to the overhead in the collision handshaking.

	N_{max}	Clip Length (ms) Mean	Std dev	Max	Reservation slots/con.
$\eta=4$	114	0.46	0.38	1.84	3.21
$\eta=6$	128	0.49	0.83	7.17	1.93
$\eta=8$	133	0.62	1.25	125	1.38
$\eta=10$	134	0.83	7.94	470	1.09
Ethernet $\eta=10$	107	0.64	2.73	183	1.32

Table 4.2 Clip statistics for AR-VT-CSMA/CD with T_{min} =12.5ms and T_{max} =15ms at 2% bit loss.

The delay and standard deviation of delay for Tmin=7.5ms,10ms and 12.5ms are shown in figures 4.8, 4.9 and 4.10 respectively.

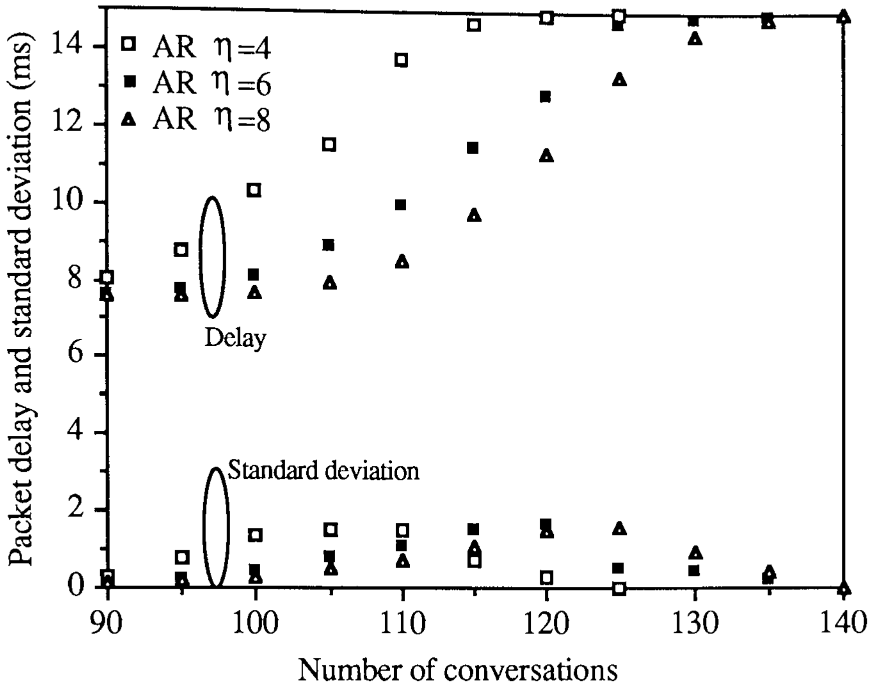


Figure 4.8 Packet delay and standard deviation using AR-VT-CSMA/CD with T_{min} =7.5ms and T_{max} =15ms.

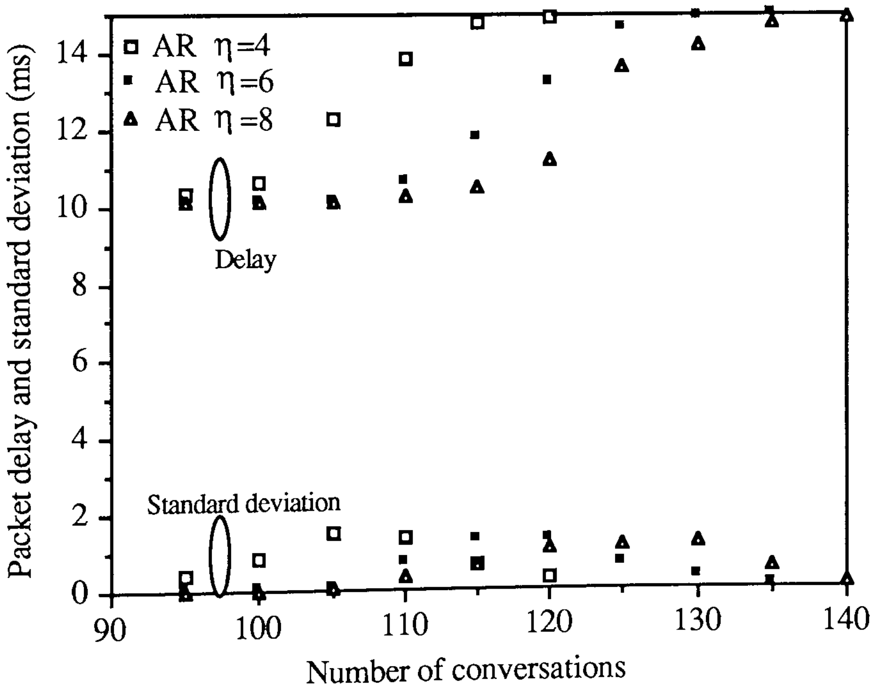


Figure 4.9 Packet delay and standard deviation using AR-VT-CSMA/CD with T_{min} =10ms and T_{max} =15ms.

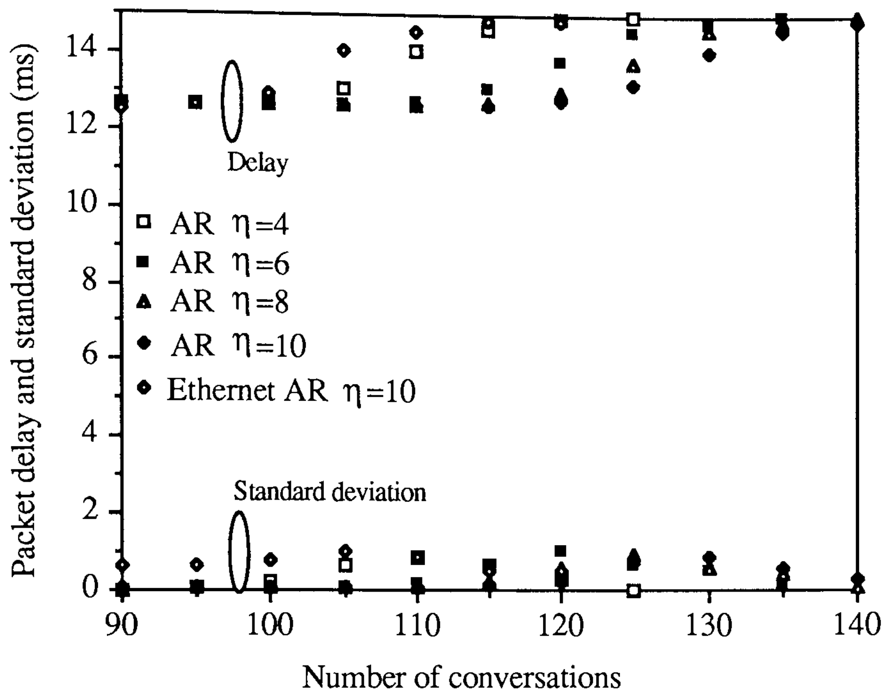


Figure 4.10 Packet delay and standard deviation using AR-VT-CSMA/CD with T_{min} =12.5ms and T_{max}=15ms.

Obviously as T_{min} increases so does the mean delay. With this convergence of the minimum and maximum delay of successful voice packets, the standard deviation decreases. It was seen from figures 4.4 to 4.6 that bit loss is more severe at lower values of η. This is verified from the delay curves by noting that at lower η values, the delay tends towards the maximum of T_{max} earlier than the other curves. The reason for this is that bits have to wait longer to be transmitted due to a slower virtual time clock update.

Data packets can only be transmitted in the AR-VT-CSMA/CD protocol once the voice virtual time clock is up-to-date. The same principle applies for data transmission as described in chapter 3 for the R-VT-CSMA/CD and the DC-R-VT-CSMA/CD protocols. The amount of data that can be transmitted depends on what proportion of the time there is no backlog on the voice virtual time clock. To guarantee some data transmission, a specific fraction of the channel time could be solely for the purpose of data transmissions. Figure 4.11 compares the amount of time the voice virtual time clock is up-to-date for the Adaptive, Dual Channel and original Reservation protocols.

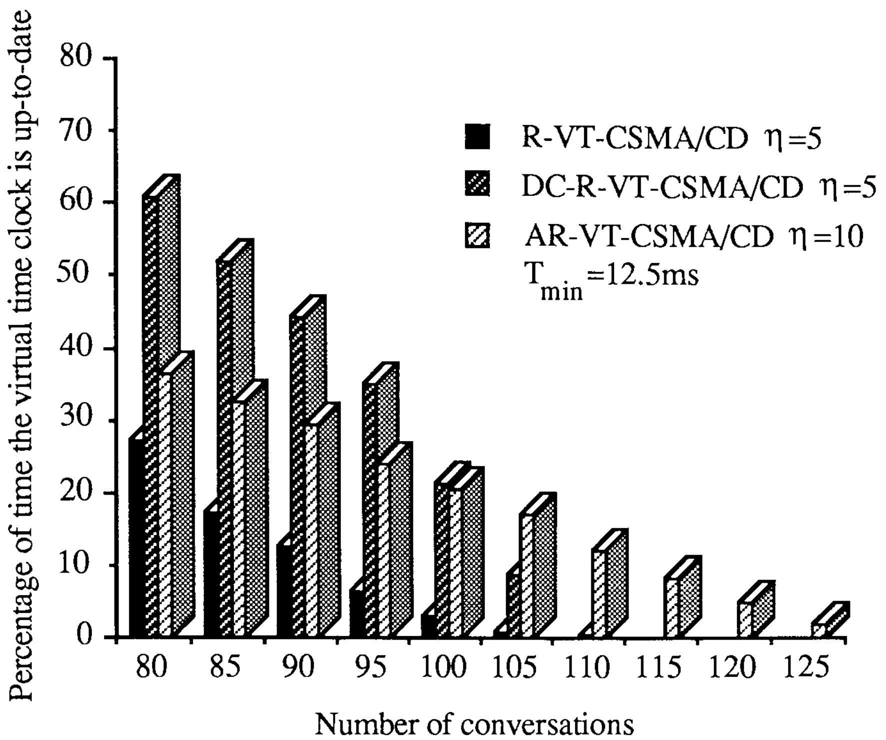


Figure 4.11 A comparison between R-VT-CSMA/CD, DC-R-VT-CSMA/CD and AR-VT-CSMA/CD of the amount of time the virtual time clock is up-to-date with the real time clock.

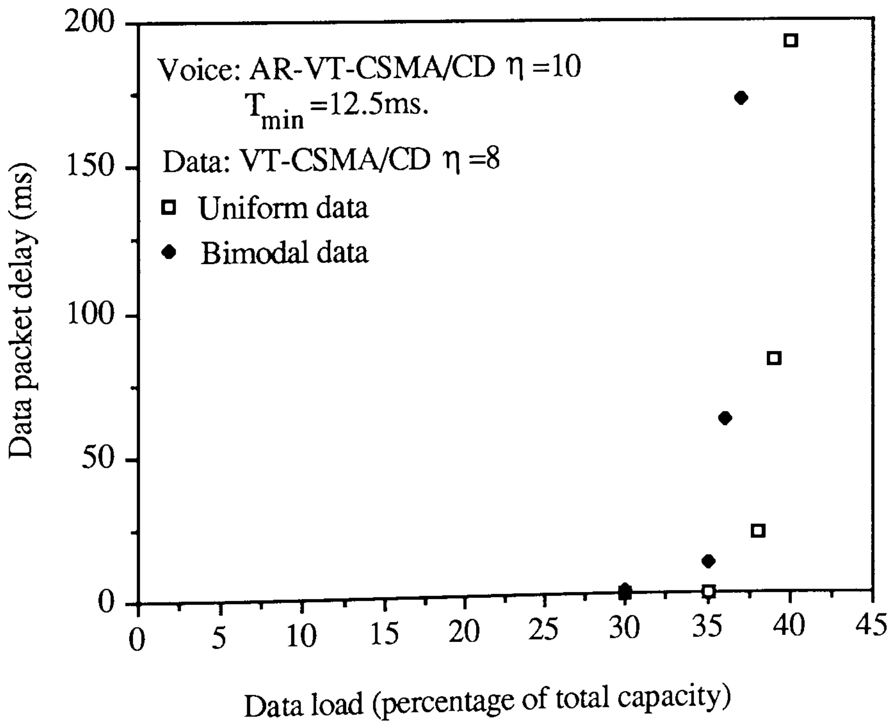


Figure 4.12 Data packet delay at various data loads and for a constant voice load of 40 conversations.

It can be seen that at low voice loads, the dual channel protocol spends more time up-to-date than the adaptive protocol because the dual channel protocol can update in parallel with a transmission whereas the adaptive protocol must update in a serial manner. However, as the load increases, the dual channel protocol also must spend time updating after the transmission is complete i.e. in serial. Since the adaptive protocol serially updates in larger jumps of $\eta\tau$, it spends longer up-to-date periods at higher voice loads than the dual channel protocol.

Figure 4.12 illustrates the amount of data load the VT-CSMA/CD protocol can support with a constant voice load of 40 conversations using AR-VT-CSMA/CD, before the data packet delay becomes unstable. Again the dual channel protocol can support slightly more, since at this light voice load all virtual time update is performed in parallel. It can be seen from figure 4.11 that at higher voice loads, the opposite will be true.

4.3 Logical TDMA

All the protocols described earlier have taken advantage of the fact that voice packets arrive periodically during talkspurts. They have scheduled them onto the channel using implicit reservations based upon their packet arrival times. However, since implicit reservations are used, collisions between established and new stations can still occur. In this section a scheme is examined whereby stations which have just started a new talkspurt, contend to join a distributed global queue through explicit reservations.

The objective is a distributed scheduling scheme, that will coordinate the channel access and transmissions of each station. If voice stations in talkspurt could be arranged into a distributed global queue, then a station could transmit onto the channel at a time depending upon its queue position. Since voice traffic is highly predictive due to the large number of periodic packets produced in a talkspurt, if stations contend to join this distributed queue only once in a talkspurt, then the contention overhead per-packet will be small.

Data traffic can be transmitted using either a fixed boundary or a movable boundary approach. The former allows data transmissions only within the data region of the frame, whereas the latter allows data transmissions within the data region plus any capacity in the voice region not being used by the voice traffic. This protocol therefore has the advantage that it provides collision-free transmissions for voice packets in the global queue, whilst providing periods of time for data and talkspurt set-up packets, necessary for voice stations to join the queue, using CSMA/CD.

Several authors have examined this approach [12,13,100]. The Ethernet compatible protocol of [13] used collision-handshaking to protect the order of voice transmissions from being disrupted by data packet transmissions. An active voice station remembers which station precedes it in the queue and would know when it could transmit by sensing the completion of its predecessors transmission. When a station enters a silent period, it transmits a terminate packet so that its successor can update its predecessor to the predecessor of the station which has just terminated. The approach adopted in [100] achieves this global synchronisation by assigning each station a sequence position. Each station knows when it can transmit by counting the number of transmissions that have occurred so far from the beginning of the frame. During the silent periods voice stations transmit packets, but with no voice information so that they can maintain their queue position.

A similar scheme is adopted in [12] except that voice stations relinquish queue positions at the beginning of each silent period. Although stations must then contend again to join the queue at the commencement of each talkspurt, results in [12] illustrate that contention is quickly resolved if voice packets have priority over data packets. Hence the amount of overhead in keeping the global position during silent periods will generally greatly exceed the amount of contention time necessary to join the queue at the beginning of each talkspurt. An approach similar to that in [12] has been adopted, called Logical Time Division Multiple Access (Logical TDMA). In this Logical TDMA protocol, each active station has a logical queue position, i, and begins its transmission after $i-1$ transmissions have been sensed in the current frame.

A conversation entering the talkspurt state will first contend to join the distributed queue. Once joined, the conversation will have collision-free transmissions. Upon entering the idle state, the conversation then leaves the queue. The channel time is partitioned into frames and each frame is divided into one voice region and one data region. At the commencement of each frame, the voice packets are transmitted collision-free in the voice region, the order dictated by a stations global queue position. Once all the queued voice stations have been transmitted or the voice region is complete, the data region commences which consists of data and talkspurt set-up packets using a CSMA/CD algorithm. The frame structure is illustrated in figure 4.13. The Ethernet-type compatible protocol again uses collision-handshaking.

In the absence of a data region, a frequency-division scheme would be possible. However, the advantage of using a time-division approach other than the fact that different frequency filters are not required, is that there is a compensation effect for a fixed number of stations on the LAN. For example, the more stations that are queued, fewer stations will require contention, so more channel time is utilised for the queued packet transmissions. The opposite is also obviously true. This compensation effect is not available with the frequency-division approach.

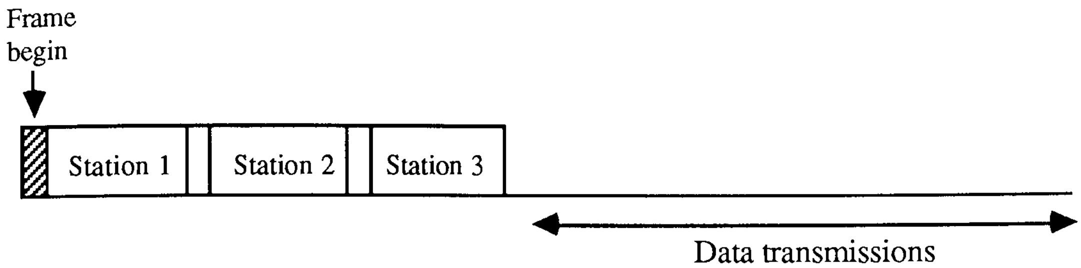


Figure 4.13a Standard Logical TDMA

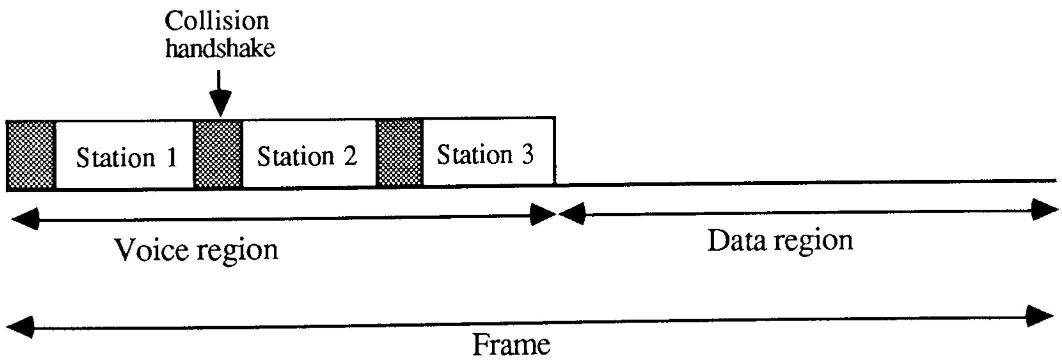


Figure 4.13b Ethernet-type compatible Logical TDMA

To implement the global queue, a station x requires three counters; σ, the total number of stations queued, i_x, the position of station x in the queue, and j, the current queue position. A station, x, which commences a talkspurt, attempts to join the queue in the data region. After successful contention, station x is allocated queue position $i_x = \sigma + 1$. All stations then update $\sigma = \sigma + 1$. A control packet at the beginning of each frame could be broadcast by the station at the head-of-the-queue, so that any stations who have just commenced a talkspurt will know the value of σ. Control information in a voice packet header from a station broadcasts whether or not a station is terminating its queue position,

i, in the current frame. If this is the case, then all stations with queue positions $>i$ decrement their queue position by 1.

The maximum number of packet transmissions in the voice region, L, is limited by the length of the voice region, the channel capacity, the voice encoding rate, the per-packet overhead and the network propagation time. For the situation where $\sigma>L$, a fixed length packet protocol as in [12] has been examined and two variable length packet schemes have been developed.

4.3.1 Fixed Length Packet Logical TDMA

Stations contend to join the queue once the packetisation interval for the first talkspurt packet is complete. If σ exceeds the maximum number of allowed packet transmissions in the voice region, L, a round-robin service discipline is used to guarantee fairness in accessing the channel. This has the effect of a cyclic queue in which a station in queue position i changes position after L transmissions i.e. at the end of each voice region, by $i = (i+\sigma-L) \bmod \sigma$ in preparation for the next frame. A non-queueing service discipline was adopted [12], where all voice packets that arrived in the previous frame are either transmitted in the current frame or discarded once L voice packets have been transmitted. This is illustrated in figure 4.14.

Although this scheme distributes the channel capacity fairly among the stations, it has the disadvantages that the minimum clip length is a frame time, which for long frame times is subjectively undesirable, and the maximum bound on the packet delay in the LAN, including the packetisation delay is now three frame times. These disadvantages can be overcome by using the variable length packet protocols developed.

4.3.2 Variable Length Packet Logical TDMA

Suppose that when a station commences a talkspurt, it contends to join the distributed queue whilst still in its packetisation period. It is assumed that stations have a buffer of maximum length P_{max}, as described in section 4.2. Then the delay of this packet has an upper bound of two frame intervals, if $P_{max}/V=2F$. Two versions of the variable length packet protocol have been examined:

```
{j is the position of the next station ready to transmit in the current frame}
begin
    repeat
        if beginning of frame then
            j=0;
        j=j+1;
        if i=j then                                          {transmission by station i}
            if final packet in talkspurt then
                begin
                    set control bits to silent;
                    transmit packet;
                    σ=σ-1;
                    j=j-1;
                end
            else
                begin
                    set control bits to talkspurt;
                    transmit packet;
                end
        else                                                 {transmission by station j}
            if control bits on current transmission set to silent then
                begin
                    σ=σ-1;
                    if i>j then
                        i=i-1;
                    j=j-1;
                end
    until j=min{L,σ}
    if j<i then
        discard packet generated in previous frame;
    if σ>L then
    i = (i+σ-L) mod σ;
end
```

Figure 4.14 Round-Robin fixed length packet Logical TDMA: Protocol executed by station i.

4.3.2.1 Standard Variable Length Packets

If a queued station does not transmit in a frame, it is again cyclically rotated to a position nearer the head-of-the-queue using the round-robin service discipline. However, the previous packet is now no longer discarded. A variable length packet is used with a maximum length of P_{max} , which has a maximum delay of P_{max}/V, two frame times. A station transmits its complete buffer contents, up to P_{max} , at its scheduled queue position. Voice loss now only occurs when voice bits have to wait longer than P_{max}/V.

This has the advantage over the fixed length scheme that as the load increases, per-packet overhead is reduced, voice clips do not have a minimum length of one frame and the packet delay is bound by two frame times.

4.3.2.2 Variable Length Packets With Scaling Factor

At the beginning of each frame, the available capacity is divided amongst all the stations on the queue, σ, based on the assumption that each station has one frame time of voice bits. This division of capacity will not always be equal since new stations which joined the queue in the previous frame can have up to two frames of speech. If minimum size packet constraints are neglected, this imbalance should be cancelled out by stations which are finishing in the current frame and do not have one frame of speech to transmit. Since all stations on the broadcast network know the value of σ, each station calculates a *Scaling Factor* to determine the proportion of voice bits it can transmit. Obviously the scaling factor must always be greater than zero. The scaling factor is equal to:

$$\text{Scaling Factor} = \text{Min}\left\{ \frac{(\text{Voice region} . C - \sigma . \text{Packet overhead})}{\sigma . V . F} , 1 \right\}$$

where C, V and F are the channel capacity, voice encoding rate and frame time respectively. Thus a voice station is allowed to transmit the number of bits in its buffer multiplied by the scaling factor. The remainder of bits, if any, are discarded. With this scaling mechanism, there is the subjective advantage of minimising the length of speech clips fairly amongst the stations, unlike the fixed packet scheme which although fair, uses a minimum clip of one frame time. However, the cost of this scaling factor is a larger proportion of packet overhead transmitted per voice region.

This equation is adequate for the asynchronous mode of operation. The synchronous mode requires stations to transmit at the beginning of a slot and so, to maximise channel efficiency, packet transmission lengths should be an integer number of slots, where a slot is the end-to-end propagation time. For example, when the scaling factor is 1, assume that a packet transmission including all overhead requires 8 slots. At overload, if a scaling factor is introduced which decreases this transmission time to 6.7 slots say, the 0.3 slots between 6.7 and 7 will be wasted capacity. Therefore, for the synchronous version, the scaling factor is increased at a particular station if the transmission time is a non-integer number of slots up to the next highest integer. In this situation, not all the queued voice stations may be able to transmit in the current frame. These stations are then cyclically placed at the front of the queue in the next frame using the round-robin service discipline. Hence in this version of logical TDMA described in figure 4.15, bit loss occurs only due to the scaling factor and bits discarded if they are delayed longer than two frame intervals. Figure 4.15 also applies to the standard variable length packet Logical TDMA of 4.3.2.1, except that the scaling factor is always 1 and hence bits are only discarded when their waiting time exceeds two frame times.

To summarise, the disadvantage of this protocol compared to the fixed length version is that at high load the total overhead in the voice region will be greater and this will obviously lead to increased voice bit loss. However, it has the advantage that packet delay is bound by two frame times for the first talkspurt packet and is usually just the packetisation delay for subsequent packets. Also the bit loss at overload is evenly spread amongst the stations *each* frame time, which will reduce the mean clip length compared to the fixed length scheme. The amount of bit loss is now approximated for this and the fixed length packet scheme.

4.3.3 Voice Loss With Logical TDMA

If voice packets have preemption headers over data packets, the mean voice contention time in the data region can be calculated using equation (3.5). If the data region is sufficiently long, then any new talkspurts that arrive in a frame will succeed in their contention to join the queue at the end of the voice region in the frame in which they arrived. Using this assumption, the network can now be viewed as a centralised system.

Consider first, the fixed packet length protocol. Assume N voice stations, of which N_{talk} are in the talkspurt state delivering packets once a frame time, F, to a centralised queue. If, out of the N_{talk} packets which arrive, there is a limit, L, on the number that can be transmitted every F seconds, the total number of packets offered then, over a long period of time X.F is given by:

```
{j is the position of the next station ready to transmit in the current frame}
begin
    repeat
        if beginning of frame then
            begin
                j=0;
                scale = min{ (voice region.C-σ.packet overhead)/(σVF) , 1};
            end
        j=j+1;
        if i=j then                                          {transmission by station i}
            begin
                if (packet overhead+scaling factor.buffer contents)/C.τ ≠ integer then
                    increase scale for integer slot transmission time;
                if final packet of talkspurt then
                    begin
                        set control bits to silent;
                        transmit packet;
                        if scale < 1 then
                            discard excess voice bits;
                        σ=σ-1;
                        j=j-1;
                    end
                else
                    begin
                        set control bits to talkspurt;
                        transmit packet;
                        if scale < 1 then
                            discard excess voice bits;

                    end
            end
        else                                                 {transmission by station j}
            if control bits of current transmission are set to silent then
                begin
                    σ=σ-1;
                    if i>j then
                        i=i-1;
                    j=j-1;
                end
    until j=σ or voice region exceeded
    if σ>j then
        i = (i+σ-M) mod σ
end
```

Figure 4.15 Round-Robin variable length packet Logical TDMA: Protocol executed by station *i*.

$$\text{Packets offered} = X \sum_{n=0}^{N} n \, P\{N_{talk} = n\}$$

$$\text{where} \quad P\{N_{talk} = n\} = \binom{N}{n} \gamma^{n} (1-\gamma)^{N-n}$$

$$\therefore \quad \text{Packets offered} = X \sum_{n=0}^{N} n \binom{N}{n} \gamma^{n} (1-\gamma)^{N-n} \qquad \ldots(4.3)$$

where γ is the proportion of time a station is in the talkspurt state.

Packets will be discarded once the queue length exceeds L. Over the same period of time X.F, the number discarded is given by

$$\text{Packets discarded} = X \sum_{n=L+1}^{N} (n-L) \, P\{N_{talk} = n\}$$

$$= X \sum_{n=L+1}^{N} (n-L) \binom{N}{n} \gamma^{n} (1-\gamma)^{N-n} \qquad \ldots(4.4)$$

Therefore, the fraction of packets lost is the ratio of (4.4) to (4.3) and is given by [117]

$$P_{loss} = \frac{1}{\gamma N} \sum_{n=L+1}^{N} (n-L) \binom{N}{n} \gamma^{n} (1-\gamma)^{N-n} \qquad \ldots(4.5)$$

Consider now, the asynchronous variable length packet version of 4.3.2.2, again using the centralised queue assumption. The scaling factor divides up the available capacity in the voice region between all the queued stations. The fraction of voice bits lost can be expressed as

$$\text{Fraction of voice loss} = \text{Max}\left\{ \frac{\left(\sigma.\text{Voice bits per packet} + \sigma.\text{Overhead per packet} - \text{Total bits in voice region}\right)}{\sigma.\text{Voice bits per packet}} , 0 \right\}$$

Over a period of time X.F, the total number of voice bits generated is given by

$$\text{Total voice bits generated} = X F V \sum_{n=0}^{N} n\, P\{N_{talk} = n\}$$

$$= X F V \sum_{n=0}^{N} n \binom{N}{n} \gamma^{n} (1-\gamma)^{N-n}$$

$$= X F V N \gamma \qquad \text{...(4.6)}$$

where V is the voice encoding rate. Therefore, an approximation for the fraction of bits lost using the variable length packet version is

$$P_{loss} = \text{Max}\left\{ \frac{N\gamma\ (VF + \vartheta)\ - T_v C}{V F N \gamma}\ ,\ 0 \right\} \qquad \text{...(4.7)}$$

where ϑ is the total per-packet overhead and T_V the duration of the voice region. The fixed and variable length versions of Logical TDMA are now examined through simulation.

4.3.4 Simulation Results Of Logical TDMA

The various versions of synchronous Logical TDMA have been examined on an error-free 10Mbps LAN with an end-to-end propagation delay of 10μs. Voice packets are encoded at 64Kbps with a frame time of 7.5ms.The simulation model is described in greater detail in chapter 6.

Figure 4.16 illustrates the voice loss for the protocols with no data load when the voice region has a length of 3.76ms. This leaves 3.74ms for new voice stations to contend to join the queue. The simulation results show that the contention for all new stations is successfully achieved in the frame in which they arrived. Therefore, this configuration can be considered as a centralised system. The theoretical points for the variable packet lengths with scaling are slightly pessimistic since the asynchronous analysis assumes that every queued station, including its overhead, is transmitted. However, in the synchronous version modelled, stations may use larger scaling factors to utilise an integer number of slots and so not all queued stations may transmit. Thus in the synchronous version at high load, less overhead is transmitted in the voice region. The 'talkspurt setup' packets are assumed to be 400 bits in length. They contend to join the distributed queue in the data region using slotted CSMA/CD.

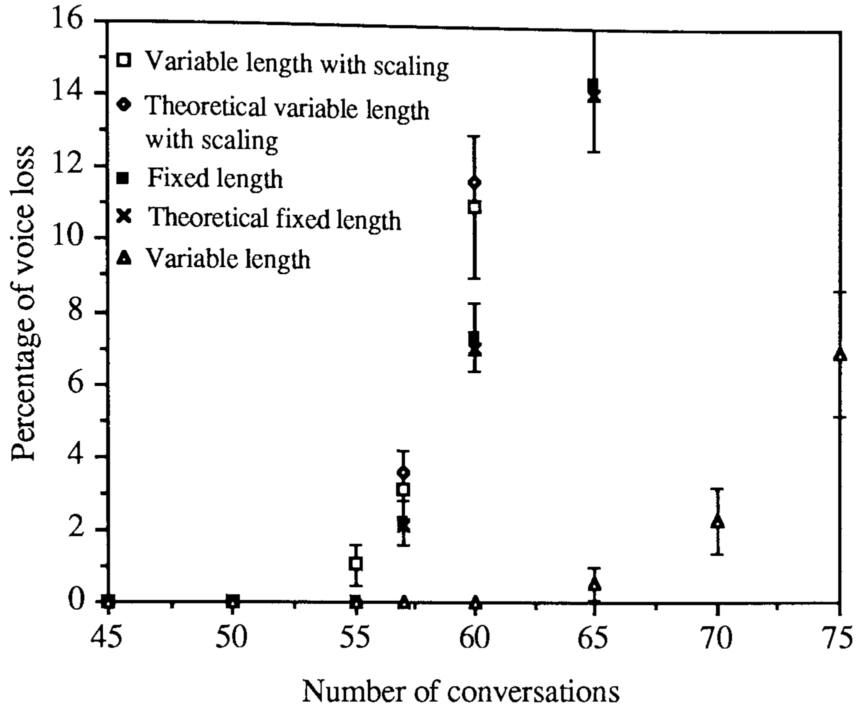


Figure 4.16 Percentage of voice loss at various voice loads using Logical TDMA with a voice region of 3.76ms.

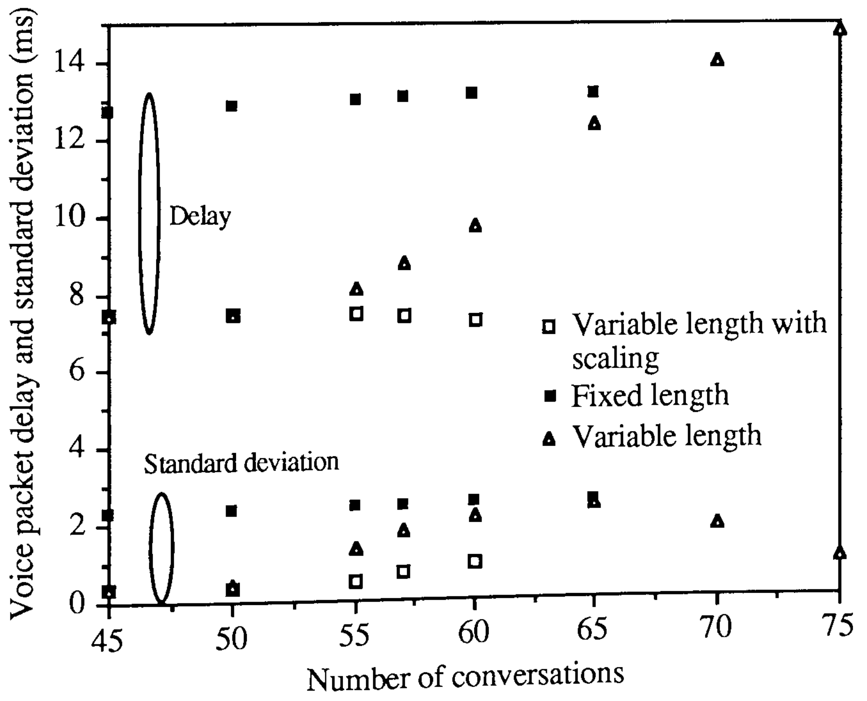


Figure 4.17 Voice packet delay and standard deviation of delay at various voice loads using Logical TDMA with a 3.76ms voice region.

As the voice load increases, the variable length packets with scaling attempt to divide the available capacity amongst the queued stations. The percentage of voice loss is thus higher with this scheme since there is more overhead in each voice region. The delay for most packets is just the packetisation delay, thus as the load increases and the packet sizes are scaled down, the mean delay decreases.

The fixed length version can support a larger number of conversations than the variable length packets with scaling, due to reduced overhead per voice region. However, although the mean packet delay is below the 15ms constraint, the maximum packet delay is bound by three frame times. The delay has three main components; the packetisation delay, the deferred service delay and the voice region delay. The deferred service delay is due to the fact that a packet arriving in the current frame will not be transmitted until the next frame.The average size of this component is F/2. The voice region delay is the amount of time on average a station must wait to transmit its packet. This is $T_V/2$ scaled by the number of allowed queued stations. Thus the mean total delay is given as

$$\text{Mean total delay for fixed length packet version} = F + \frac{F}{2} + \frac{T_V}{2} \cdot \frac{\min\{L, N\Upsilon\}}{L} \quad \ldots(4.8)$$

At high load, this is equal to 13.13ms, which corresponds well with the simulation.

The variable length packets without scaling can support the largest number of conversations due to reduced overhead in the voice regions. As the load increases, more stations are cyclically rotated at the end of a frame. This causes them to wait longer before their transmission and so it can be seen from figure 4.17 that as the load increases, the packet delay tends towards two frame periods.

The clip statistics for these three versions of Logical TDMA with a voice region of 3.76ms and a frame time of 7.5ms are illustrated in table 4.3. It can be seen from the fixed length statistics that packet loss only occurs when packets are discarded at the end of a voice region, since the maximum clip length is one frame time. Thus all new stations are able to join the queue in the frame in which they arrived, so the assumption of a centralised system was valid. Instead of discarding whole packets at the end of a frame, it can be seen that smaller mean clip times can be achieved through variable length packets, either from a scaling factor or by just discarding voice bits that exceed the 15ms time constraint. Without a scaling factor, it can be seen that the maximum length clip for variable length packets can however, exceed a frame time. This can occur at high loads due to stations sometimes having to be queued for two frame times before they can be transmitted.

	N_{max}	Clip Length (ms) Mean	Std dev	Max
Fixed length	57	7.50	0.00	7.50
Variable length with scaling	56	0.61	0.58	2.38
Variable length	70	0.71	1.20	8.33

Table 4.3 Clip statistics for Logical TDMA with frame time = 7.5ms and voice region = 3.76ms at 2% voice loss.

The Logical TDMA protocol is now examined to find the maximum number of supportable conversations. Various voice region lengths were examined within the 7.5ms frame. A voice region of 7.44ms was found to maximise the number of conversations. Obviously as the load increases, the contention time increases and the network may no longer be considered to be a centralised system. Figure 4.18 illustrates the percentage of voice loss using the various versions. Again it can be seen that the variable packet protocol with no scaling is superior. The Ethernet-type version is also illustrated, which again can support fewer conversations due to the collision handshaking procedure.

Figure 4.19 illustrates the packet delay. The fixed packet delay is in general agreement with equation (4.8). At high load, where the system may not be considered centralised, the delay is slightly less than expected because the contention times may be so great for the contention period allocated, that the actual number of queued stations may actually decrease and so the minimum term in (4.8) is then in error. The delay standard deviation is larger for the fixed length packets compared to the variable length packets because they can have a maximum delay of three frame intervals whereas the latter is limited by two frame times. The variable length packets with scaling remains very close to the packetisation period. The variable length packets with no scaling does not tend towards the two frame time limit as in figure 4.17. This is due to new stations being unable to join the queue due to collisions with other new stations. Hence, as voice stations leave the queue, more queued stations will be able to transmit in a frame without waiting until the next frame. This will obviously mean that the voice buffers will contain fewer bits than if a station had to wait and so the delay is reduced. The price for this delay is, however, very large subjectively unacceptable clip statistics for new stations wishing to join the queue, as illustrated in table 4.4.

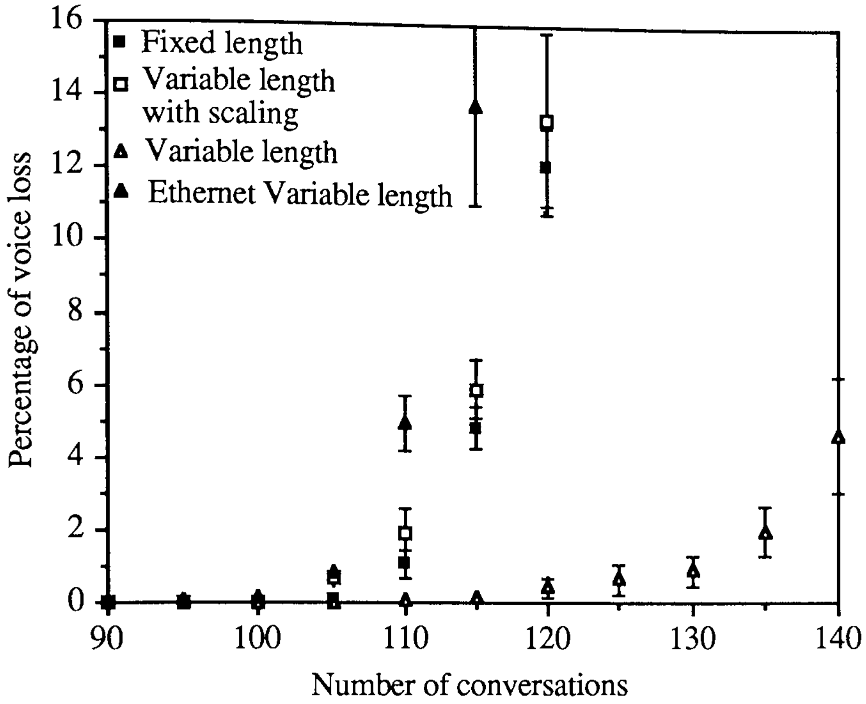


Figure 4.18 Percentage of voice loss at various voice loads using Logical TDMA with a voice region of 7.44ms.

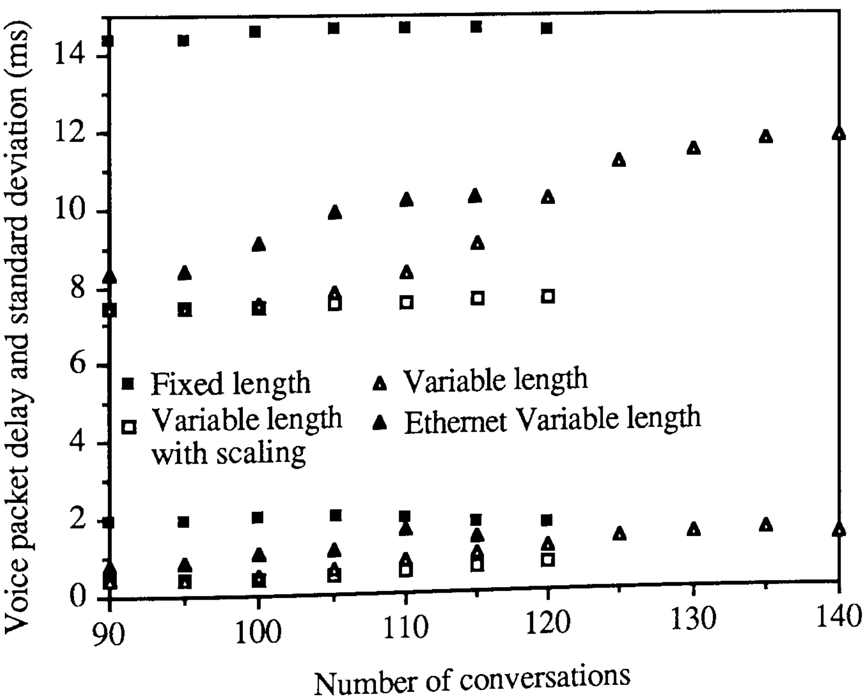


Figure 4.19 Voice packet delay and standard deviation of delay at various voice loads using Logical TDMA with a 7.44ms voice region.

	N_{max}	Clip Length (ms) Mean	Std dev	Max
Fixed length	114	10.06	21.05	480
Variable length with scaling	110	1.35	11.54	362
Variable length	135	22.30	32.49	506
Ethernet Variable length	109	11.83	22.83	416

Table 4.4 Clip statistics for Logical TDMA with a frame time = 7.5ms and a voice region = 7.44ms at 2% voice loss.

It can be seen from table 4.4 that the maximum clip statistics for the fixed length protocol is now much larger than with a voice region of 3.76ms, i.e. not all stations contend successfully to join the queue in the frame in which they arrived. Although the mean clip statistics may be acceptable to a listener, the maximum length clip statistics may actually loose syllables and so would be unacceptable. Hence to implement either version of the Logical TDMA protocol, a suitable contention period is necessary. This would usually be incorporated into the data region, with the voice packets having preemptive headers to gain priority over data packets. The network could then be considered to be a centralised system and similar clip statistics would be obtained as in table 4.3 with a maximum clip statistic of the order of one frame period.

Figure 4.20 illustrates the Logical TDMA protocol using variable length packets with a data load. There is a constant voice load of 40 conversations with a varying data load, using a movable boundary. Outside the voice region of 3.76ms, new voice stations had priority over data packets using slotted p-persistent CSMA/CD (p=0.5), which resulted in a centralised network.

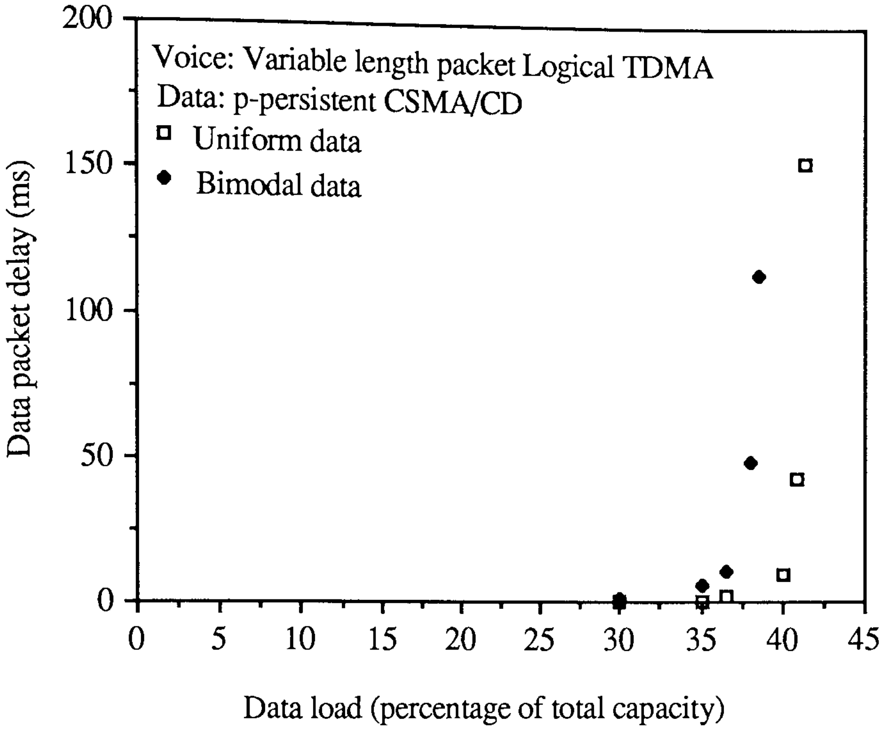


Figure 4.20 Data packet delay at various data loads with a constant voice load of 40 conversations.

4.4 Summary

In this chapter, two types of protocol have been developed using variable length packets. The first protocol called Adaptive R-VT-CSMA/CD used implicit reservations and enabled the use of larger virtual time update speeds, η, compared to the protocols in chapter 3, to reduce the amount of wasted channel time. This reduction in the virtual time update periods resulted in a superior performance compared to the other virtual time protocols.

The second class of protocol arranged voice stations into a distributed queue using explicit reservations to determine the transmission order. The advantage of these variable length packet protocols is that short length packets are used at low network load to reduce delay and longer packets at higher network loads to increase efficiency. The superior performance of these protocols compared to the CSMA/CD protocol are summarised in figures 4.21 and 4.22. The nonpersistent CSMA/CD protocol has a packetisation interval of 12.5ms, with a maximum bit delay of 15ms. The clip statistics for the three protocols are illustrated in table 4.5. It can be seen that the mean clip for the CSMA/CD protocol is significantly larger compared to the other two protocols. The logical TDMA protocol has a voice region of 7.44ms. As was mentioned earlier, if a larger data

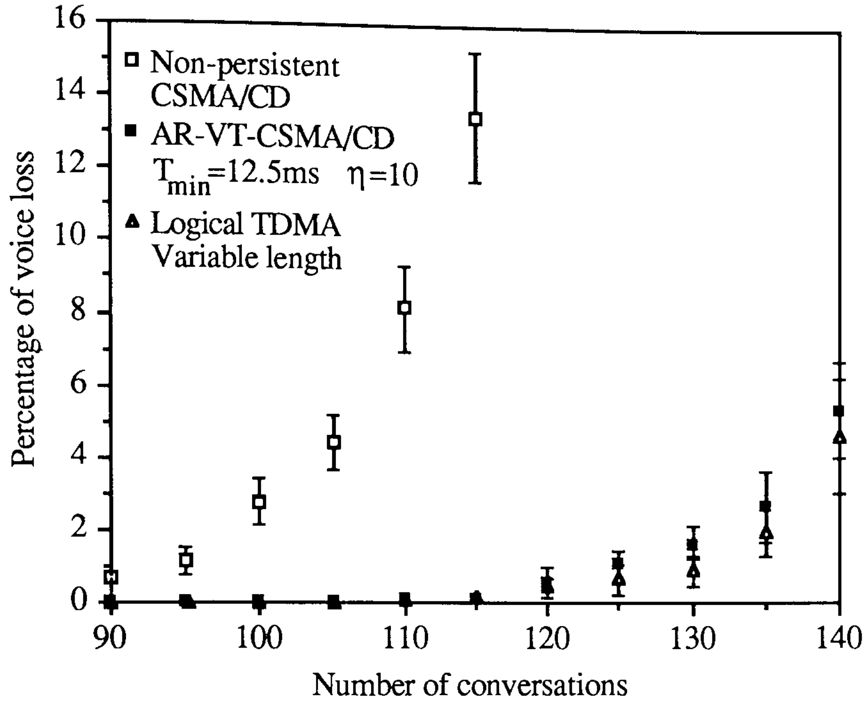


Figure 4.21 Percentage of voice loss at various voice loads using CSMA/CD, AR-VT-CSMA/CD and Logical TDMA.

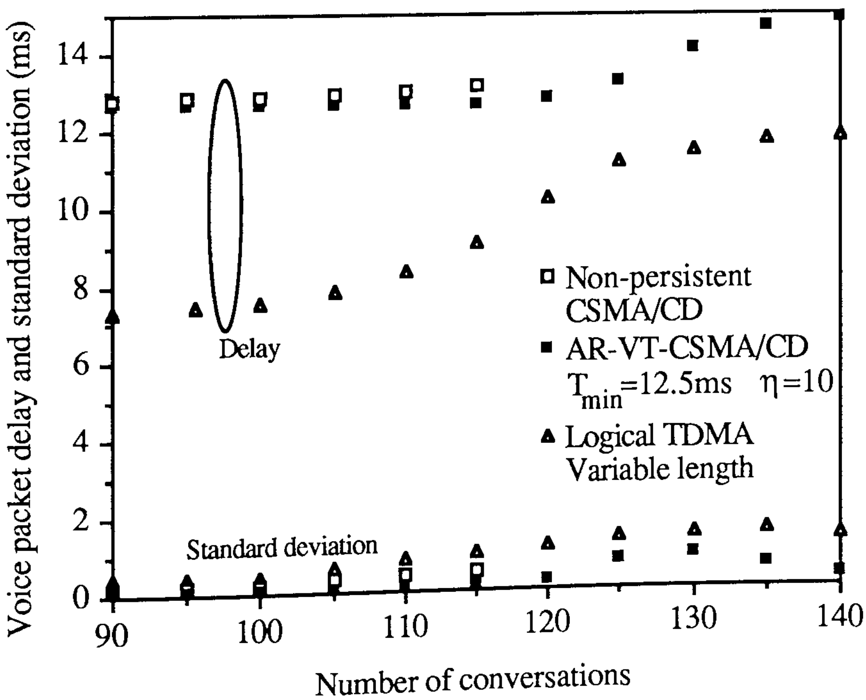


Figure 4.22 Voice packet delay and standard deviation of delay at various voice loads using CSMA/CD, AR-VT-CSMA/CD and Logical TDMA.

(contention) region is used, the maximum clip length can be reduced significantly. This is a significant advantage over the AR-VT-CSMA/CD protocol, since at high load it can be seen that the maximum speech clip with the adaptive protocol may be subjectively unacceptable to the listener.

	N_{max}	Clip Length (ms) Mean	Std dev	Max
Non-persistent CSMA/CD	97	41.76	63.91	468
Logical TDMA Variable length	135	22.30	32.49	506
AR-VT-CSMA/CD	134	0.83	7.94	470

Table 4.5 Clip statistics for CSMA/CD, AR-VT-CSMA/CD and Logical TDMA at 2% bit loss.

These variable length packet protocols can support a greater number of conversations on the network partly due to the reduction in per-packet overhead of longer packets. Figure 4.23 illustrates the effect of per-packet overhead on the protocols simulated at various equivalent frame times. It is termed equivalent frame time for reference to the fixed length protocols. For example, the AR-VT-CSMA/CD protocol limits voice delay to two equivalent frame times and not to two packetisation intervals.

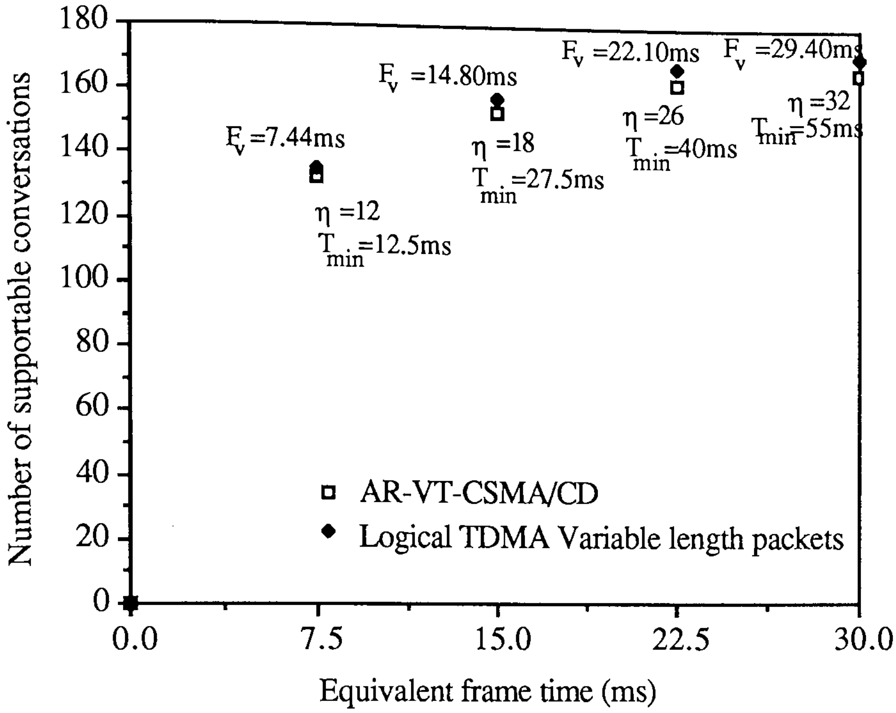


Figure 4.23 Number of supportable conversations, assuming 2% bit loss is acceptable, for AR-VT-CSMA/CD and Logical TDMA at various equivalent frame times.

CHAPTER 5

VOICE MAC PROTOCOLS IN A MOBILE RADIO ENVIRONMENT

5.1 Introduction

The recent growth of mobile communications started with cellular phones. In 1978, the U.S. cellular system AMPS (Advanced Mobile Phone Service) [125] was introduced. The TACS (Total Access Communications System) succeeded this, defining the air interface specification for the system. Since cellular networks were first introduced, there have been advances in digital technology. The use of digital instead of analogue voice provides various advantages, including more consistent voice quality throughout the range of the radio link by providing sophisticated protocols, encription for speech privacy, and the ability to integrate voice and data traffic. In Europe, digital cellular will be introduced in the early 1990s in the form of the new pan-European digital cellular network, GSM (Groupe Special Mobile), which will include all the national systems in Europe. The integration of the GSM system and ISDN is examined in [121], which explains how an ISDN exchange can be expanded to provide additional GSM functionality of a mobile services switching centre.

A mobile digital service has already been introduced into the U.K., a short-range system using the CT2 (second generation cordless telephony) standard. This cordless handset can be used in three basic ways: (i) With a domestic radio base station plugged into a standard telephone socket, operating as a conventional cordless phone; (ii) The same handset could be used at work, with a network of linked base stations to function as a cordless PBX; (iii) It can be used at a public *Telepoint* where radio base stations are linked to the PSTN (Public Switched Telephone Network). The systems main limitation arises from the fact that coverage only exists when in range of a radio base station which can be up to 200m [16].

Another form of mobile system is that of paging. These can be either local pagers for use in hospitals and offices etc., or wide area pagers that use the national system. The national system is based on the POCSAG (Post Office Code Standardisation Advisory Group) standard that was introduced into service in 1980 [93,106].

The objective of mobile communications is to provide a personal phone, which is cheap, small and satisfies weight and power consumption constraints. A handset in personal communications will then become an object associated with a person and not a place. This will allow people to make calls wherever they are likely to be. Cellular systems meet most of the requirements for facilities, but they are expensive. Cordless systems, such as CT2 are cheap but their main disadvantage is that radio coverage is not contiguous.

Due to the growing demand for mobile services, efficient and economical use of the allocated radio spectrum should be made. Conventional mobile radio systems, based on large coverage areas per base station, are not spectrally efficient and do not provide nationwide coverage. In the cellular system, geographical areas are split into cells; at the centre of each is a base station, which has been assigned a set of radio frequencies for telephone communications and signalling purposes [65,66,71,106]. Frequency reuse can then be achieved through base stations covering different areas using the same carrier frequency, separated from one another by a sufficient physical distance so that co-channel interference is not perceivable. Various cell sizes are used depending upon expected subscriber density, thus rural cells will be larger than urban cells. The disadvantage of the cellular approach is that in heavily populated areas the number of cell handovers from one cell to another may increase during the duration of a call. This has been examined in [71,86].

Frequency reuse is one way of achieving greater spectrum efficiency. The type of coding and modulation schemes adopted will also effect this efficiency. With the cellular and CT2 systems, voice traffic is circuit switched. This dedication of reserved resources without resource sharing is also a cause of spectrum inefficiency. The demand for personal communications set against the limited spectrum allocated for mobile radio gives an impetus to consideration of packet radio. This will provide the user with exactly the capacity required, unlike circuit switching. Thus, call blocking can be a function of the required average capacity rather than the peak capacity. It also seems logical that with the introduction of ATM, calls that are already in a packetised form should not be transmitted over radio channels also in this format. This would greatly simplify the interaction of communications involving Broadband ISDN and cellular radio networks.

The speech protocols developed in chapter 4 will now be examined in a single cell broadcast network. The use of these protocols will enable a greater number of conversations to be supported whilst providing greater flexibility compared to the presently available circuit switched systems. It may be necessary to adapt the protocols used in a LAN, which was assumed to be error-free, in order that their operation is robust to the mobile channel characteristics. To achieve suitable protocol performance, it is first necessary to examine the mobile radio channel characteristics.

5.2 Mobile Radio Channel Characteristics

In mobile radio networks, signal propagation can occur over several paths, such as line-of-sight path and path due to scatterings caused by reflections from structures. As a result of the vehicle movement, the signal arriving at a receiver consists of the sum of randomly delayed versions of the transmitted signal due to this multipath propagation, resulting in fading, along with any encountered path loss. The received signal shows fading consisting of very rapid fluctuations around the mean signal level superimposed on relatively slow variations of the mean level. The mobile radio channel then, can be considered to be a time-varying channel due to the following parameters [42,65]:

- Path loss, where the received signal power is inversely proportional to $(\text{distance})^i$, where $3<i<4$.

- Short-term variations, as a result of the multipath propagation of the signal in which nulling or reinforcement of the direct path results. With the rapid fluctuations, the signal amplitude distribution can be approximated by a Rayleigh distribution while the phase of the received signal is uniformly distributed.

- Long-term variations. These are due to the terrain topography such as hills and man-made obstacles etc. The mean value of the received signal follows a log-normal distribution.

- Noise impulses, principally due to vehicular ignition noise.

- Doppler effect due to vehicle motion. Instability of the received signal frequency can occur due to Doppler shift in the carrier frequency.

As a result of the above phenomena, RF connectivity is difficult to predict and may abruptly change as mobile stations move. Radio networks should be able to support mobile stations at normal vehicular ground speeds within the area of coverage with full connectivity. Field measurements have shown that the major source of errors over mobile channels is due to fading [101]. Throughout this chapter it is assumed that any impulse noise is corrected by some form of error correction within the packet [102] and the effects due to shadowing and Doppler shift are ignored. Thus the model assumes that transmission errors are caused only by short-term fading due to the multipath propagation of the signal, these being the prominent source of errors. The probability of a transmission error based upon these assumptions is examined in 5.4.

The lowest and highest frequencies which can be used for a packet radio system are determined primarily by considerations of bandwidth and propagation loss respectively. Practical radio systems should use radio frequencies in the upper VHF band, in the UHF band from 300MHz to 3GHz, and in the lower portion of the SHF band from 3GHz to 10GHz. Propagation in the HF band can provide long distance communication due to sky wave reflection from the earth`s ionosphere, but the propagation suffers from multipath spreading of the signal which limits the data rate that can be used. Multipath spreading in the VHF band, using line-of-sight propagation, is typically a few microseconds as compared to the millisecond spreading encountered at HF. Multipath fading and distortion can still be a problem at VHF for mobile stations. Diversity and spread spectrum techniques, however, can overcome these problems.

The delay spread due to the multipath scattering causes symbol distortion and intersymbol interference. This intersymbol interference can increase the symbol error probability, which cannot be improved by increasing the signal-to-noise ratio on the channel. This then requires that the signalling rate be slowed to a period of less than the reciprocal of the delay spread. From the delay spreads given in [65], a maximum signalling rate of 300Kbps has been assumed in this chapter within the frequency range 800MHz to 900MHz.

5.3 Data Transmission Using Radio Channels

Investigation of data transmission over radio channels has been predominantly limited to the study of stationary packet radio networks. The ALOHA system was one of the first such systems [55,109]. Due to its poor throughput performance the CSMA protocol was developed [55,110]. The performance of the CSMA protocol is dependent on the ability of each station to sense the carrier of any other transmission on the channel. However, in the radio environment, some stations are *hidden* from each other because they are out-of-sight or out-of-range. The protocol described in [107] overcomes this hidden terminal problem by the application of Busy-Tone Multiple-Access (BTMA). A problem still exists with this though, if two stations transmit simultaneously. They will both sense the busy-tone signal from the base station and so believe they have captured the channel and continue with their transmissions, ignorant of the collision. An improvement to BTMA is achieved through Rude-CSMA [90]. Similar busy-tone schemes were developed in [87,88].

The integration of voice and data over mobile radio channels has been considered in [72]. The system uses FM-based channels for the digitised speech and interleaves data packets in the voice silent periods. The main source of degradation in this system is the delay a voice station experiences in finding a free channel at the beginning of each talkspurt. Stations attempt to find a free channel by sensing busy-tones on each channel. If a free channel has not been found after a certain number of attempts, the talkspurt is lost. The main disadvantages with this system is that voice loss may include whole talkspurts, channel capacity is wasted due to guard bands, packet delay compared to a TDM approach will be larger, and the mobile and base stations will be relatively complex due the different tones and filters necessary for each channel.

Local area radio networks for voice/data integration are described in [8,17,32,36,119,126]. In local radio networks, the round trip propagation time between terminals and base stations is of the order of a few tens of microseconds outdoors, and less than one microsecond indoors. Voice stations in [36] contend for a time slot using slotted ALOHA. Once successful, they reserve a reservation slot for the remainder of the talkspurt. A similar mechanism is described in [105], except that free slots are subdivided into minislots, so as to provide several chances of a successful reservation within the free slot. The system in [126] uses a framed-polling scheme as the common channel access technique. Such an approach however, in an outdoors environment, is inefficient due to the long propagation times experienced [108].

To gain the advantages of mobility, it is necessary that data transmitted over the radio channel utilises error-control techniques to constrain errors within the information. Typical error-control schemes for use in mobile radio are now described.

5.3.1 Error-Control Techniques In Mobile Radio

Due to the characteristics of the radio channel, some form of error-control scheme is required. The effect of transmission errors depends to some extent on the type of transmitted data and may cause severe effects on the received information. Digital speech contains large amounts of redundancy and so the occasional error may have little subjective effect on the received signal. However, the integrity of the voice packet header and data packets from file transfers etc. are very important.

Error-control techniques enable a transmission system to approach reliable performance, despite the presence of noise. With these techniques, some additional redundancy through parity digits are transmitted with the information bits over the channel. There are two different techniques that can be used for error control, error detection with retransmission [69,79], and forward error correction (FEC) [69,102]. Both techniques add redundancy to data information before transmission in order to reduce the effects of errors that occur over the channel.

Error detection with retransmission utilises parity bits to detect if an error has occurred. The receiver station does not attempt to correct the error, it simply requests the transmitter to retransmit the data. It is thus necessary for the receiver to alert the transmitter if an error has occurred; this is achieved by automatic repeat request (ARQ). There are three forms of ARQ [69]:

- ***Stop-and-wait ARQ***; where the transmitter waits for an acknowledgement of each transmission before it proceeds with the next transmission.

- ***Go-back-N continuous ARQ***; here both terminals are transmitting simultaneously, the transmitter sending message data and the receiver sending acknowledgements. A sequence number is assigned to each block of data. Whenever the receiver detects a block in error, it transmits a negative acknowledgement, and the transmitter goes back N blocks to retransmit all blocks starting from the block in error.

- ***Selective-repeat continuous ARQ***; this is the same as go-back-N ARQ, except that only the corrupted block is retransmitted.

In any packet communication system where some degree of reliability across channels is required, ARQ mechanisms are typically used to notify a station of its success in the transmission of a packet. Although they are simple to implement, ARQ schemes have the disadvantage that as the error rate increases, the time delay of supplying correct information to the receiver also increases.

FEC codes enable receivers to attempt to correct transmission errors. If the receiver fails to correct the errors, the received data block will be incorrectly decoded and erroneous data delivered to the user. Since no retransmission is required with FEC, no feedback channel is necessary for acknowledgements and the throughput efficiency of the system is maintained at a constant rate, regardless of the channel error rate. However, to achieve this error correction, a large amount of redundancy is usually required. Another problem is that it is hard to achieve high system reliability as the channel error-rates increase.

The error-correcting codes protect digital information by inserting redundant bits into the stream of information bits. The redundant bits are calculated from the information bit stream; this relationship between the two sets of bits is used in the decoder to detect and correct errors [79,102]. In a block coding scheme, the current set of input bits are processed independently in the encoder. A convolution code, alternatively, is obtained from the current input set, and the h-1 previous input sets stored in the encoder. The quantity h is called the constraint length of the convolution code. The type of error correction code used is dependent upon the form error patterns take. Mobile radio channels are characterised by having bursty error patterns caused by fading due to multipath propagation [101]. For mobile channels, there are three main types of burst error correction:

- ***Burst error correcting codes.*** If information relating to a single digit is spread more widely in the digit stream, then large bursts can be corrected if longer error-free periods exist between bursts [68]. The variation of fade durations in mobile radio can be unpredictable in length and so the use of these codes is restricted.

- ***Interleaving.*** This process separates codeword symbols in time and the intervening times are similarly filled by symbols from other code words. Separating the symbols in time effectively transforms a burst of errors into random errors within a codeword, thereby enabling random error correcting codes to be used [102]. The usual permutation of a transmission block is accomplished by filling the columns of an M-row N-column array with the encoded sequence. The array is then transmitted one row at a time. For use with single error correcting codes the interleaver parameters are selected such that the number of columns N exceeds the expected burst length. For block codes, M should be larger than the code block length, while for convolution codes, M should be larger than the constraint length. Thus the main disadvantage of interleaved codes is that they cannot be used if the message length is small.

- ***Repeated codeword transmission with random error correction.*** This technique is applied in the AMPS network [23], whereby each message is transmitted five times. A bit-by-bit 3-out-of-5 vote then determines the message, with one bit error correction if necessary.

The previous sections have reviewed data transmission over mobile radio channels. The channel characteristics require that appropriate error control techniques are used to restrain the effects of errors on the transmitted information. The basic network topology that is used in conjunction with the speech protocols of chapter 4 is now described.

5.4 Idle-Signal Casting Multiple Access With Collision Detection

To overcome the hidden terminal problem, Idle-Signal Casting Multiple Access with Collision Detection (ICMA/CD) has been used. The topology is based upon a population of mobile stations communicating with a central base station over a packet switched multiple access radio channel. The ICMA/CD approach splits the available capacity into two, a mobile-to-base station upward-channel (U-channel) and a base station-to-mobile downward-channel (D-channel). Although mobile terminals may be out of range with each other to detect a transmission, it is assumed that all mobiles can sense the transmissions from the base station on the D-channel. The basic network topology is illustrated in figure 5.1. The base station broadcasts idle-signals on the D-channel if no mobile terminals are transmitting on the U-channel. After the detection of an upward packet, the base station converts the idle-signals into busy-signals to inhibit other terminals from transmitting. It is assumed that each idle/busy-signal is so brief that mobile terminals can obtain immediately the idle/busy state of the U-channel. The base station continues to transmit busy-signals on the D-channel until the U-channel transmission is complete. The repeated sequential transmission of idle/busy-signals compensates for any loss of fading on the D-channel and can also provide a form of synchronisation for mobile transmissions. By receiving a preamble signal at the beginning of an upward channel transmission, the base station detects the start of a packet transmission and synchronises bit timing.

However, collisions can still occur due to the propagation delay between the time a terminal started its packet transmission, until the time when busy-signals are detected by the other terminals. If a transmission is initiated in this region, all packet transmissions are considered corrupted. The base station can detect packet errors through the use of a cyclic redundancy check (CRC). If the base station detects the CRC of a packet in error due to either a collision or fading, it transmits stop-signals on the D-channel. A speech packet is considered lost if it is received with an error in the packet header otherwise, the packet is accepted by the base station. Errors in the voice portion of the packet are not detected due to the redundancy in the speech. However, such errors tend to degrade the quality of the reconstructed speech. The frame format of a voice packet is illustrated in figure 5.2.

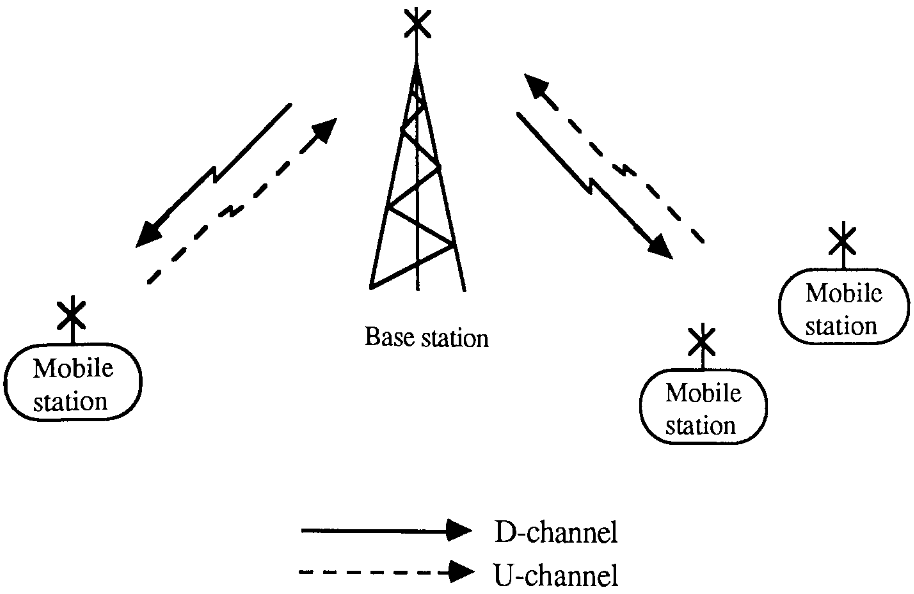


Figure 5.1 ICMA/CD network topology

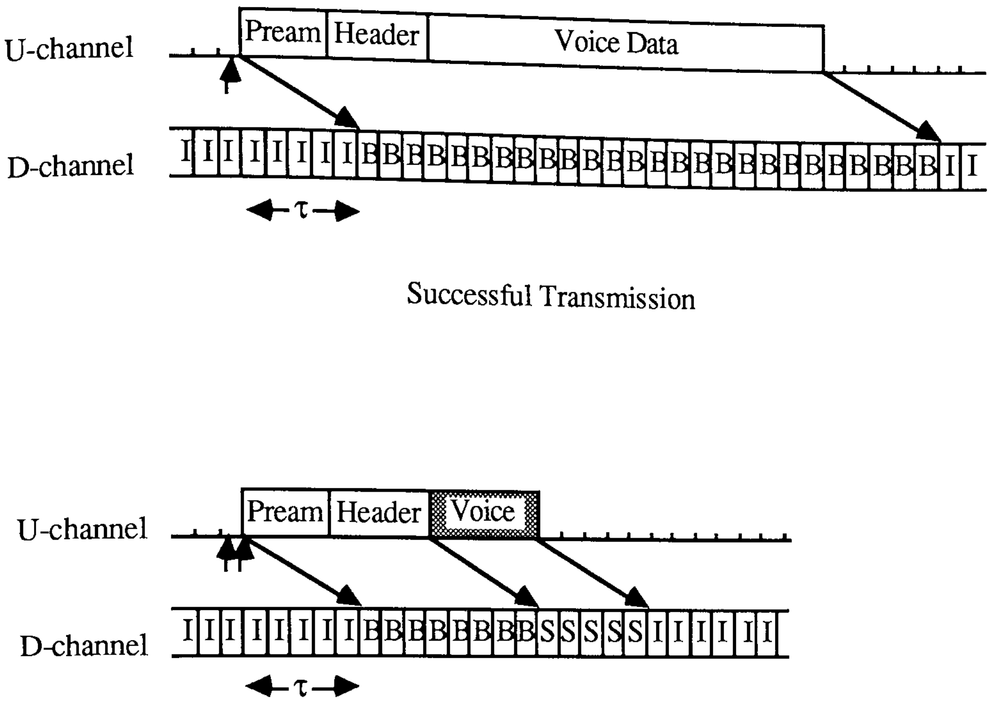
Preamble	Flag	Address	Control	CRC	Voice Data	Flag

Figure 5.2 Voice frame format

A voice packet is considered lost if either fading or a collision corrupts the preamble or packet header. Since the CRC is only computed against the voice packet header, the base station can determine quickly if an error has occurred in order to transmit stop-signals and limit the amount of wasted channel time. This then overcomes the problem of BTMA described in 5.3. Successful and Unsuccessful transmissions of voice packets using ICMA/CD are illustrated in figure 5.3.

5.4.1 Throughput Analysis Of Slotted ICMA/CD With Fast Fading

Data that has inherent redundancy has the advantage that errors may be tolerated within the information field. The throughput of such data on the U-channel is now examined for slotted non-persistent CSMA/CD using the idle-casting topology. The basic assumptions are as follows:



Unsuccessful Transmission

I: Idle signal
B: Busy signal
S: Stop signal
τ: Transmission detection time

Figure 5.3 Successful and unsuccessful transmissions with ICMA/CD

- The number of stations is infinite and the arrival process from this infinite population is Poisson distributed; the process includes both new and previously collided packets. The average channel traffic is equal to G packets per T seconds.
- The maximum propagation delay from mobile-to-mobile via the central base station is τ seconds.
- τ seconds constitutes a slot. Stations may only transmit at the beginning of a slot.
- At any one point in time, each station has at most one packet ready for transmission, including previously collided packets.
- A packet transmission time has a duration of T seconds. T is equal to an integer number of slots.
- Carrier sensing by both the base station and mobiles takes place instantaneously. The recognition of idle, busy or stop-signals is also instantaneous.

- Fast fading is assumed to be the only source of noise on the U-channel. If a fade occurs within the packet header, the packet is considered corrupted. The D-channel is considered to be noiseless.
- The overlap of any fraction of two or more packets results in corruption, so that involved packets must be retransmitted.

The activity on the channel may be divided into busy periods during which transmission attempts are made, and idle periods during which no station transmits. The busy period is followed by an idle period. This period of time is defined as a cycle. To find the channel throughput S, we let E[U] be the average time during the cycle that the channel is used without collisions or fading in the packet header, E[B] to be the expected duration of the busy period, and E[I] the expected length of the idle period. The throughput based on renewal theory is then given by

$$S = \frac{E[U]}{E[B] + E[I]} \qquad \ldots(5.1)$$

E[U], the average time during a cycle that a transmission is successful is given by

$$E[U] = T\,P_s\,(1-P_e) \qquad \ldots(5.2)$$

where P_s is the conditional probability that if a transmission has occurred during a slot, there is only one, and P_e is the probability of a fade occurring in the packet header. P_s can be written as

$$P_s = P\{\text{one packet arrives in a slot } \tau \mid \text{some arrival occurs}\}$$

$$= P\{\text{one packet arrives in a slot } \tau\} \,/\, P\{\text{some arrival occurs}\}$$

Using Poisson arrival statistics we have

$$P\{\text{one packet arrives in a slot } \tau\} = G(\tau/T)e^{-G(\tau/T)} = aGe^{-aG} \qquad \ldots(5.3)$$

where a = propagation time/packet transmission time ratio.

$$P\{\text{some arrival occurs}\} = 1 - e^{-aG} \qquad \ldots(5.4)$$

Therefore P_s is given by

$$P_s = aGe^{-aG}/(1-e^{-aG})$$

Combining with (5.2) gives

$$E[U] = T(1-P_e)\, aGe^{-aG}/(1-e^{-aG}) \quad ...(5.5)$$

Consider now E[B], the average length of the channel busy period. Since a busy period can be either a successful transmission period or a contention/fading period, E[B] is given by

$$E[B] = (T+\tau)\, P_s\,(1-P_e) + \zeta\,(1-P_s+P_sP_e) \quad ...(5.6)$$

where ζ is the channel time wasted due to a collision and or fading. It takes $\tau/2$ seconds for a transmission to reach the base station, a further H seconds for the header to be transmitted and so for the base station to detect an error, and a further $\tau/2$ to notify the mobile station of any error. It is assumed that mobiles instantaneously detect stop, busy and idle-signals on the D-channel. Therefore, $\zeta=\tau+H$.

An idle period always consists of an integral number of time slots. If a packet arrives during the last time slot of a busy period, then the next slot immediately starts a busy period, so the idle period is zero. To find E[I], first consider the case I=0. The probability, p, of this occurring is merely the probability of some packet arriving in the interval τ, which is given by (5.4)

$$P\{I=0\} = p = 1 - e^{-aG} \quad ...(5.7)$$

If we have an idle period of length I=i slots, the probability for no arrivals in i consecutive slots followed by some arrival in the next slot is

$$P\{I=i \text{ slots}\} = (1-p)^i p \quad ...(5.8)$$

This describes a geometrically distributed random variable J with a mean value of

$$E[J] = \sum_{i=0}^{\infty} i(1-p)^i p = \frac{1-p}{p} \quad ...(5.9)$$

The average length of the idle period is then τ E[J], thus using (5.7) and (5.9), E[I] is given by

$$E[I] = \frac{\tau e^{-aG}}{1 - e^{-aG}} \quad ...(5.10)$$

Collecting the results for E[U], E[B] and E[I] and using them in (5.1) gives the following throughput

$$S = \frac{aGe^{-aG}(1-P_e)}{a + (1-P_e)\, e^{-aG} + h\,(1 - e^{-aG} - aGe^{-aG} + P_e aGe^{-aG})} \qquad ...(5.11)$$

where h=H/T. An error occurs during the packet header when the signal fades below the receiver threshold level. The effects of interference, signal distortion and other forms of noise such as ignition and thermal are assumed to be negligible compared to fast fading. $1\text{-}P_e$ is the probability that the whole header was contained in a non-fade interval. Let,

t_f = fade interval, with a mean value T_f. The envelope of the fade interval is assumed Rayleigh distributed.
t_n = non-fade interval, with a mean value T_n. The non-fade interval can be approximated to be exponentially distributed.
t_i = inter-fade interval, with mean value T_i.

The density function of t_n can be written as,

$$f_{t_n}(t) = \frac{1}{T_n} e^{-t/T_n}, \qquad t>0 \qquad ...(5.12)$$

The probability that a packet is received correctly, if there is no contention, is given by the probability that the non-fade interval t_n will exceed a period longer than the packet header transmission time. Therefore, if H is the packet header transmission time, then

$$\begin{aligned} 1 - P_e &= \frac{T_n}{T_i} P[t_n > H] \\ &= \frac{T_n}{T_i} (1 - P[t_n \le H]) \\ &= \frac{T_n}{T_i} e^{-H/T_n} \qquad ...(5.13) \end{aligned}$$

The average time the channel stays in a fade state T_f, is given by [48,65]:

$$T_f = \frac{e^{\rho} - 1}{f_d \sqrt{2\pi\rho}} \qquad ...(5.14)$$

where ρ is defined as the ratio of the threshold power level to the RMS signal power level, and $f_d = V/\lambda_c$ is the Doppler frequency, where V is the vehicle speed, and λ_c the carrier wavelength. The average number of fades per second is given by:

$$N_f = f_d \sqrt{2\pi\rho}\, e^{-\rho} \qquad ...(5.15)$$

Thus, the average non-fade duration is:

$$T_n = (1-N_f T_f)/N_f = \frac{1}{f_d \sqrt{2\pi\rho}} \qquad ...(5.16)$$

Hence T_i, the mean inter-fade interval is simply $T_f + T_n$ which is given by:

$$T_i = \frac{e^{\rho}}{f_d \sqrt{2\pi\rho}} \qquad ...(5.17)$$

Therefore substituting (5.16) and (5.17) into (5.13) yields the probability of a packet header transmission time occurring totally in a non-fade interval,

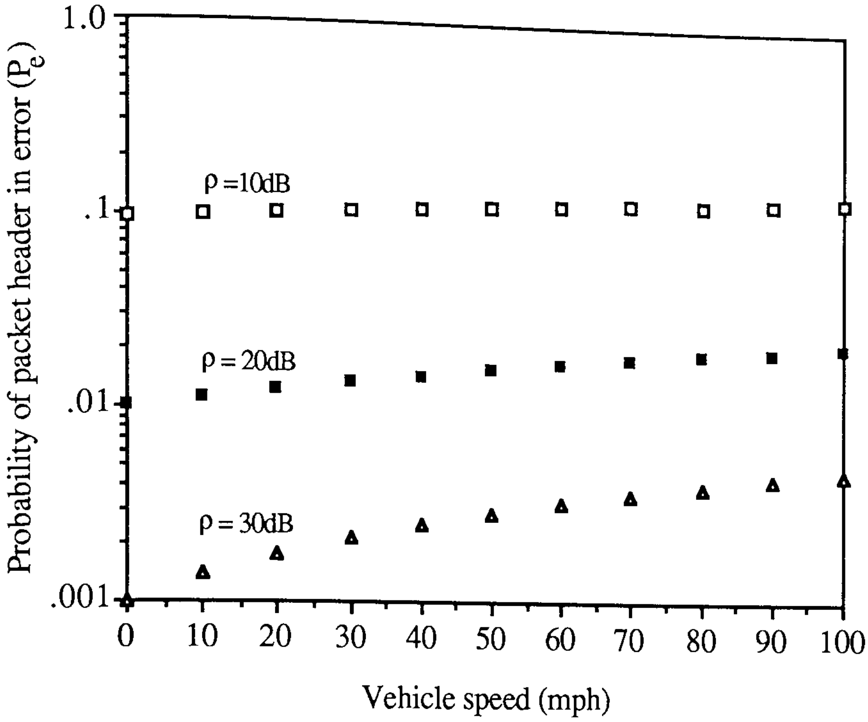
$$1 - P_e = e^{-(\rho + f_d\sqrt{2\pi\rho}H)} \qquad ...(5.18)$$

Substituting (5.18) into (5.11) therefore gives the throughput of synchronous non-persistent ICMA/CD in a fast fading environment for information that can tolerate errors in the data field due to fading. The effect of fading on the packet header is illustrated in figure 5.4. Obviously, as ρ (in dB) increases, i.e. the rms signal power increases, the error probability decreases. At these higher values of ρ, it can be seen more clearly that the error rate degrades as the vehicle speed increases.

5.5 AR-VT-CSMA/CD In The Radio Environment

The available bandwidth is divided into three channels:

- ***Upward-channel*** (U-channel); mobile stations transmit their packets towards the base station using this channel.
- ***Downward-channel*** (D-channel); the base station simultaneously transmits packets it is receiving on the U-channel down on the D-channel.
- ***Acknowledgement-channel*** (A-channel); Once a receiving mobile has sensed a packet transmission towards it on the D-channel, it notifies the base station of this event by transmitting an acknowledgement mini-packet on the A-channel.



Carrier frequency = 800MHz, channel capacity = 300Kbps, packet header = 120 bits.

Figure 5.4 The probability of the packet header in error due to fading, at various vehicle speeds.

The basic operation of the AR-VT-CSMA/CD protocol is the same as in the LAN environment, except that mobiles cannot detect collisions. Errors resulting either from collisions or due to fading are detected through error control at the base station. The protocol operation for a transmitting mobile, the base station and a receiving mobile are described in figures 5.5, 5.6 and 5.7 respectively.

It is assumed that mobiles can only commence transmissions at the beginning of a slot, when they sense idle-signals on the D-channel from the base station. The synchronisation for slots could be achieved either through the idle-signals themselves or through some out-of-band signalling. When the base station has detected the preamble at the beginning of a mobile transmission, it withdraws the idle-signals from the D-channel and commences transmitting the frame it is receiving on the U-channel down onto the D-channel. This then defines the slot length as the maximum propagation time from a mobile to the base station and back (τ), plus the time for the base station to receive the preamble, T_{pre}.

Error correction is only performed on the voice packet header. The transmission from the mobile towards the base station could have experienced a collision and/or a fade. If the U and D-channels have a capacity of 300Kbps each, the transmission time of a 120 bit header is 400μs. If the worst case is assumed (ρ=10dB, V=20mph) and for a carrier frequency of 800MHz, from equation (5.14) the mean length of a fade is 5.57ms If the probability of error in a fade is 0.5, then it can be seen that without very low coding efficiency, burst error codes, interleaving, or repeated sequential transmissions will be ineffective in correcting errors due to a fade, since the mean fade length is many times longer than the packet header transmission time. Consequently, only error detection is employed in the packet header.

If the base station detects an error in the packet header, it ceases transmitting the packet on the D-channel and instead transmits stop-signals until no transmission is sensed on the U-channel. When a transmitting mobile detects the stop-signals on the D-channel, its course of action depends on whether it is established or not, i.e. if it has been previously successful and implicitly reserved the virtual time slot or not. If it is established, it retransmits immediately after the stop-signals have ceased, believing its packet was in error due to a collision. If stop-signals are again sensed, it assumes that it is in a fade and randomly reschedules its transmission time to a future virtual time and so is no longer classified as an established station. All other stations automatically randomly reschedule their transmission times. The base station continues to transmit stop-signals on the D-channel until it has sensed that all transmissions have ceased. It then delays one slot before transmitting idle-signals on the D-channel, to enable any established mobiles to transmit.

If the base station received the header correctly, it assumes that the accompanying voice information is good. However, there is still the possibility that the packet may experience fading on the D-channel, towards the receiving mobile. To notify the base station of a successful packet header reception, the destination mobile transmits an acknowledgement towards the base station on the A-channel. If the base station has not received this acknowledgement after the timeout interval $\tau + T_a$ from when it finished transmitting the header on the D-channel, where T_a is the transmission time of an acknowledgement on the A-channel, it ceases the packet transmission on the D-channel and transmits stop-signals. Timing diagrams of successful and unsuccessful transmissions are illustrated in figure 5.8.

```
while in talkspurt do
   begin
      while packetisation interval < T_min do
         assemble packet;
      repeat
         begin
            wait until virtual time clock ≥ arrival time;                         {5.5b}
            commence packet transmission on U-channel (all voice bits up to P_max);
            while transmission is incomplete do
               begin
                  if stop-signals are detected on D-channel then
                     if established station then
                        begin
                           cease transmission;
                           mobile assigned unestablished;
                           commence packet retransmission when stop-signals
                           cease;
                        end
                     else
                        begin
                           cease transmission;
                           compute backoff;
                        end
                  else                                         {successful transmission}
                     complete successful transmission;
               end
         end
      until successful transmission;
   end
```

Figure 5.5a Mobile station operation of the AR-VT-CSMA/CD protocol - packet transmission

```
while virtual time < arrival time do
   begin
      while idle-signals are not sensed on D-channel do
         begin
            stop virtual time clock;
            wait τ+T_pre seconds;
         end
      virtual time = min{virtual time + η(τ+T_pre), real time};
      if virtual time < arrival time then
         wait τ+T_pre seconds;
      if packet length has exceeded P_max then
         discard oldest voice bits;

   end                                            {packet ready for transmission}
```

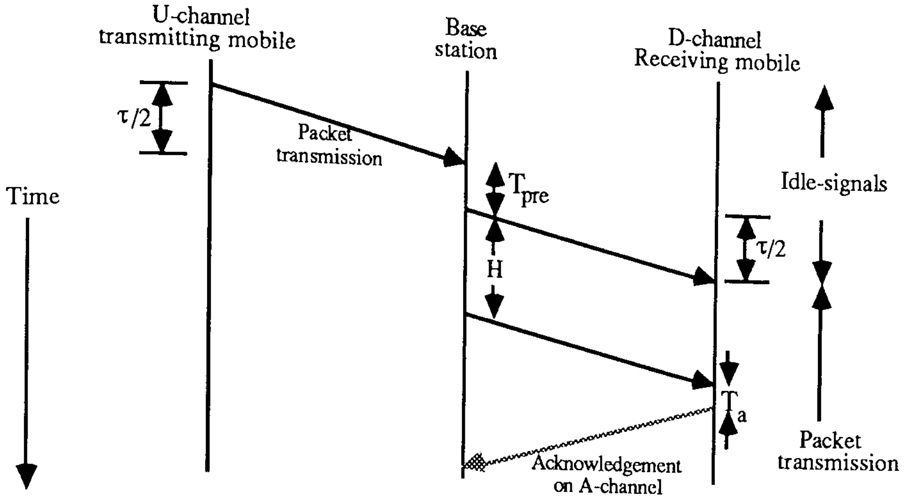
Figure 5.5b *Wait until virtual time clock ≥ arrival time* for mobile station.

```
begin
    while no transmission sensed on U-channel do
        transmit idle-signals on D-channel;
    withdraw idle-signals and initiate packet transmission on D-channel;
    while transmission detected on U-channel is incomplete do
        begin
            if error detected in packet header then
                begin
                    withdraw transmission on D-channel;
                    repeat
                        transmit stop-signals on D-channel;
                    until no transmission detected on U-channel
                    wait τ+T_pre seconds;
                end
                if no acknowledgement received on A-channel after timeout τ+T_a from
                when packet header was transmitted then
                    begin
                        withdraw transmission on D-channel;
                        repeat
                            transmit stop-signals on D-channel;
                        until no transmission detected on U-channel
                        wait τ+T_pre seconds;
                    end
                else
                    continue packet transmission on D-channel;
        end
end
```

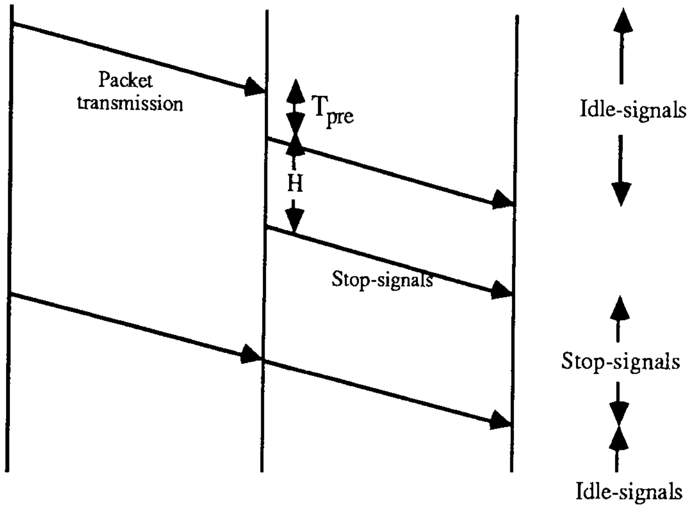
Figure 5.6 Base station operation of the AR-VT-CSMA/CD protocol.

```
begin
    listen to D-channel;
    if packet transmitted on D-channel addressed to mobile j is received correctly then
        transmit acknowledgement packet on A-channel;
end
```

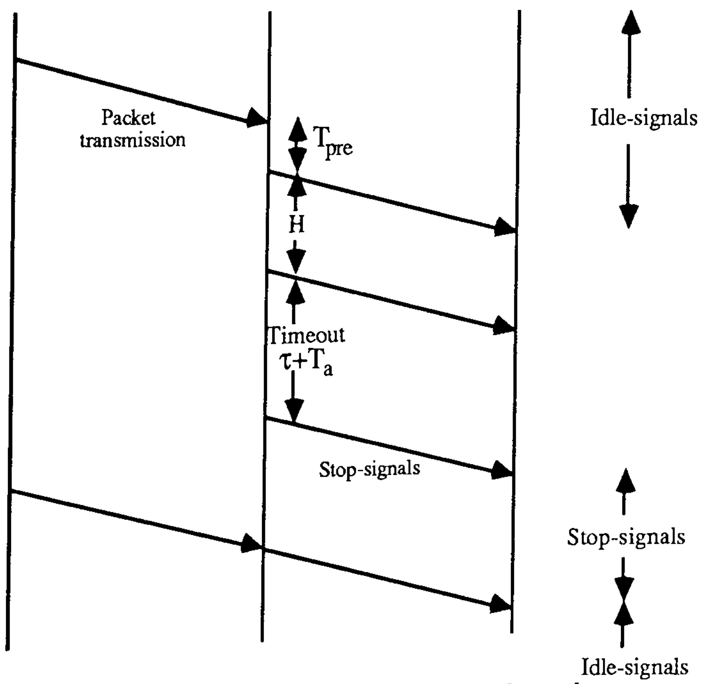
Figure 5.7 Mobile station operation of the AR-VT-CSMA/CD protocol - packet reception.



(a) Successful transmission



(b) Unsuccessful transmission due to fade/collision on U-channel.



(c) Unsuccessful transmission due to fade on the D-channel.

Figure 5.8 Packet transmissions using AR-VT-CSMA/CD in the radio environment.

5.5.1 Simulation Results Of AR-VT-CSMA/CD In The Radio Environment

The synchronous version AR-VT-CSMA/CD protocol was simulated in a mobile radio environment. It has been assumed that the only cause of error, other than collisions, was due to fast fading. The probability of error in a fade was considered for the worst possible case, of 0.5. The fading model is described in chapter 6. The capacity of the U-channel and D-channels were considered the same and equal to 300Kbps each, with carrier frequencies of 800MHz and 802MHz respectively. The A-channel capacity was assumed to be 50Kbps with a carrier frequency of 804MHz. The maximum propagation delay from a mobile to the base station was 1.5μs, which is equivalent to a radius of 450m. Before the base station removed the idle-signals it was first necessary to detect the preamble. The preamble had a length of 16bits, which gives a slot time, defined as the time from when a mobile starts transmitting until when it detects the absence of idle signals, as 32μs + 2x1.5μs = 35μs. Speech was encoded at a rate of 16Kbps, with packet headers of length 120bits. A time constraint of 30ms has been applied to voice packets; to use shorter length packets would be very inefficient with such a long slot duration.

The effects of capture have been ignored, and so if one or more transmissions overlapped, all transmissions were assumed to be in error. This assumption obviously puts a lower bound on the packet throughput. If a packet header is received correctly, any bit loss thereafter in the voice information is ignored.

Initially all stations were assumed to maintain their own virtual and real time clocks. Figure 5.9 illustrated this case with ρ=20dB. At the beginning of each slot, if a mobile senses idle-signalling on the D-channel, it knows there are no transmissions at present on the U-channel and so updates its virtual time clock. However, if the mobile does not detect the idle-signalling due to a fade, its virtual time clock is not updated. Consequently, when the virtual time clock of an established station eventually reaches its packet`s arrival time, this transmission time may now collide with the transmission from another established mobile. When this occurs, they must both reschedule to find a new free reservation slot. This breakdown in the implicit reservation mechanism of AR-VT-CSMA/CD is due to misalignment in the virtual time clocks at mobile stations.

This problem can be overcome by the base station appending its virtual time lag of the real time clock to the end of each packet transmission on the D-channel. Mobile stations still update their virtual time clocks when idle-signals are sensed, but when a value is received appended to a packet transmission, the mobile then assumes this new value. This should reduce, although not totally alleviate, the problem of misaligned reservation slots. The improved performance of this system is also illustrated in figure 5.9. If the virtual time clock lag exceeds the maximum time constraint, significant bit loss occurs. Therefore, if we assume that the maximum virtual time clock lag is equal to this time constraint under normal working conditions, then the relationship between the number of bits required, x, the value of η, the slot duration, and the maximum time constraint is

$$2^x = T_{max} / \{\eta(\tau+T_{pre})\}$$

If T_{max}=30ms and η=30, then the number of bits required is $\log(T_{max}/\eta(\tau+T_{pre}))/\log 2$, which equals 5 bits. The frame structure for voice using this scheme is illustrated in figure 5.10. This method also has the advantage that if a mobile becomes active, or if a cellular system is being used and a new mobile enters the cell, the new mobile can determine what is the correct value of the virtual time clock lag of this particular cell.

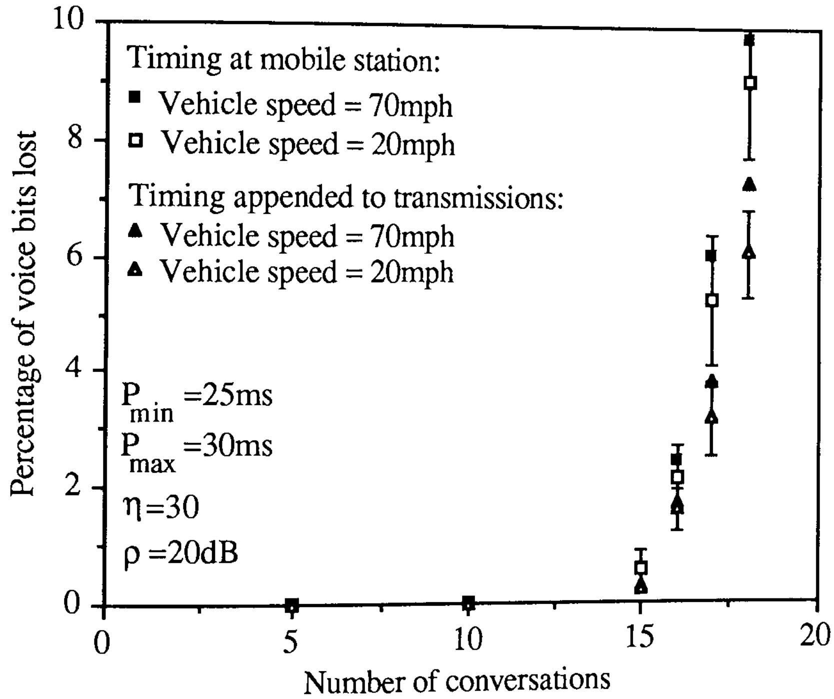


Figure 5.9 Percentage of voice lost at various loads, with mobile stations maintaining virtual time and with virtual time lag appended onto packet transmissions.

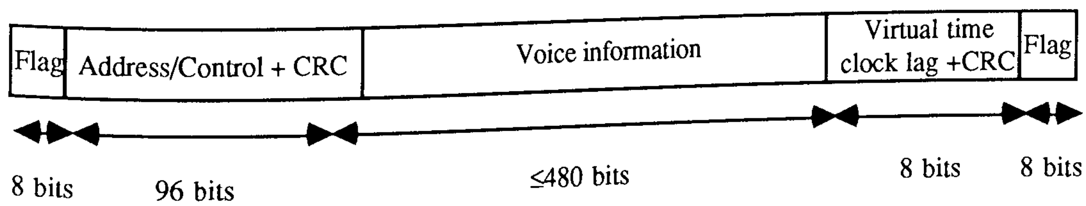


Figure 5.10 Frame format for voice, with the virtual time lag appended by the base station.

The introduction of the virtual time lag appended to the packet transmissions on the D-channel reduces the number of collisions between existing mobile stations. This not only gives a superior performance with respect to the amount of voice loss, but also provides more preferable clip statistics, as illustrated in table 5.1. Maximum clips occur whilst a station is attempting to find a free reservation slot. When no virtual time lag is appended to transmissions, established stations can collide with each other frequently, and so effectively acting as stations with a new talkspurt. This is illustrated by the fact that the mean and standard deviation of the clip length is significantly larger than when the virtual time clock is appended to transmissions.

	N_{max}	Clip Length (ms) Mean	Std dev	Max
Timing at mobile.	16	3.59	7.93	122
Timing appended to transmission.	17	2.15	2.00	70

Table 5.1 Clip statistics for AR-VT-CSMA/CD in the mobile radio environment at 2% bit loss.

When the virtual time lag is appended to the transmission on the D-channel, collisions between established mobile stations can still occur, from either a mobile receiving the time lag whilst in a fade, or missing idle-signals in a fade. With a slot time of 35μs, a minimum length packetisation interval of 25ms encoded at 16Kbps and with a 120bit header, the maximum value of η that could be used for the total capacity of 300Kbps is 49. Table 5.2 illustrates the clip statistics for ρ=20dB at the vehicle speeds of 20 and 70mph for two values of η, 30 and 45. It can be seen that, although the higher value of η can support a similar number of conversations, the clip statistics are significantly inferior to that when η=30. The reason for this is that established stations which have collided with other established stations, or have transmitted in a fade find it more difficult to find a free reservation slot with the higher value of η.

ρ=20dB	N_{max}	Clip Length (ms) Mean	Std dev	Max
η=30 V=20mph	17	2.15	2.00	70
η=30 V=70mph	16	2.20	2.31	91
η=45 V=20mph	17	2.49	5.84	173
η=45 V=70mph	16	2.82	8.19	211

Table 5.2 Clip statistics for AR-VT-CSMA/CD in the mobile environment at 2% bit loss, at various values of η.

Figure 5.11 illustrates the protocol, with timing appended to the packet transmissions on the D-channel, for a 10dB SNR at vehicular speeds of 20 and 70mph. Also shown is the performance of the protocol in a non-fading environment. A virtual time clock update speed of 30 has been used in order to limit the subjective effects of speech clipping.

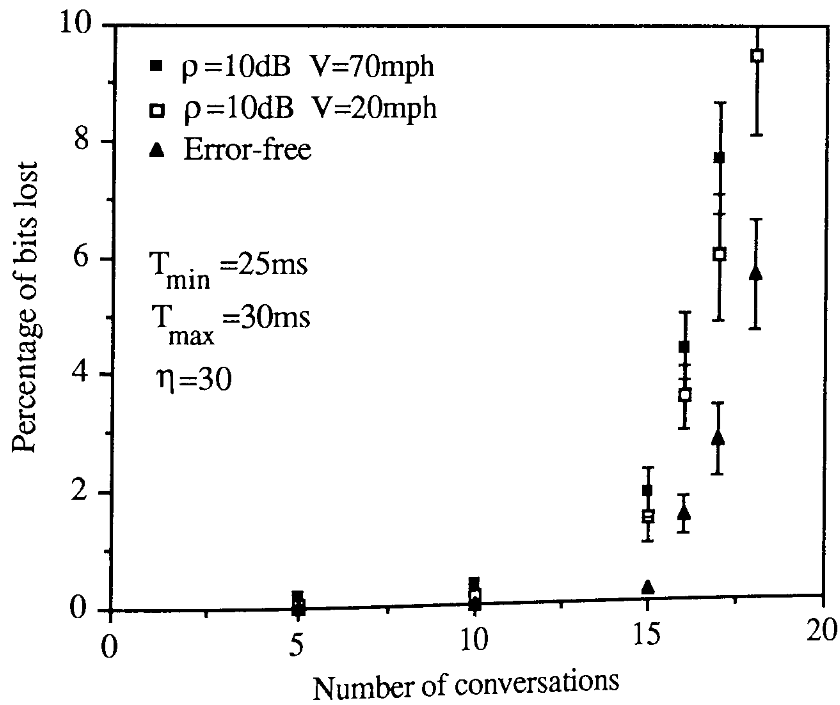


Figure 5.11 Percentage of bits lost at various voice loads using AR-VT-CSMA/CD for an error-free and 10dB SNR environment.

Figure 5.12 illustrates the the percentage of bits lost as a function of the minimum buffer size, P_{min}. A constant virtual time update factor of 20 was used, to enable all values of

P_{min} from 10ms upwards to be used. A fixed voice load of 15 conversations was applied throughout. At low values, the percentage of bits lost starts to increase due to the increased redundancy of the packet header per transmission. As P_{min} approaches P_{max} , the percentage lost again increases, since there is less time that a packet can afford to wait to be transmitted before its maximum buffer length, P_{max} starts to overflow. It can be seen that the optimum value of P_{min}, which minimises bit loss, is smaller for the the lower signal-to-noise ratio. This occurs since, at lower signal-to-noise ratios, voice packet headers are transmitted more frequently in error and thus have to be randomly retransmitted at future times. Therefore, stations in this noisier environment need more time to establish a successful transmission.

Figure 5.13 demonstrates the effect of the protocol with a combined voice and data load. For a constant voice load of 10 conversations, the data load was increased using VT-CSMA/CD, until there was instability in the data packet delay. Since the voice virtual time clock has priority over the data virtual time clock, the percentage of voice bits lost was comparable to that with no data load. Two schemes for error detection in data packets have been examined. The first utilised error detection only at the end of the packet, while the second used error detection both at the end of the packet header and at the end of the packet. If an error was detected at the end of the header, the base station would transmit stop-signals on the D-channel until no transmission was sensed on the U-channel. Error detection in the header provides a form of collision/fade detection. Instability occurred at lower data loads where error detection was was only computed at the end of the packet, since if a transmission was involved in a collision or occurred during a fade, the channel capacity wasted is for an entire packet duration and not just the header transmission time.

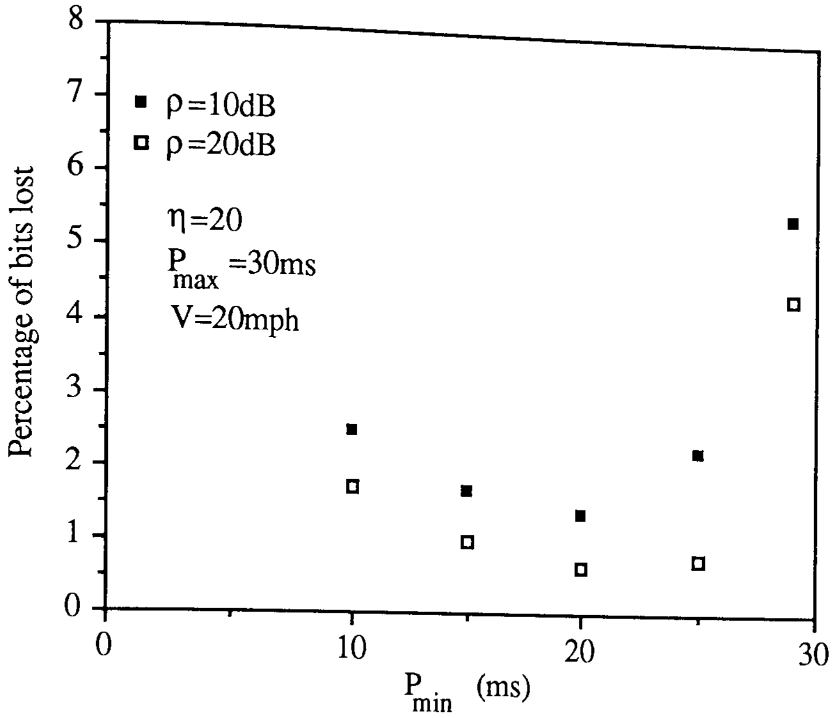


Figure 5.12 Variation of percentage of bits lost versus P_{min} with a constant voice load of 15 conversations.

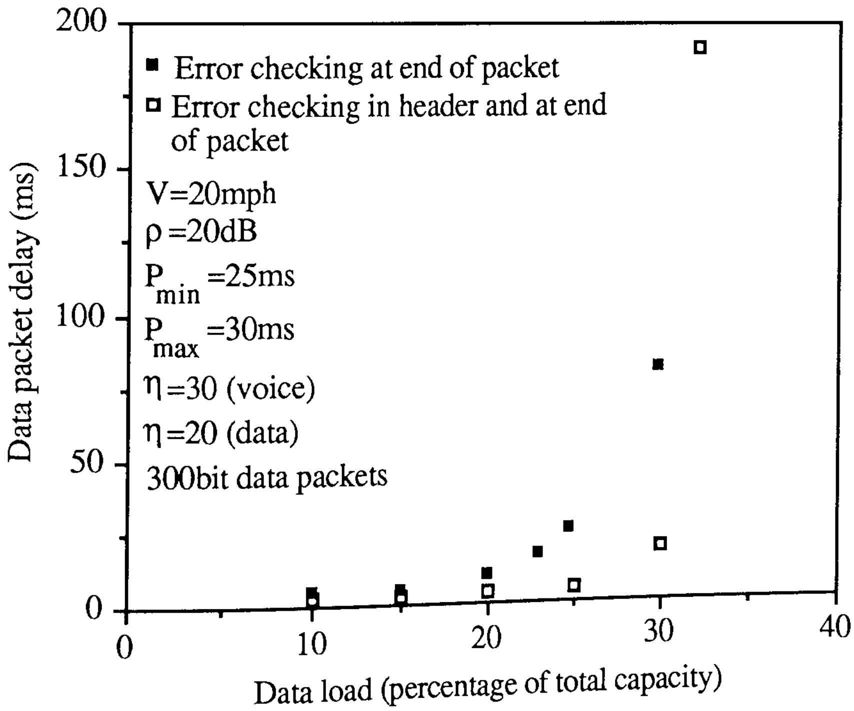


Figure 5.13 Data packet delay at various data loads with a constant voice load of 10 conversations.

5.6 Logical TDMA In The Radio Environment

In the Logical TDMA protocol described in chapter 4, voice stations commencing a new talkspurt joined the global queue through a reservation in the data region. In the reservation scheme developed in this section, both voice and data mobile stations must achieve successful reservations in order to transmit. When such a request is received at the base station, it informs the mobile that its request has joined the distributed queue. In order to prevent collisions between request mini-packets and actual data packets, the channel can either be time-divided or frequency-divided between the two types of data. A frequency-division type protocol has been examined. Frequency-division reservation schemes for data only are examined in [41,108]. In order to obtain benefit from the contention protocol, at the same time eliminating the hidden terminal problem, the access to the channel should be well regulated. This regulation existed in the AR-VT-CSMA/CD protocol in LANs through implicit reservation slots. However with the effect of fading in radio channels, this regulation of virtual time reservation slots deteriorated.

In this section a protocol is proposed in which mobile stations access the channel on a contention basis in order to make a reservation for a transmission. This regulates message packet transmissions and confines the corruption due to collisions between reservation mini-packets on a separate channel. Voice loss and data packet delay are mathematically modelled on error-free channels, and then simulated in the mobile radio environment.

5.6.1 Channel Assignments

The available bandwidth is divided into four channels:

- ***Reservation channel*** (R-channel), which is used by both voice and data mobile stations to gain a transmission reservation in the distributed queue.
- ***Acknowledgement channel*** (A-channel), which is used by the base station to acknowledge to the mobile stations the successful reception of request mini-packets.
- ***Upward channel*** (U-channel) used to transmit conflict-free packet transmissions from the mobile-to-base station.
- ***Downward channel*** (D-channel) used by the base station to relay the U-channel transmissions to the mobile stations.

The process of transmitting reservation mini-packets and attaining a reservation on the global queue is now described.

5.6.2 Reservation Mini-Packets

Mobile stations contend to transmit reservation mini-packets towards the base station on the R-channel in order to gain transmission reservations. The R-channel is assumed to be slotted in time, where one slot equals the maximum propagation time from the mobile stations to the base station, the time for the base station to detect a preamble (T_{pre}), and the maximum propagation time from the base station to the mobile stations, which equals $\tau+T_{pre}$. The transmission of reservation mini-packets can be achieved through ALOHA or CSMA. Non-persistent CSMA has been considered, which is coordinated by idle-signal casting from the base station on the A-channel. If idle-signals are sensed, a mobile can transmit at the beginning of the next slot, otherwise the mobile must randomly reschedule to a future sensing time. Once the base station detects a preamble, it converts the idle-signals into busy-signals on the A-channel.

The base station cannot detect collisions through the superposition of signal strengths, it can only detect if a packet was involved in a collision or a fade at the end of a transmission through the packet`s CRC. It is for this reason that reservation mini-packets are used for data since, if data packets are long, much channel time can be wasted without the collision detection facility. An alternative scheme for data only, using transmission request packets is described in [85]. If the reservation mini-packet has been received correctly at the base station, the base station answers the mobile with an acknowledgement on the A-channel. This is illustrated in figure 5.14. The mobiles request has now been queued at the base station. If a mobile has not received an acknowledgement after a timeout equal to $2\tau+2T_a$, where T_a is the acknowledgement transmission time on the A-channel, from the time where it finished transmitting its reservation mini-packet, then it assumes that the reservation request was unsuccessful and randomly reschedules its next attempt to a future time. This timeout duration will be explained in the next section.

```
while mobile station wants to make a reservation do
   begin
      if idle-signals detected on A-channel then
         begin
            transmit reservation minipacket at beginning of next slot on R-channel;
            if no acknowledgement on A-channel after timeout period 2τ+2Ta then
               wait random amount of time;
            else                                              {reservation successful}
               listen to D-channel for permission to transmit;
         end
      else
         wait random amount of time;
   end
```

Figure 5.14 Reservation process performed by mobile voice/data station.

5.6.3 Packet Transmissions

Once the base station has received successful mini-packets on the R-channel, it stores the reservations on two queues, one for voice and the other for data. The base station divides the time on the D-channel into frames, of lengths equal to the voice packetisation period. These frames are further divided into a voice region and a data region.

In the Logical TDMA protocol described in chapter 4, each station maintained its own queue position by monitoring the number of queued stations. This process is now coordinated by the base station for two main reasons; (i) to enable the protocol to be more robust to errors - it may be difficult for mobiles to monitor the number in the queue and hence their position, and (ii) if a cellular system is in operation, when a mobile crosses a cell boundary, the base stations can exchange the mobile`s identity information and the new base station schedule the mobile`s transmissions. Thus the mobile can transmit in the new cell without either having to contend again, or having to know the number of mobiles already queued in the new cell.

The base station transmits access-tokens on the D-channel containing the address of a mobile station, enabling the particular mobile to transmit on the U-channel. This packet transmission is then relayed simultaneously on the D-channel. The base station then appends an access-token at the end of the packet to authorise which mobile should transmit next. This type of operation was first examined in [30]. Two kinds of reservation exist; (i) the continuity reservation, for messages that comprise of several packets such as voice and file transfers, and (ii) the one-off reservations for single packet messages. Mobiles maintain continuity in their reservations by appending a continuation-token to the end of each transmission, except the final one. When the base station receives the transmission, it strips off the continuity reservation, places the mobile reservation in one of the queues, appends its own access-token, and transmits the packet on the D-channel. This process obviously reduces the contention on the R-channel and so reduces the voice packet delay. The one-off reservations simply do not append continuation-tokens to their transmissions.

The base station schedules voice transmissions, through access-tokens, in the voice region of a frame. Fixed or variable length packets can be used, as described in chapter 4. If there are too many queued voice stations, a round-robin approach is again used to distribute loss fairly. The data queue is served using a FCFS discipline, with transmissions only in the data region if a fixed boundary is used, or also in any remaining time in the voice region if a movable boundary approach is adopted. The process is illustrated in figure 5.15 for an error-free environment.

When the base station has issued an access-token to a particular mobile, there is the possibility that the base station will not receive the mobile`s packet, due to a fade. If the base station has not detected the start of a transmission after a timeout $\tau+T_{pre}$, where T_{pre} is the preamble transmission time, from the time when it finished transmitting the access-token, it assumes the transmitted packet is in a fade. When the system is used without the cellular concept, the system can be considered as one large single cell. In such a case, the transmission on the D-channel provides a mobile with a verification of its transmission, i.e. if a fade is present on the U-channel, it will detect no transmission on the D-channel and thus stop its transmission. This is practical, since the mobile will already be tuned to the D-channel.

Fades can also disrupt transmissions on the D-channel. To overcome wasting channel time in continuing with the transmission, the receiving mobile should acknowledge that the packet header was error-free through an acknowledgement on the A-channel. If no acknowledgement is received after a timeout, the base station ceases the transmission, transmits stop-signals on the D-channel until no transmission is sensed on the U-channel, and then sends the next access-token. The timeout is equal to $\tau+T_a$, from when the base station finished transmitting the header, where T_a is the acknowledgement transmission time on the A-channel.

A mobile only transmits on the A-channel in order to acknowledge a correctly received header. In order to prevent collisions on this channel, the base station does not transmit when it is expecting an acknowledgement. Hence, mobiles expecting an acknowledgement in response to a reservation mini-packet may have to wait $\tau+T_a$ seconds before the base station can acknowledge. A mobile station therefore expecting a reservation acknowledgement will have a timeout limitation of $2\tau+2T_a$.

If the network was cellular, transmissions on the U-channel may not necessarily be transmitted on the D-channel. In this case, it may be difficult for the base station to transmit stop-signals to notify the mobile of the fade if it is already transmitting a packet from another cell on the D-channel. In this situation, it would be necessary to acknowledge the transmitting mobile that its packet header has been received correctly at the destination, through an acknowledgement on the A-channel.

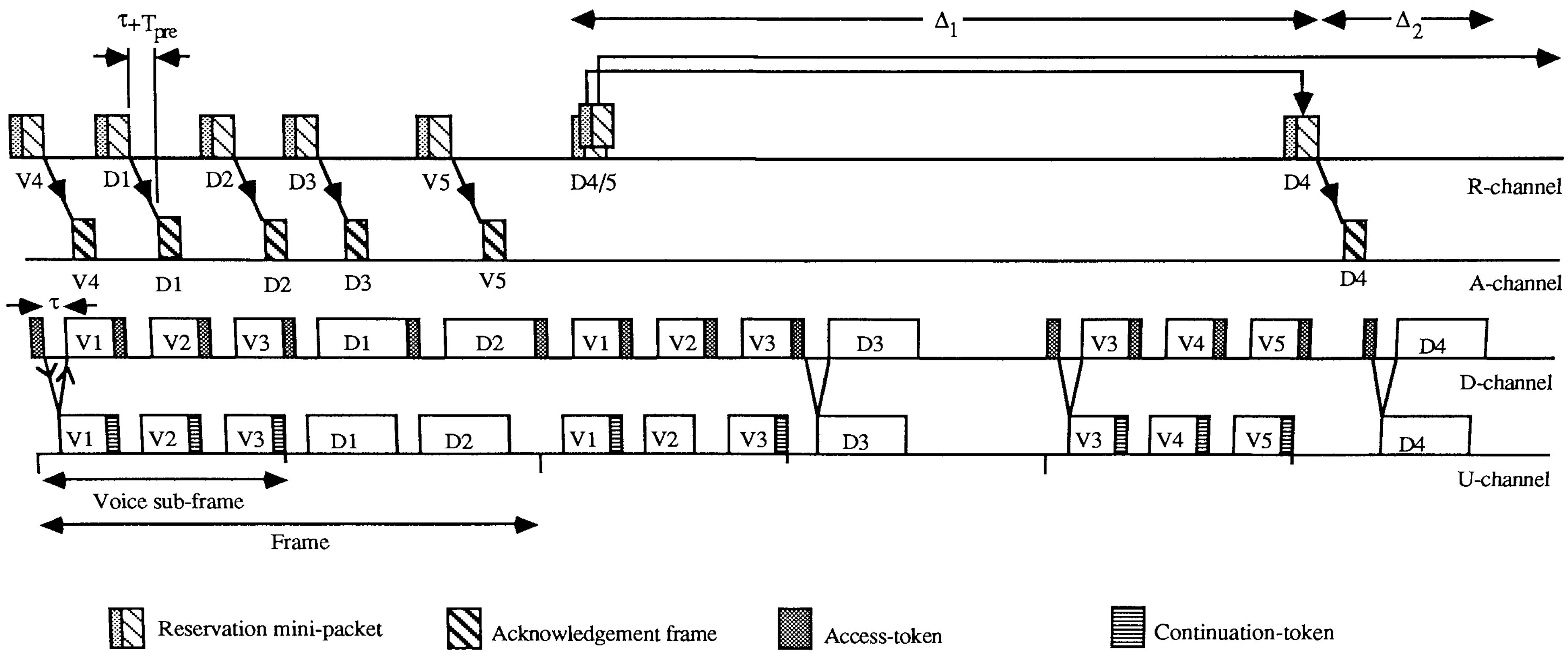


Figure 5.15 Logical TDMA operation using reservation and acknowledgement channels.

At the end of the talkspurt, voice stations relinquish their continuity. At the beginning of the next talkspurt they then contend again on the R-channel. However, it is more preferable for mobiles to maintain their continuation reservations throughout the duration of the call for two main reasons; (i) if the data traffic is heavy, voice reservation mini-packets could experience large delays in establishing a reservation and, (ii) for the cellular operation, at cell boundaries, it is simpler just for the base stations to exchange the mobile`s identification information than for the mobile to contend again. In the silent periods, mobiles on receiving their access-tokens would simply transmit back their continuation-tokens. For the protocol to remain robust to fading, if the base station does not receive a packet header correctly, or if the destination mobile does not acknowledge, the base station assumes that the mobile still requires a reservation for the next frame. If this is not the case, after maybe two timeouts in consecutive frames at the base station when it transmits an access-token, the mobile station could then be deleted.

The protocol for a single cell is described in figures 5.16 and 5.17 for a transmitting mobile and base station operations respectively.

while *in talkspurt* **do**
    begin
        repeat
            listen to D-channel;
        until *address of access-token equal to address of station j on the D-channel*
        initiate packet transmission on U-channel;
        while *transmission is incomplete* **do**
            begin
                if *stop-signals or another access-token sensed on D-channel* **then**
                    cease transmission on U-channel;
                else
                    complete transmission;
            end
        if *transmission is completed and is not final packet in talkspurt* **then**
            append continuation-token;
    end

Figure 5.16 Operation of transmitting mobile *j*, using Logical TDMA in the radio environment.

while *voice region is incomplete* **do**
    begin
        if *scheduled voice queue is greater than zero* **then**
            transmit access token on D-channel for first queued station;
        if *transmission has not been sensed on U-channel after timeout* $\tau+T_{pre}$ **then**
            assume station is in a fade;
        else
            begin
                while *transmission is incomplete* **do**
                    begin
                        transmit packet on D-channel;
                        if *header sensed in error or receiver has not acknowledged transmission after tiemout* $\tau+T_a$ **then**
                            repeat
                                transmit stop bits on D-channel;
                            until *no transmission sensed on U-channel*
                    end
            end
            if *transmission received correctly with no appended continuation -token* **then**
                delete station from voice reservation queue;
            else
                maintain reservation;
    end
if *not all queued voice stations have transmitted in voice region of current frame* **then**
    assign new head of queue using round-robin discipline;

Figure 5.17 Base station operation of the Logical TDMA protocol in the radio environment.

5.6.4 Analysis Of Voice Packet Loss, Data Packet Delay And Bandwidth Allocation

Throughout this section, the analysis of packet loss, data packet delay and bandwidth allocation are examined for the Logical TDMA protocol described in the previous sections. For simplicity, an error-free environment is assumed.

As illustrated in chapter 4, either fixed length or variable length voice packets can be used. The fraction of bit loss using variable length packets is found through simulation in the next section. If fixed length packets are used, then the fraction of speech loss is as given in (4.5) which is shown below

$$P_{loss} = \frac{1}{\gamma N} \sum_{n=L+1}^{N} (n-L) \binom{N}{n} \gamma^{n} (1-\gamma)^{N-n} \qquad ...(5.19)$$

where N is the number of voice stations within the radio cell, γ their fraction of talkspurt activity, and L the maximum number of voice transmissions the base station can schedule within the voice region of a frame. It is assumed that voice stations entering a talkspurt successfully transmit on the R-channel within one frame time. This could be assured through some form of priority for voice reservation mini-packets over data reservations.

Consider now the data packet delay. The total data packet delay is defined as the time period from where a station has a packet to transmit until when it is received at the destination. The delay a data packet experiences is comprised of two components, Δ_1 and Δ_2, which are illustrated in figure 5.15.

Δ_1 is the delay a data station encounters transmitting a successful reservation mini-packet on the R-channel and so is obviously a function of the access protocol used. For the slotted non-persistent CSMA algorithm used, this may be found either from simulation or by analysis. The analysis described in [111] examines the delay of such a protocol. Ignoring voice reservations (which will only constitute a small fraction of the reservations, since voice stations only reserve at the beginning of talkspurts), the delay for a finite population can be expressed as

$$\Delta_1 = E[n]\, T_{dr} / S_d \qquad \text{...(5.20)}$$

E[n] is the average number of backlogged stations, defined as a station which either had a collision or it found the channel busy, and which is derived in [111]. T_{dr} and S_d are the data reservation mini-packet transmission time and throughput respectively.

In order to calculate Δ_2, consider a preemptive-resume priority queueing system [57]. Assume a customer as a member of priority group p, where there are P different priority classes, indexed by the subscript p (p=1,2,...,P). The larger the value of the index associated with a group, the higher is the priority of that group. If Δ_{2p} is the average of the total time spent in the system by a customer from priority group p, then its average delay consists of three components. The first is its average service time $E[T_p]$. Secondly, there is the delay due to the service required by those customers of equal or higher priority whom it finds in the system. Assuming Poisson arrival statistics with mean arrival rate λ_i for priority group i, then by the conservation results for an M/G/1 queue [57], this mean wait is obtained from the Pollaczek-Khinchin mean value formula and is given by

$$\frac{\sum_{i=p}^{P} \lambda_i \overline{T_i^2} / 2}{1 - \sum_{i=p}^{P} \rho_i} \qquad \text{where } \rho_i = \lambda_i \overline{T_i}$$

Thirdly, it will be delayed by any customers who enter the system before it leaves and who are members of higher priority groups; the average number of such arrivals from group i must be $\lambda_i\Delta_{2p}$, each of which delays the customer by another $E[T_i]$ seconds.

In the protocol considered there are two priorities, one for voice and the other for data. If the voice region appears deterministically once every frame time, with a maximum duration of less than or equal to the frame time then, clearly, with this preemptive queueing, the delay experienced by the voice region in accessing the channel is zero. This is not the case for data packets which have a lower priority than voice. If a suffix d is used to represent data and if each data packet is treated separately, and a suffix v is used to represent the voice region within the frame, then the delay experienced by a data packet in this type of queue is given by

$$\Delta_{2d} \;=\; \overline{T}_d \;+\; \left(\frac{\lambda_d\overline{T_d^2}/2}{(1-\rho_d)} + \lambda_v\overline{T_v^2}/2\right) \;+\; \rho_v\Delta_{2d} \qquad ...(5.21)$$

assuming data packets have a Poisson arrival statistics and exponential service times, and where T_d and T_v are the service times for a data packet and the voice region respectively. The first term in (5.21) represents the mean transmission time of the data packet. The first term in parenthesis is the average number of data packets already queued up to be scheduled for transmission, including residual service time of any data packet currently being transmitted, and the second term in the parenthesis is the residual service time of the voice region. The final term in (5.21) is the mean number of voice regions by which the data packet is delayed whilst waiting to be transmitted. Hence the delay Δ_2 in the Logical TDMA protocol can be reduced to

$$\Delta_2 \;=\; \frac{(1-\rho_d)\overline{T}_d + \lambda_d\overline{T_d^2}/2 + (1-\rho_d)\lambda_v\overline{T_v^2}/2}{(1-\rho_d)(1-\rho_v)} \qquad ...(5.22)$$

For a fixed boundary scheme, the service time of the voice region, T_v, is fixed. Therefore the queueing delay the data packet experiences is given by

$$\Delta_2 \;=\; \frac{(1-\rho_d)\overline{T}_d + \lambda_d\overline{T_d^2}/2 + (1-\rho_d)\rho_v T_v/2}{(1-\rho_d)(1-\rho_v)} \qquad ...(5.23)$$

Hence the total delay a data packet experiences, Δ_D, for the Logical TDMA protocol with a fixed boundary, is given by (5.20) and (5.23),

$$\Delta_D = \frac{\bar{n}T_{dr}}{S_d} + \frac{(1-\rho_d)\overline{T}_d + \lambda_d\overline{T_d^2}/2 + (1-\rho_d)\rho_v T_v/2}{(1-\rho_d)(1-\rho_v)} \quad ...(5.24)$$

Another aspect to be considered with this protocol is what, for a given data throughput, is the optimal assignment of bandwidth to the different channels that will minimise the total delay Δ_D? A similar approach to that used in [108], which only considered data, is adopted. It is assumed that the data sources collectively form an independent Poisson source, with a mean arrival rate λ_d. Voice mini-packet reservations are again ignored, based upon the assumption that the data reservation traffic will be the more frequent, since voice stations need only transmit at the beginning of a talkspurt. Under steady-state conditions, λ_d is also the throughput. If the total available bandwidth is W, then the normalised throughput (average number of packets per transmission time of a packet on the total bandwidth) is denoted by

$$S_D = \lambda_d b_d / W \quad ...(5.25)$$

where b_d is the number of bits in a data packet transmission. If a fraction φ of the total available bandwidth is dedicated for the packet transmission channel i.e. the U-channel, then the bandwidth assigned to the R and A-channels is given by

$$W_R = W_A = (1-\varphi)W/2 \quad ...(5.26)$$

assuming they are of equal capacity. If the normalised input rate on the R-channel is denoted as S_d (average input rate on the R-channel per transmission time of a packet on the R-channel, T_{dr}), this can be expressed using (5.25) and (5.26) as

$$S_d = \lambda_d T_{dr} = \frac{2\lambda_d b_r}{W(1-\varphi)}$$

$$= \frac{2\varsigma S_D}{(1-\varphi)} \quad ...(5.27)$$

where $\varsigma = b_r/b_d = b_a/b_d$. This assumes that the message sizes of packets on the R and A-channels, b_r and b_a respectively, are the same. For the error-free environment assumed, the only packets transmitted on the A-channel are in acknowledgement to successfully received request mini-packets. Therefore, the assumption that $\varsigma = b_r/b_d = b_a/b_d$ appears reasonable.

To calculate the maximum bandwidth utilisation, the following two constraints must be satisfied:

$$S_d \leq C_r$$
$$\rho_d \leq (1-\rho_v)$$

The first condition states that the normalised throughput of the R-channel must be less than the capacity of the R-channel, C_r. The second condition states that the data transmission utilisation on the U-channel ρ_d, which equals $\lambda_d T_d = S_D/\varphi$, must not exceed the available utilisation of the data region within a frame, again assuming a fixed boundary. The maximum input rate S_d, is determined by the tighter of these two constraints for a given value of φ. Therefore, in a similar manner to [108], the maximum bandwidth utilisation C_D, is obtained from the above two constraints by:

$$C_D = \max_{0 \leq \varphi \leq 1} S_D \qquad \text{with the constraints}$$

$$S_d = \frac{2\varsigma S_D}{(1-\varphi)} \leq C_r$$

$$\rho_d = \frac{S_D}{\varphi} \leq (1-\rho_v)$$

The solution of C_D can be obtained from the above two constraints through the following equation:

$$\frac{C_r(1-\varphi)}{2\varsigma} = \varphi(1-\rho_v) \qquad \ldots(5.28)$$

However, the capacity of the R-channel C_r is also a function of φ. Therefore, to calculate the optimum value of φ, the following two equations can be plotted,

$$S_D = \varphi(1-\rho_v)$$
$$S_D = \frac{C_r(1-\varphi)}{2\varsigma}$$

which are illustrated in figure 5.18. The intersection of these two equations gives the optimum value of φ. Assume the values of a_r and ς are $0.05(1-\varphi)$ and 0.46 respectively. a_r is the ratio of the slot time to reservation mini-packet transmission time on the R-channel and equals $(1-\varphi)\tau/(2b_r/W)$. This can correspond to a total capacity of 400Kbps with a slot time of 35μs, b_r=140bits and b_d=300bits. Therefore, if ρ_v is 0.7, from figure

5.18 it can be seen that the joint capacity of the R and A-channels should be approximately 12% of the total capacity, W. Hence, for a total capacity of 400Kbps, the U-channel capacity is approximately 352Kbps and the capacities of the R and A-channels are 24Kbps each. The U-channel capacity is also equal the capacity of the D-channel, since this is merely an extension of the U-channel, only at a different frequency.

These capacities are assigned on the basis of the voice utilisation, ρ_V, of a frame. Since, with this centrally controlled scheme, the complexity of a movable voice/data boundary is no more than that of a fixed boundary, data packet transmissions should be allocated any unused capacity within the voice region. However, on the assumption above that ρ_V is 0.7, the reservation mini-packet throughput on the R-channel would limit the utilisation of the U-channel by data packet transmissions within the voice region, if the voice utilisation is lower than 0.7. Voice stations must be allocated a definite fixed capacity within a frame time. If the voice load is unpredictable, any spare capacity within the voice region should be allocated for data. Therefore, the capacity of the R and A-channels should be larger than that for ρ_V is 0.7, when the voice load is unpredictable. There is obviously a trade-off between wanting to use any excess capacity in the voice region when the voice load is low and wasting excess capacity on the R and A-channels when the voice region is utilised completely by voice.

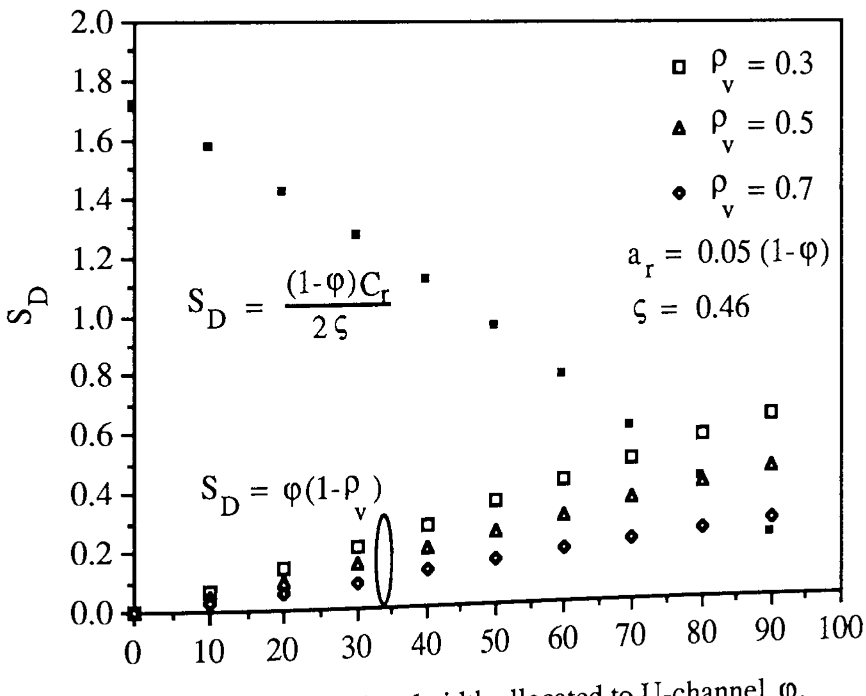


Figure 5.18 Bandwidth assignment of Logical TDMA, for transmissions towards the base station.

5.6.5 Simulation Results Of Logical TDMA Over Mobile Radio

The maximum voice utilisation within the frame is 0.8 for a voice/data load. However, since the voice load can be unpredictable, the R and A-channel capacities have been calculated based upon the previous error-free analysis for $\rho_v = 0.3$. The capacities of the U and D-channels are 300Kbps each, and, from figure 5.18 for $\rho_v = 0.3$, the capacities of the R and A-channels are 50Kbps each. The reservation mini-packets are 140 bits in length with a 16 bit preamble. Reservation is achieved using non-persistent ICMA. The frame time for this version of Logical TDMA is 15ms with voice encoded at 16Kbps. The maximum mobile-to-mobile propagation delay via the central base station is 3μs. Thus the slot times on the R-channel are 35μs.

Initially, the protocol was examined for a voice load only. In this case, the voice region equals the duration of a frame time. Figure 5.19 illustrates the percentage of voice packets lost using the Logical TDMA protocol with fixed length packets. As the signal-to-noise ratio decreases, the performance clearly deteriorates. Figure 5.20 shows the delay experienced by packets at the receiver. They are in general agreement with equation (4.8). The delay with a fading channel is slightly higher than the error-free channel because of packet transmissions being rescheduled later in the frame, if the original packet was transmitted in a fade. The standard deviation of delay is significant at high loads because packets can have delays between zero and three frame times.

The clip statistics for the protocol under these conditions are illustrated in table 5.3. It can be seen from the maximum length clip that packets are only lost when there is no room for them to be transmitted within a frame, i.e. the whole packet of length 15ms is discarded. Thus, whenever a station wishes to join the queue at the base station, its reservation transmission via the R-channel using non-persistent ICMA is always achieved within one frame time. Hence the upper limit on the clip length under normal operating conditions using fixed length packets is one frame period.

Figure 5.21 illustrates the percentage of voice bits lost with Logical TDMA when variable length packets are used. The frame time is again 15ms and a maximum time constraint of 30ms is used. Channels with signal-to-noise ratios of 10dB can again support fewer conversations due to retransmissions on the U or D-channels when original transmissions occur during a fade. All reservations on the R-channel at 10dB were successful within a frame time. Figure 5.22 illustrates the delay characteristics of the voice packets. For all channel conditions the packet delay has a maximum upper limit of 30ms. It can be seen that variable length packets can support a greater number of conversations than fixed length packets. The reason for this is that, at high loads, voice stations will invariably have to wait longer to be transmitted since there is only limited space within a frame.

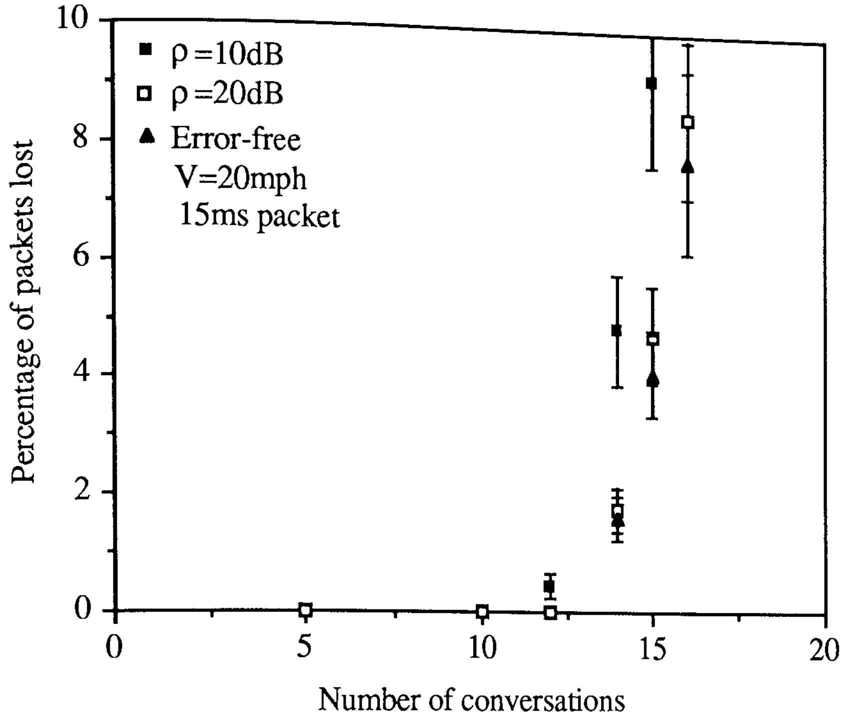


Figure 5.19 Percentage of voice packets lost for various voice loads using Logical TDMA over mobile radio with fixed length packets.

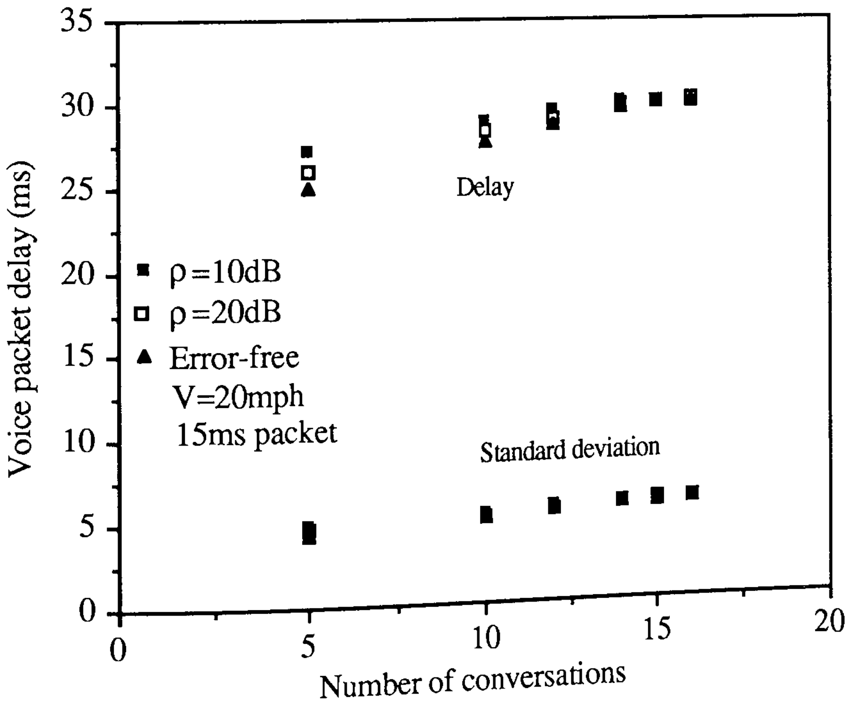


Figure 5.20 Voice packet delay at various voice loads using Logical TDMA over mobile radio with fixed length packets.

When variable length packets are finally transmitted, their increased length reduces the per-packet overhead, whereas, for fixed length packets, this overhead remains constant.

14 conversations	Percentage of voice packets lost	Clip Length (ms) Mean	Std dev	Max
ρ=10dB V=20mph	4.87	15	0	15
ρ=10dB V=70mph	5.57	15	0	15
ρ=20dB V=20mph	1.77	15	0	15
ρ=20dB V=70mph	2.08	15	0	15
Error-free	1.63	15	0	15

Table 5.3 Clip statistics for Logical TDMA in the mobile radio environment with fixed length packets.

Table 5.4 shows the clip statistics for the variable length version of the protocol. Figure 5.23 compares the clip statistics of Logical TDMA using both fixed and variable length packets at various voice loads. It can be seen that, although the mean clip length for variable length packets is smaller compared to the fixed length scheme, the maximum length statistics can be significantly larger than one frame time with fixed length packets. Hence, although the use of variable length packets provides an overall lower bit loss, the subjective quality of the speech with the given clip statistics may be unacceptable to a listener. Figure 5.24 illustrates the protocol with fixed and variable length voice packets under a combined voice/data load. The base station assumes a movable boundary scheme, whereby voice transmissions can have a maximum frame utilisation of 0.8. It can be seen that a larger data load can be supported with variable length packets.

16 conversations	Percentage of voice packets lost	Clip Length (ms) Mean	Std dev	Max
ρ=10dB V=20mph	3.85	9.03	8.52	68
ρ=10dB V=70mph	3.98	9.82	9.26	76
ρ=20dB V=20mph	0.65	4.35	5.14	20
ρ=20dB V=70mph	0.74	5.24	5.93	29
Error-free	0.19	2.29	1.82	11

Table 5.4 Clip statistics for Logical TDMA in the mobile radio environment with variable length packets.

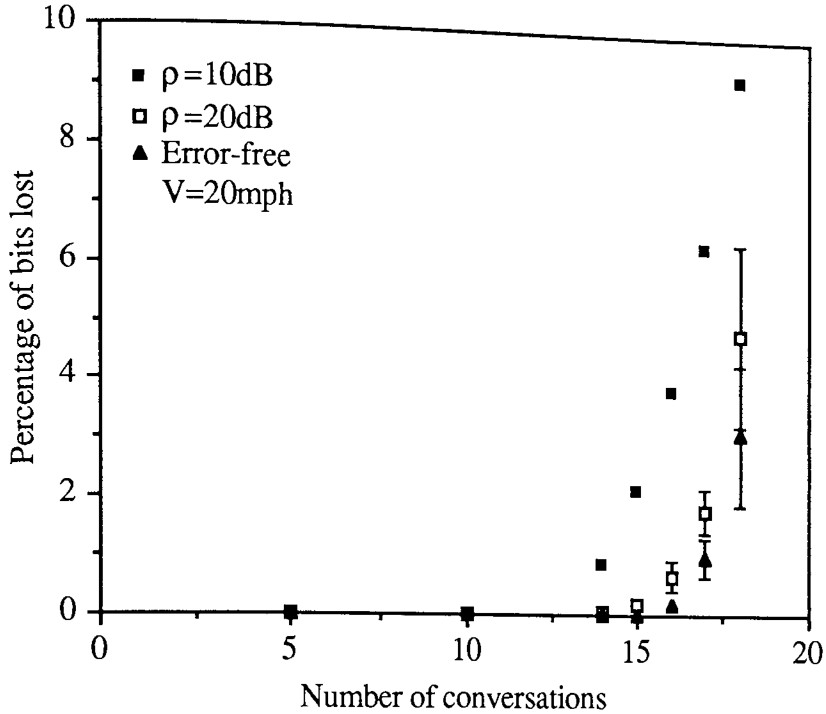


Figure 5.21 Percentage of bits lost using Logical TDMA with variable length packets over mobile radio channels.

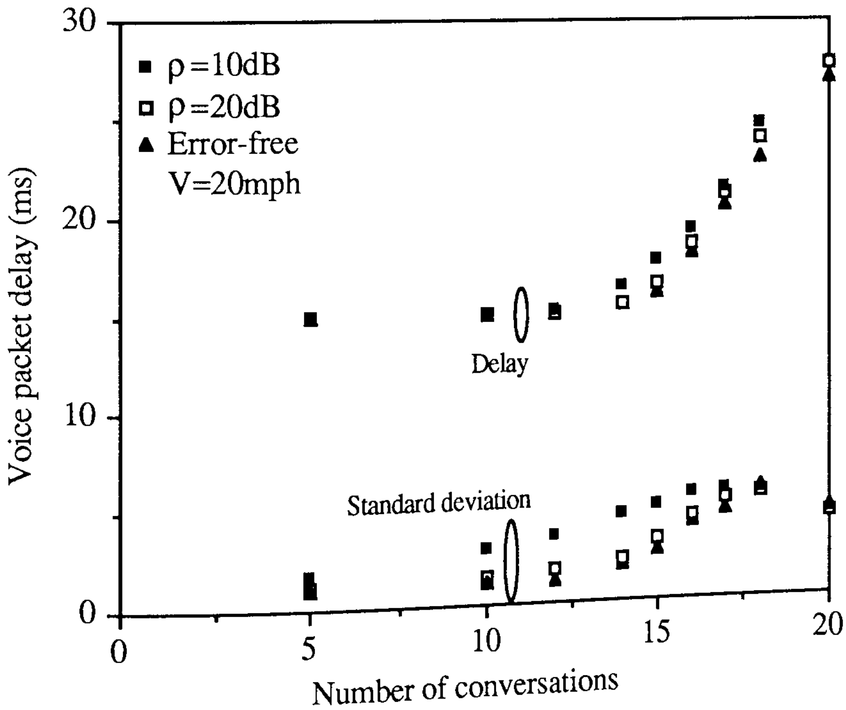


Figure 5.22 Voice packet delay and standard deviation of delay for various voice loads using Logical TDMA with variable length packets over mobile radio channels.

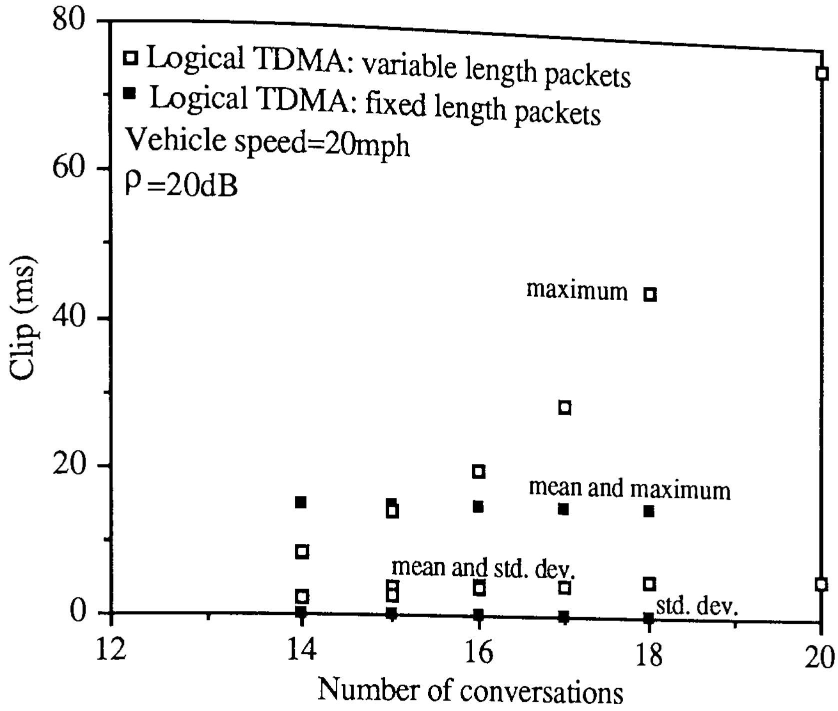


Figure 5.23 Mean, maximum and standard deviation clip statistics with Logical TDMA over mobile radio channels using fixed and variable length packets.

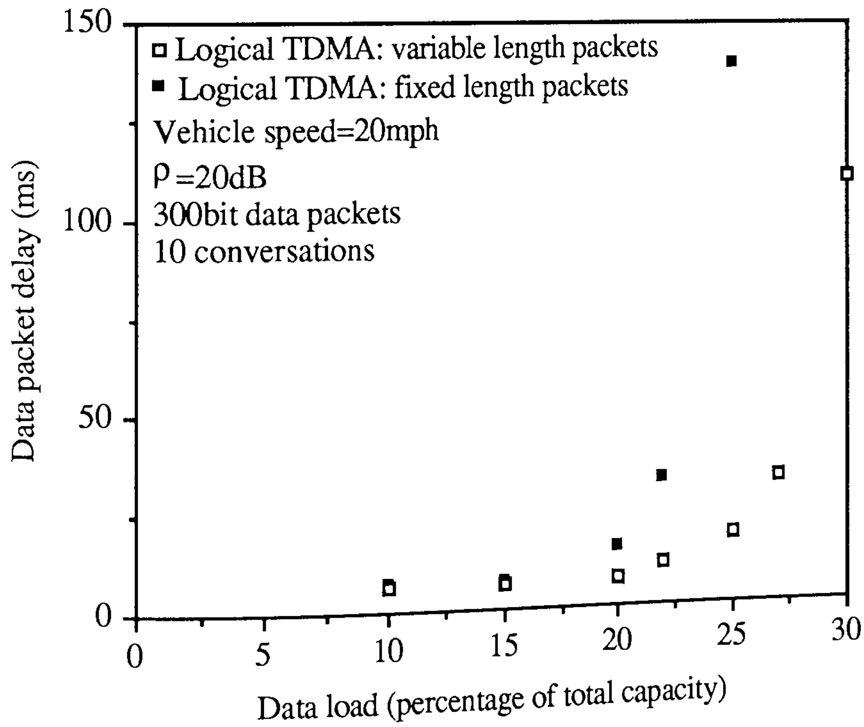


Figure 5.24 Data packet delay using Logical TDMA over mobile radio channels with a constant voice load of 10 conversations.

5.7 Summary

The AR-VT-CSMA/CD and Logical TDMA protocols have been examined over mobile radio channels. Adaptions to the protocols have been necessary to ensure that their operation was robust to signal fading. This was achieved through centralised control using a base station. Such base stations already form part of the cellular radio topology.

The AR-VT-CSMA/CD protocol takes advantage of voice periodicity through implicit reservations. The implicit reservation mechanism was found to break down when station`s virtual time clocks became misaligned due to fading. To overcome this, the virtual time lag of the real time clock was appended to transmissions on the D-channel by the base station. Even with this control, established stations could still collide through virtual time clock misalignment. The effect of these collisions and transmissions in a fade resulted in an inferior protocol performance compared to an error-free channel. Since the base station cannot detect collisions through superposition of signal strengths, collisions were only detected at the end of a transmission through error checking. This provides a relatively fast response time to a collision/fade for voice packets, since error checking is only computed on the packet header. Since the information in data packets is required to have high integrity, it is obviously necessary to provide error detection for all the data. To improve the base station`s response time to a collision/fade with a data packet transmission, error detection was also provided in the data packet header.

The Logical TDMA protocol utilised a frequency-division multiplexing approach for voice and data station reservations. Both voice and data stations used the R-channel for this purpose. The base station maintained the queue order by implementing a scheme of access-tokens on the D-channel. This provided the protocol with a form of robustness against mobile stations miscalculating their queue positions. The variable length packet version was found to support more conversations than when fixed length packets were used. However, in the region of operation (voice loss < 2%), fixed length packets have an upper length speech clip of one frame time, whereas the variable length packets exceed this limit. A comparison of the clip statistics is illustrated in table 5.5.

16 conversations ρ=20dB V=20mph	Percentage of voice bits lost	Clip Length (ms) Mean	Std dev	Max
AR-VT-CSMA/CD (η=30, P_{min}=25ms)	1.62	2.15	2.00	52
Logical TDMA (variable length)	0.65	4.35	5.14	20
Logical TDMA (fixed length)	8.52	15	0	15

Table 5.5 Comparison of clip statistics for AR-VT-CSMA/CD and Logical TDMA in a mobile radio environment.

The AR-VT-CSMA/CD protocol exhibits similar maximum length clip characteristics, although the mean length clip is much smaller than one frame time. Large clips occur with this protocol when a station attempts to find a free reservation slot. This may occur many times within a talkspurt, due to fades and also due to the possibility of hand-off while crossing a cell boundary. The Logical TDMA protocol has the advantage that once a voice station has reserved, it need not contend again for the remainder of the talkspurt (or call, if continuation-tokens are transmitted during silence periods). The Logical TDMA protocol also has the advantage of reduced complexity at the mobile stations since no virtual time clock management is necessary.

Compared to the currently available circuit switched systems, the protocols described can provide greater flexibility to user demands and also support a greater number of conversations. For example, a 300Kbps channel divided into frequency bands of capacity 16Kbps, could only support about 9 conversations, compared to 15 or more conversations for the packet protocols developed here.

CHAPTER 6

NETWORK MODELLING BY SIMULATION

6.1 Modelling

Network modelling is a necessary step before going to the expense of implementing a full working system. The system design process is effected by modelling, since experiments can be performed on which inferences can be drawn about the system without actually building it. Communication systems are often so complex that it can be extremely difficult to build analytically tractable performance models. However, simulation models, due to their versatility, can be employed as design assessors to synthesize and evaluate such communication systems.

Since a model is a description of a system, it is also an abstraction of the system. This abstraction in simulation involves building mathematical-logical relationships in accordance with the model objectives. Due to the complexity of analytical models, a high degree of abstraction is often necessary and considerable effort may be required by the network modeller to develop a performance model which accurately reflects the system under study. Reference to the model objectives should be made when deciding if an element of the system is significant and, hence, if it should be modelled. The model must be sufficiently complex to realistically reflect the important characteristics of the real system without including unnecessary detail. The amount of detail included in a model should be based on the model objectives, in this case the performance of various MAC protocols.

Once built, the performance of the simulation model must be evaluated through verification and validation. Verification determines if the translated model executes on the computer as intended. Validation determines if the simulation model is a reasonable representation of the system. Ideally, the validation should consist of a comparison based on experimental knowledge of the system performance.

6.2 Classes Of Simulation

Simulation modelling relies on system state descriptions. A system can be characterised by a set of variables, with each combination of variables representing a unique system state. The simulation is a representation of the dynamic behaviour of the system by moving it from state to state in accordance with well-defined operating rules. These rules in the communication networks are largely based on the network protocols. A system can be simulated using one of the following two techniques, depending on the nature changes of the system parameters or variables:

- ***Continuous simulation***. The state of the system is represented by dependent variables which change continuously over time. A continuous simulation model is constructed by defining differential equations for a set of state variables whose dynamic behaviour simulates the real system.

- ***Discrete simulation***. Discrete simulation occurs when the dependent variables change discretely at specified points in simulated time. State changes can only occur at event times. Since the state of the system remains constant between event times, a complete dynamic portrayal of the state of the system can be obtained by advancing simulated time from one event to the next. This is called the event orientated approach. It is also possible for the simulation to be process orientated, which models the flow of entities through a system.

Simulation models using these styles can be programmed using a general purpose language. However, there are several advantages in using a simulation language, including assistance in model formulation by providing a set of concepts for articulating the system description, which obviously saves programming time. The choice between particular simulation languages will depend upon the type of system to be simulated and the efficiency of program execution. A comprehensive review of such packages can be found in [62,99]. SLAM II (Simulation Language for Alternative Modelling) [96] is a simulation orientated language which can be utilised for both discrete and continuous models. It has been selected for simulating the protocols developed in this research since one can combine SLAM simulation statements with standard Fortran routines. SLAM also possesses an *event trace* facility which aides in the model verification by chronologically stepping through the system`s events.

6.3 Network Simulation Using SLAM

An isolated point in time where the state of the system may change is called an event time and the associated logic for processing the possible state change is called an event. It has been found more applicable to model the networks within the research using discrete event simulation.

In the discrete event simulation, network status changes occur only at the beginning of an activity, when a process has started, or at the end of an activity, when a process is terminated. Events are used to model the start and completion of activities. Time does not advance within an event and changes in the network state occur only at event times. Therefore a complete dynamic portrayal of the network state is obtained by advancing simulated time from one event to the next. The procedure for executing event orientated discrete simulation within SLAM is illustrated in figure 6.1.

```
repeat
    begin
        read SLAM statements;
        initialise variables;
        call INTLC;
        repeat
            begin
                remove first event from event calendar;
                set I equal to event code for this event;
                Advance simulated time TNOW, to the event time for this event;
                call EVENT(I);
            end
        until execution is complete;
        call OUTPUT;
        print SLAM summary report;
    end
until no more executions requested
```

Figure 6.1 Execution of discrete event orientated simulation using SLAM II.

The objects within the boundaries of the discrete system model are called entities. These entities represent the packets within the network model. Associated with each entity are a set of attributes, such as packet arrival time, source address, talkspurt packet number and the number of previous collisions etc. The aim of the discrete simulation model is to reproduce the activities that the entities engage in and thereby learn something about the behaviour and performance potential of the system. This is done by defining the states of the system and constructing activities that move the network from state to state. The basic states that can exist on the network channel is that it is either idle, i.e. no transmission, or it is busy. A busy state can represent either a collision or a successful transmission.

The behaviour of the protocols described in previous chapters have utilised four basic events, which dictate the activity and hence the state of the channel. Each station within the network executes a protocol at event times, the output of which characterises the network activity.

6.3.1 Event Definitions

The four events which characterise the network behaviour are briefly described below.

- ***The arrival event*** is used to generate the voice/data packet load on the network. The modelling of the voice load will be described in section 6.4. When a voice packet arrives, statistics are also collected for voice loss on voice packets that are waiting to be transmitted, but have exceeded the time constraint. The data load has an assumed Poisson arrival statistics, with a mean that is specified in the INTLC (initial) subroutine of figure 6.1.

- ***The update event*** is used to update the real and virtual time clocks of stations. The real time clock is updated periodically once every slot time in the synchronous protocols, where one slot time is the maximum network propagation delay. If the channel is idle, the virtual time clocks are updated. If the channel is idle, all stations are then tested to see if they have a packet that has arrived at a real time that is now less than or equal to the virtual time. If they have, then the packet is transmitted and the channel is assigned to servicing the particular station for a transmission time. If only one station transmits then the transmission is assumed good, the channel is set busy for a period equal to a transmission time and the Success event scheduled. If an unsuccessful transmission attempt occurs, the channel is assigned a service time corresponding to a jamming signal transmission time and the Delay event scheduled. In the Logical TDMA protocol, the update event initiates transmissions from the global queue. Over mobile radio channels, this LAN approach is slightly different, depending on whether a station was in a fade. The modelling of the mobile channel characteristics are described in section 6.5.

- ***The delay event*** is used to delay packets for random periods of time based on exponential backoff distributions and reset the channel free for service again. This can be due to either a collision - simultaneous station transmissions at the update event, a single packet transmission at the update event that was involved in a fade with the mobile radio network, or when the channel was sensed busy in the update event for data/reservation packet transmissions that use the non-persistent CSMA algorithm.

• ***The success event*** collects statistics on successful transmissions, e.g. the packet delay, and resets the transmission channel available for service again.

The flow of events for a voice packet using the previous descriptions is illustrated in figure 6.2.

```
while execution is incomplete do
    begin
        arrival event;
        repeat
            begin
                while packet has not been transmitted do
                    update event;
                if transmission is successful then
                    success event;
                else
                    delay event;
            end
        until successful transmission or time constraint exceeded
    end
```

Figure 6.2 Flow of events for a voice packet.

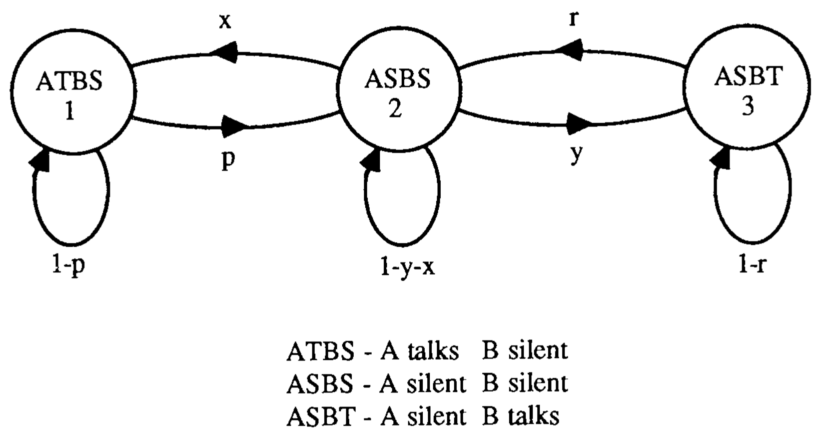
6.4 Voice Source Modelling

The distribution and correlation of talkspurts and silences has been examined for typical two-person conversations in [7]. To obtain a better fit to these measured speech patterns, models must grow correspondingly in sophistication and compiexity.

A talkspurt can be considered to be a contiguous sequence of packets from a single talker. A packet is classified when the speech exceeds an energy threshold detector within a frame time. During silence periods, when no speech exceeds the threshold detector, no packets are transmitted. It is assumed that talkspurts represent an integer number of packets, i.e. the talkspurt duration is an integer number of frame times, F. It is also assumed that only one member of the speaker-listener pair changes their speech activity state during a frame.

The conversation model has paired stations together, i.e. the model is not valid for conference calls. By ignoring the possibility of doubletalk when both participants talk simultaneously, the conversation can be represented by a 3-state Markov chain model as shown in figure 6.3. The assumption of ignoring doubletalk should make little difference, since it only accounts for approximately 5% of the time. More complex models which include the doubletalk can be found in [81].

For a particular talker, the talkspurt in this model is characterised by a single state and the silence period is characterised by two states. In normal conversation, the duration of active periods fits the exponential distribution reasonably well, unlike the duration of silent periods [7]. To facilitate analysis, the lengths of both talkspurt and silence periods have been assumed to be exponentially distributed. The quality of this model has been found to be good if the number of voice sources is more than about 25 and is less suitable if the number of voice sources is less than about 10 [117]. The values for these durations are dependent on the threshold level of the speech activity detector used. The durations of talkspurts and silent periods for a participant in each conversation were obtained from an exponential sampling procedure within SLAM, with mean lengths of 1.36 and 1.8 seconds respectively [7].



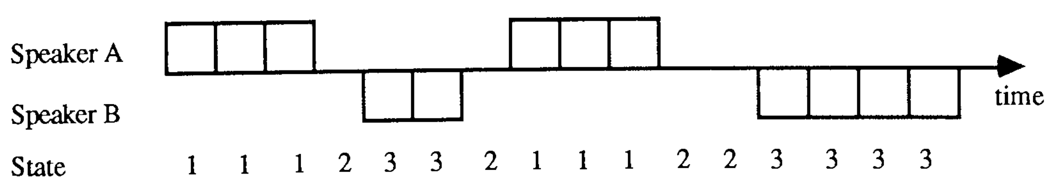


Figure 6.3 3-state Markov chain model, with possible sequence of states.

6.5 Modelling Of The Mobile Radio Channel

To determine the effect of errors on packet transmissions and virtual time clock update at mobile stations, it was necessary to model the mobile radio channel characteristics. Each station required its own perspective of the channel at any particular time, which determined whether it was in a fade or not.

The simplest channel model is the binary symmetric channel, where errors are considered statistically independent. The model is memoryless with a probability of error P_e due to additive Gaussian noise. It was suggested in chapter 5 that the mobile channel is characterised by bursts of errors, which are Rayleigh distributed. Thus the binary symmetric channel is inadequate, since it assumes that errors occur independently and so does not take into account the previous errors on the channel.

To simulate the mobile channel, Gilbert`s 2-state Markov chain model has been used [29], which is illustrated in figure 6.4. The model consists of two states, a good state and a bad state, representing non-fade and fade conditions respectively. The transitions from one state to another occur at random. The good state represents the case when the received signal is above the receiver threshold and consequently the probability of error is very low. The simulations assumed the probability of error in this state was zero. The bad state represents the situation where the signal value falls below the receiver threshold. A worst case situation with an error probability of 0.5 has been assumed. Thus, during the fade periods, the probability of data in error is very high. The probabilities of p and q are small, which means errors occur in bursts due to the high probability of remaining in the bad state.

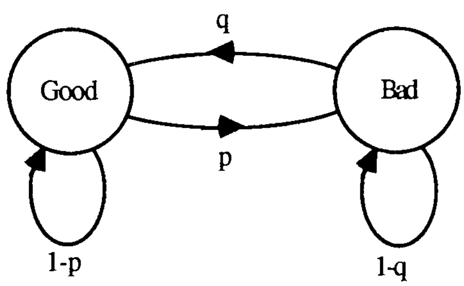


Figure 6.4 Gilbert model for mobile channel characteristics

Statistics of fade durations have illustrated the fact that they follow an exponential distribution [6]. The non-fade durations, however, are a function of the threshold level. For a threshold of -10dB or lower relative to the mean signal-to-noise ratio, the non-fade durations can also be approximated by an exponential distribution [77]. To use exponential non-fade distributions, the simulations in chapter 5 assumed a threshold level of -10dB. The mean durations of the fade and non-fade periods are given in equations 5.14 and 5.15 respectively. An exponential sampling procedure within SLAM, with mean lengths from 5.14 and 5.15, was used to obtain the durations of the fade and non-fade states. The number of bits within each period is simply the duration length multiplied by the channel capacity.

Since each station required its own perspective of the channel, each station`s channel state was updated once every slot time, at the update event described in 6.3.1. If any part of a header (or virtual time lag) appeared during a fade, the probability of success, i.e. no errors, was 0.5 to the power of the number of bits that were transmitted/received during the fade. If an error did occur, it was assumed that this was detected through error detection, and the packet (or virtual time lag) was not received correctly.

6.6 Modelled Network Parameters

The parameters used in the network models are illustrated in table 6.1. A hyphen indicates that a parameter was either not simulated, or is not applicable to the particular network type.

6.7 Design Of The Simulation Experiments

In this section, the various policies adopted in the design of the simulation experiments are described. Such policies are very important since they dictate the inferences one can make about an actual system. The simulation model is a stochastic one and therefore random elements of the model will produce outputs that are probabilistic. A stochastic process is said to be stationary if the underlying joint distribution of the random variables remains the same as time progresses. It is the stationary results that have been of interest throughout the research. However, tests for stationarity of a sequence are not well developed.

To obtain the steady-state results of a simulation, the simulation must be run long enough such that the initial state of the simulation on the performance measures of interest are negligible. The simulation results in this research were obtained from runs having a total simulation time of 1.5×10^6 time units each. In the synchronous LAN model, the time unit equalled the end-to-end propagation delay, $10\mu s$. Therefore, each simulation lasted for 15 seconds of real time. The time unit in the mobile radio network was equivalent to the propagation delay from mobile-to-mobile via the base station, plus the preamble detection time at the base station, which equals 35μs. Therefore, for the mobile radio model, each simulation lasted for 52.5 seconds of real time.

The start-up policy used for setting the initial conditions assumed that conversations commenced in the talkspurt state for each conversation pair and at a time taken from a uniform distribution in the range 0 to 200000 time units. To increase the confidence of the results, the transient portion of the simulation run, which depends on the initial conditions, was discarded and performance statistics collected only after the simulation had reached steady-state. The problem of determining the end of the transient phase is difficult, but the effects of the initial configuration becomes less important as the length of the simulation increases. The length of the transient period has been examined in [118]. The deletion of initial statistics tends to reduce the bias on the output results from the initial conditions. However, deleted values may also increase the variability in the estimates if the deleted values are samples that have a variability similar to the variability associated with steady-state samples.

A truncation period, where no statistics were measured in the range 0 to 500000 time units, has been used. With a simulation length of 1.5×10^6 time units, longer truncation periods were found not to significantly effect the measured variables. Simulation runs of 6×10^6 time units were examined with the same truncation period. The results indicated that the mean voice loss was within ±4% of the value obtained with a simulation run of 1.5×10^6 time units. A run time of 1.5×10^6 is preferable, since its execution requires less CPU time.

Since the performance predicted by one run of the simulation depends on the particular stream of pseudorandom numbers used to drive the stochastic process, the results of the model will typically vary from one run to another. To increase the confidence in the results, the model was run for each set of input parameters three times. For 100 conversations on the LAN with a frame time of 7.5ms, this corresponds to measured statistics on a total of approximately 3×10^5 voice packets for each plot on a graph. 95% confidence intervals have been used throughout the thesis, calculated from the t-distribution using sample means with the assumption that they are normally distributed.

The model validation process should answer the question of whether inferences drawn about the real network from simulation results are valid. In order to attempt to validate the simulation model, a CSMA/CD protocol for voice described in [26] was examined using the same network parameters. The simulation results obtained showed very good agreement in the percentage of voice lost compared with the experimental system results published in [26].

Parameter	Local Area Network	Mobile Radio Network
Capacity	10Mbps	300Kbps + R, A-channels
Carrier frequency	–	800 - 804 MHz
End-to-end propagation delay	10μs	3 μs + preamble detection time, 32μs
Voice encoding rate	64Kbps	16Kbps
Voice frame overhead:		
Standard protocol	120bits	120bits
Ethernet protocol	208bits	–
Interframe spacing	10μs	10μs
Channel error characteristic	errorless	fast-fading
Channel collision duration	40μ s	400μs
Collision backoff, b - voice: (n=number of previous transmission attempts)	uniform distribution in range: $0 \leq b \leq \min\{2^n, \text{frame time}\}$	uniform distribution in range: $0 \leq b \leq \min\{2^n, \text{frame time}\}$
Mean talkspurt duration	1.36s	1.36s
Mean silence duration	1.80s	1.80s
R-VT-CSMA/CD:		
Voice packetisation interval	7.5ms	–
DC-R-VT-CSMA/CD:		
Message channel capacity	9.95Mbps	–
R-channel capacity	50Kbps	–
Voice packetisation interval	7.375ms	–
AR-VT-CSMA/CD:		
Message channel capacity	10Mbps	300Kbps
A-channel capacity	–	50Kbps
Voice packetisation interval	7.5 - 14.9ms	10 - 29ms
Logical TDMA:		
Message channel capacity	10Mbps	300Kbps
A, R-channel capacities	–	50Kbps
Voice packetisation interval	7.5ms	15ms
Frame length - data:		
standard	1000bits	300bits
Bimodal	90% 800bits, 10% 8000bits	–
Collision backoff, b - data: (n=number of previous transmission attempts)	uniform distribution in range: $0 \leq b \leq 2^k$, where k=min(n,10)	uniform distribution in range: $0 \leq b \leq 2^k$, where k=min(n,10)

Table 6.1 Simulation model - network parameters.

6.8 Summary

Due to the complexity of communication protocols, it is often necessary to model them using simulation techniques rather than through exact mathematical analysis. In this chapter, the basic structure of the event-orientated discrete simulation used to model the networks has been discussed. The main stages of development are identified as:

- ***Problem formulation*** involves defining a problem-solving objective.
- ***Model building*** creates an abstraction of the system into mathematical-logical relationships in accordance with the problem formulation.
- ***Model verification*** is the process of ensuring that the model executes the way an experimenter intended.
- ***Model validation*** tests the accuracy between the behaviour of the model and that of the real system.

The basic structure of the model was described using events, based around the objective of medium access control performance. Due to various advantages over standard high level languages, including assistance in model formulation, the simulation language SLAM II was used. Model verification was by inspection, using an event trace in SLAM. The model design and network parameters were then discussed. The performance of the model using the CSMA/CD protocol was compared to that of an experimental system to provide a form of validation. This comparison was found to be in close agreement. Validating the protocols developed in this research is more difficult due to their originality and hence the lack of comparisons with real systems. However, these protocols have been logically verified using the event trace and the network model appears to be satisfactory, due to the good agreement when supporting the CSMA/CD protocol.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

7.1 Introduction

Medium Access Control protocols for packet voice and data have been investigated in local broadcast networks. Contention-type LANs have been examined, along with a mobile radio network using a central base station, which share a similar topology. Previous work on such protocols was examined and it was evident from past research that the protocols which could support a greater number of conversations took direct advantage of voice periodicity within talkspurts. This was achieved through the use of implicit or explicit reservations, rather than merely upon random transmissions as in the CSMA/CD protocol.

Due to the spatial distribution and changing channel requirements of the stations, it was impossible for stations to schedule themselves into a perfect single server queue. Stations formed a queue through reservations, by using channel capacity to exchange protocol information, either in the form of a collision or explicit reservation mini-packets.

7.2 Advantages And Disadvantages Of The LAN MAC Protocols

The first two developed LAN protocols adopted the idea behind the R-VT-CSMA/CD protocol, where virtual time clocks are used to determine the transmission order, through implicit reservations based on packet arrival times. The DC-R-VT-CSMA/CD protocol increased the time-bandwidth product of the protocol by using a dual channel architecture. The advantage of this is that contention for the next packet transmission on the narrowband signalling channel occurs simultaneously with the current packet transmission on the wideband message channel. This dual channel protocol was found to support more conversations than the original protocol, at the expense of greater complexity necessary for a second channel. This could either take the form of a second cable in a baseband scheme, or through FDM in a broadband scheme.

The AR-VT-CSMA/CD protocol achieved a greater capacity than these two fixed length packet protocols, through the use of variable length packets. Again implicit reservations were used, although this time designated as one packetisation interval in the future from when a station had transmitted all the voice bits within its buffer. More conversations can be supported with this protocol due to a larger virtual time clock update factor, since stations are now separated by a minimum of one transmission interval, and also through the use of variable length packets which reduces the per-packet overhead as the channel load increases.

The protocols using the virtual time idea achieve higher throughputs than the CSMA/CD protocol through the use of implicit reservations, to reduce the packet collision rate. The disadvantage of this though, is the cost and complexity of stations, since each requires a real time clock, a virtual time clock, and two buffers for storing the packet arrival and offset times. This complexity can lead to problems of robustness. The protocols are fully distributed, so if one station fails, the whole communication operation will not also fail. However, a different speed ratio of virtual time clock to real time clock at a station can corrupt implicit reservations at other stations. Therefore, it is suggested that there is a master station which periodically (dependent upon the tolerances of station clocks) broadcasts the correct values (or difference) of real and virtual time. The other main disadvantage of these protocols is that, at high load, the maximum speech clip statistics may be annoying to listeners, even with only 2% voice loss.

The final LAN protocol that was developed, Logical TDMA, formed a distributed queue through the use of explicit reservation mini-packets. This protocol has the advantage that once a station has reserved its position in the logical queue, it maintains a position for the remainder of the talkspurt. Reservation packets are transmitted in the data region of a frame (or by using a separate reservation channel). These should have priority over data transmissions, which was found to allow all stations commencing a talkspurt to join the queue within one frame time. The main advantage of this protocol is that very efficient throughput can be achieved, since only one reservation is necessary per talkspurt and each station knows when to transmit based upon its logical queue position. At overload, the voice stations under distributed control discard fixed length packets through the round-robin approach, to assure fairness. Thus, mean and maximum speech clips with fixed length packets is one packetisation time. With variable length packets, the mean clip is less than one frame time and the maximum clip is of the order of one frame, if the data region is sufficiently long enough for reservation contention.

The main problem of robustness that can occur is if a station miscalculates its position in the logical queue due to channel noise etc. In the case of a collision, all involved stations should relinquish their queue positions and transmit new reservation packets. In the case of a station failure, or collision, all succeeding stations should maintain a timeout as to when they expect to transmit.

Table 7.1 summarises the characteristics of the LAN protocols designed in chapters 3 and 4. If a 10Mbps channel were available for circuit switched voice encoded at 64Kbps, approximately only 78 conversations could be supported, which is significantly less than the developed protocols.

If sufficient reservation time is available, the Logical TDMA protocol with variable length packets appears to be the most preferable. Compared to the fixed length version, since whole packets are not discarded in a round-robin fashion, the mean clip length is significantly less than one frame time, while the maximum length clip should be contained through suitable reservation time, and on a limit of the maximum number of queued stations. This could simply be achieved by new stations wishing to set up a call enquiring to a control station via the data region, as to the number of queued stations, before transmitting a reservation packet.

It has the simplicity over the virtual time protocols, which generally means lower cost and greater robustness. Only two buffers are required, one for the total number of queued stations and the other for the station`s queue position. Also voice and data integration is simply based on either a fixed or movable boundary, which guarantees data stations channel time. This is less complex than the virtual time protocols which require either collision handshaking or an extra virtual time clock at each station for data.

7.3 Advantages And Disadvantages Of The Mobile Radio MAC Protocols

The AR-VT-CSMA/CD and Logical TDMA protocols were also examined in the mobile radio environment. Due to the channel`s time-varying characteristics, it was necessary that the protocol`s LAN operations be modified to ensure that they were robust to signal fading.

The problems identified with the AR-VT-CSMA/CD protocol was that of virtual time clock misalignment, due to stations not sensing the commencement of transmissions due to fading. This then led to a breakdown of the implicit reservation mechanism since station`s reservation slots overlapped reservation slots from other stations. This problem was greatly reduced by the base station appending its virtual time lag of the real time clock to packet transmissions. This mechanism also provides a form of robustness against the possibility of the speed ratio of the real and virtual time clocks at a mobile station from being offset to that of other stations.

	R-VT	DC-R-VT	AR-VT	L-TDMA	L-TDMA
Packet length	Fixed	Fixed	Variable	Fixed	Variable
Number of conversations with 2% voice loss	101	110	134	114	135
Maximum packet delay	15ms	15ms	15ms	22.5ms	15ms
Mean speech clip at 2% voice loss	3.9ms	1.9ms	0.8ms	7.5ms*	0.7ms*
Maximum speech clip at 2% voice loss	181ms	134ms	470ms	7.5ms*	8.33ms*
Flexibility for data	Virtual time clk. or collision h/s. Complex to guarantee time.	Virtual time clk. or collision h/s. Complex to guarantee time.	Virtual time clk. or collision h/s. Complex to guarantee time.	Simple fixed or movable boundary.	Simple fixed or movable boundary.
Cost/complexity	Real time clk. virtual time clk. arrival and offset time buffers.	Real time clk. virtual time clk. arrival and offset time buffers.	Real time clk. virtual time clk. arrival and offset time buffers.	Queue position buffer and no. of queued stations buffer.	Queue position buffer and no. of queued stations buffer.
Robustness	Master station to transmit times periodically.	Master station to transmit times periodically.	Master station to transmit times periodically.	If collision then relinquish queue position. Successor timeouts.	If collision then relinquish queue position. Successor timeouts.

*Results with sufficient reservation to limit speech clip (3.74ms)

Table 7.1 Characteristic summary of the LAN protocols.

The logical TDMA protocol used a separate reservation channel for voice and data stations to join the logical queue. The main difference of the protocol operation in this environment is that mobile stations cannot reliably calculate their queue position to enable collision-free packet transmissions. The reason for this is that a mobile station may not sense a reservation request from another mobile station due to signal fading and so, if the voice capacity is exceeded for the frame, the mobile in the fade will miscalculate its queue position. To overcome this, the base station maintained the transmission order by appending access-tokens to transmissions to notify the next station of its transmission rights. To maintain a queue position, it was necessary for stations to append their transmissions with continuation-tokens. If a station does malfunction on its transmission turn, the base station simply times out and transmits the next access-token.

The characteristics of these two protocols operating over mobile radio channels is illustrated in table 7.2. Compared to the currently available circuit switched systems, all the protocols can support more conversations. For example, a 300Kbps channel, divided into frequency bands of capacity 16Kbps each, could only support 9 conversations.

Again, the Logical TDMA protocol with variable length packets seems the more suitable. It has the advantage over the AR-VT-CSMA/CD protocol due to its simplicity, there is no virtual time clock management. Also, once a station has reserved a queue position, it need not contend again for the remainder of the talkspurt (or call, if continuation-tokens are transmitted during silence periods). The AR-VT-CSMA/CD protocol, however, must attempt to find a free reservation slot at the beginning of each talkspurt and cell crossover. It is the searching for these reservation slots that produces large voice clips.

	AR-VT	L-TDMA	L-TDMA
Packet length	Variable	Fixed	Variable
Percentage of voice lost with a load of 16 cons.	1.62%	8.52%	0.65%
Maximum packet delay	30ms	45ms	30ms
Mean speech clip with a load of 16 cons.	2.15ms	15ms	4.35ms
Maximum speech clip with a load of 16 cons.	52ms	15ms	20ms
Flexibility for data	Virtual time clock or collision handshaking. Complex to guarantee time.	Simple fixed or movable boundary.	Simple fixed or movable boundary.
Cost/complexity	Real time clock, virtual time clock, arrival and offset time buffers. U, D, A-channels.	Continuation-token. U, D, R, A-channels.	Continuation-token. U, D, R, A-channels.
Robustness	Base station appends times to packet transmissions.	Base station issues access-tokens and uses transmission timeouts.	Base station issues access-tokens and uses transmission timeouts.

Table 7.2 Characteristic summary of the mobile radio protocols.

Variable length packets are more preferable to fixed length packets in Logical TDMA due to an increase in the number of supportable conversations and reduced mean voice clips. The disadvantage, is however, that the maximum clip statistic may exceed one frame time.

7.4 Recommendations For Further Work

- It is necessary to examine the subjective effects of the speech clips of the various protocols in order to see how disruptive they are in conversation. The mean and standard deviation of the clips in the AR-VT-CSMA/CD and variable length Logical TDMA protocols are significantly less than that for the fixed length Logical TDMA protocol. However, the maximum speech clips can occasionally exceed that of one packetisation interval. It is necessary to understand the subjective implications of these maximum speech clips.

The effect of coding should also be considered on the subjective effect of voice loss in a conversation. In sophisticated coding algorithms, because decoding of the current speech sample depends on the previous decoded samples, packet loss affects speech quality successively. Therefore, missing packet reconstruction techniques used for conventional PCM speech are not applicable to packetised speech communication systems with low bit-rate coding systems. The coders used in the research assumed a constant output rate. Variable rate coding attempts to handle congestion in the network by reducing the coder bit-rate rather than discarding whole packets.

- Priority amongst voice packets should be considered, depending on whether the voice calls are intra-LAN or inter-LAN. Inter-LAN voice packets may experience much delay on route to the destination and so should be given priority over packets whose destinations are within the source LAN. A related issue to this is the continuously growing need for the interconnection of homogeneous or hetrogeneous LANs and future high speed LANs. At present, interconnection problems are solved by bridges. However, Asynchronous Transfer Mode (ATM) has the potential to be an efficient technique for high speed data communications and interconnection, either in the private sector or the public sector, since ATM is the target transfer mode solution for the future Broadband ISDN.

In ATM, information, whatever its type (speech, image or data), is transmitted in the form of fixed length cells which consist of a 5 byte header and a 48 byte information field. Using 64Kbps encoded voice, a cell would represent 6ms of speech. Speech encoded into cells at the source would reduce complexity at the LAN interface with an ATM network or intermediate connection to a Metropolitan Area Network (MAN), possibly using Dual Queue Dual Bus (DQDB), since this uses the same cell format.

Hence, with the fixed length packet protocols of chapter 3, a frame time of 6ms would be used. However, with the variable length packets, the amount of information can vary up to 2 cells. This implies that padding bits would be necessary to make the information field up to the specified length of 48 bytes. It is clearly undesirable for users to pay for the transmission of nearly empty cells on the ATM network. Thus, although variable length packets can support a greater number of conversations on a LAN, if intra-LAN communication is required it may be preferable to use fixed length packets to reduce cost and complexity.

7.5 Summary

In this chapter, the advantages and disadvantages of the developed protocols have been discussed. The merits of a protocol not only depended on the number of conversations it could support, but also the manner in which speech was lost, and the complexity of its implementation which affects its robustness to errors and failures. Some further work was then recommended, based on the subjective effects of speech loss and how these could be minimised through coding, and also on priority of voice packets within a LAN, depending upon whether the voice call was inter-LAN or intra-LAN based.

PUBLICATIONS

1. Malyan A. D. and Brewster R. L.
A simulation study of using Reservation Virtual Time CSMA for packet voice in mobile radio,
6th U.K. IEE Teletraffic Symp., pp.16/1-6, Harlow, May 1989.

2. Malyan A. D. and Brewster R. L.
Performance of dual-channel R-VT-CSMA for packet voice/data in local area networks,
Electron. Lett., Vol.25, pp.1452-1454, Oct. 1989.

3. Malyan A. D. and Brewster R. L.
Using Reservation Virtual Time CSMA with split-channel reservation for packet voice and data,
U.K. IT Conf. (IEE), pp.209-212, Southampton, Mar. 1990.

4. Malyan A. D. and Brewster R. L.
Medium access control protocol for the integration of voice/data in local area networks,
Electron. Lett., Vol.26, pp.592-593, Apl. 1990.

5. Malyan A. D. and Brewster R. L.
The performance of R-VT-CSMA with split-channel reservation for packet voice/data in a LAN,
13th Int. Switching Symp., Vol.2, pp.A4.3, Stockholm, May 1990.

6. Malyan A. D. and Brewster R. L.
A speech packet protocol for mobile radio using Idle-Signal Casting Multiple Access,
3rd IEE Conf. on Telecommun., (in press), Edinburgh, Mar. 1991.

REFERENCES

1. Agrawal J. P. and Patel V. M.
An advanced reservation multiple access (ARMA) protocol for integrated local area networks,
IEEE Globecom, vol.1, pp.1-5, Dec. 1986.

2. ANSI/IEEE Standards 802.1 to 802.6 for local area networks, 1985.

3. Barberis G. and Pazzaglia D.
Analysis and optimal design of a packet voice receiver,
IEEE Trans. on Commun., Vol.COM-28, pp.217-227, Feb. 1980.

4. Bartecki J., Daemen J. M. A. and Niemegeers I. G.
Protocols for integrated voice/data communication in local area networks,
Euromicro Smp., pp.55-64, 1985.

5. Beauchamp K. G.
Computer communications,
Van Nostranf Reihold (UK), Wokingham, 1987.

6. Botmann W. F. and Arnold H. W.
Fade-duration statistics of a Rayleigh distributed wave,
IEEE Trans. on Commun., Vol.COM-30, No.3, pp.549-553, Mar. 1982.

7. Brady P. T.
A statistical analysis of on-off patterns in 16 conversations,
Bell System Tech. J., Vol.47, No.1, pp.73-91, Jan. 1968.

8. Brodsky I.
Introducing wireless LANs,
Bus. Commun. Rev. (USA), Vol.20, No.5, pp.112-120, 1990.

9. Brown C. D.
IEEE standards and issues,
Inf. Technol. Libr. (USA), Vol.9, No.1, pp.89-93, Mar, 1990.

10. CCITT Recommendation G.114, Red book, Vol.III, Oct. 1984.

11. Chen T. and Messerschmitt D.
Integrated voice/data switching,
IEEE Commun. Magazine, Vol.26, pp.16-26, June 1988.

12. Chen W., Li S. and Schwartz M.
A new voice scheduling scheme for broadcast bus local area networks,
IEEE Infocom, Vol.2, pp.448-57, April 1989.

13. Chlamtac I.
An Ethernet compatible protocol for real-time voice/data integration,
Computer Networks and ISDN Syst., Vol.10, No.2, pp.81-96, 1985.

14. Companella S. J.
Digital speech interpolation,
Comsat. Tech. Rev., Nol.6, pp.127-158, Spring 1976.

15. Conviello G. J.
Comparative discussion of circuit-versus packet-switched voice,
IEEE Trans. on. Commun., Vol.COM-27, No.8, pp.1153-1159, Aug. 1979.

16. Cox D. C.
Portable digital radio communications-an approach to tetherless access,
IEEE Commun. Magazine, Vol.27, pp.30-40, July 1989.

17. Davidovici S.
On the radio (wireless LANs),
BYTE (USA), Vol.15, No.6, pp.224A-B,224D,226,228, Jun. 1990.

18. DeTreville J.
A simulation-based comparison of voice transmission on CSMA/CD networks and on token buses,
AT&T Bell Lab.Tech. J., Vol.63, No.1, pp.33-55, Jan. 1984.

19. Dobrowski G. H., Kerner M., Spears D. R. and Wilson D. S.
Evolving the network toward B-ISDN,
XIII Int. Switch. Symp., Stockholm, Vol.1, pp.1-7, May 1990.

20. Duc N. Q. and Eng. K. C.
ISDN protocol architecture,
IEEE Commun. Magazine, Vol.23, No.3, pp.15-22, March 1985.

21. Dunlop J. and Rashid M. A.
Speech transmission capacity of standard Ethernet systems,
J. Inst. Electron. & Radio Eng. Vol.55, No.3, pp.119-122, 1985.

22. Dunlop J.
Techniques for the integration of packet voice and data on IEEE 802.3 LANs,
Comput. Commun. (UK), Vol.12, No.5, pp.273-80, Oct. 1989.

23. Fluhr Z. C. and Porter P. T.
Advanced mobile phone service: control architecture,
Bell Syst. Tech. J., Vol.58, No.1, Jan. 1979.

24. Forgie J. W.
Speech transmission in packet-switched store-and-forward networks,
Proc. of NCC, pp.137-142, 1975.

25. Forgie J. W.
Network speech implications of packetised speech,
Annu. Rep. Defense Commun. Agency, MIT Lincoln Lab., Sept. 1976.

26. Friedman E. and Ziegler C.
Packet voice communications over PC-based local area networks,
IEEE J. on Select. Areas in Commun., Vol.7, No.2, pp.211-218, Feb. 1989.

27. Garai V., Szabo C. and Szep T.
New multiple access algorithms for integrating voice and data on local area networks,
Problems of Control and Inf. Theory, Vol.17, No.6, pp.381-396, 1988.

28. Gburzynski P. and Rudnicki P.
A better-than-token protocol with bouded packet delay time for Ethernet-type LANs,
IEEE Symp. on the Simulation of Comput. Networks, pp.110-117, Aug. 1987.

29. Gilbert E. N.
Capacity of a burst noise channel,
Bell Syst. Tech J., Vol.39, pp.1253-1265, Sep. 1960.

30. Glass A. M. and Brewster R. L.
Reservation with contention-based traffic demand assignment protocol for land mobile radio communications,
IEE 5th Int. Conf. on Mobile Radio and Personal Commun., Conf. pub. 315, pp.6-9, Dec. 1989.

31. Gold B.
Digital speech networks,
Proc. IEEE, Vol.65, pp.1636-1658, Nov. 1977.

32. Golding L. and Liune A.
RALAN - A radio local area network for voice and data communications,
IEEE Globecom, pp.1900-1904, 1987.

33. Gonsalves T. A.
Packet-voice communication on an Ethernet local computer network: an experimental study,
ACM Comput. Commun. Rev., Vol.13, No.2, pp.178-185, 1983.

34. Gonsalves T. A. and Tobagi F. A.
Comparative performance of voice/data local area networks,
IEEE J. on Select. Areas in Commun., Vol.7, No.5, pp.657-669, June 1989.

35. Goodman D. J., Jaffe O. G., Lockhart G. B. and Wong W. C.
Waveform substitution techniques for recovering missing speech segments in packet voice communication,
IEEE Trans. on Acoust., Speech, and Sig. Proc., Vol.ASSP-34, No.6, pp.1440-1448, Dec. 1986.

36. Goodman D. J., Valenzuela R. A., Gayliard K. T. and Ramamurthi B.
Packet reservation multiple access for local wireless communications,
IEEE Trans. on Commun., Vol.37, No.8, pp.885-890, Aug. 1989.

37. Grela-M'Poko B. and Hayes J. F.
Comparison of accessing techniques with priority,
Comput. Commun. (UK), Vol.13, No.3, pp.149-56, Apr. 1990.

38. Gruber J. G.
Delay related issues in integrated voice and data networks,
IEEE Trans. on Commun., Vol.COM-29, No.6, pp.786-800, June 1981.

39. Gruber J. G. and Le N. H.
Performance requirements for integrated voice/data networks,
IEEE J. on Select. Areas in Commun., Vol.SAC-1, pp.981-1005, Dec. 1983.

40. Gruber J. G. and Strawczynski L.
Subjective effects of variable delay and speech clipping in dynamically managed voice systems,
IEEE Trans. on Commun., Vol.COM-33, pp.801-808, Aug. 1985.

41. Hashemi S. H., Barghi H. and Bredeson J. G.
An efficient reservation system for local area networks based on CSMA/CD,
IEEE Pacific Rim Conf. on Commun., Comput. and Sig. Proc., PP.1-4, Jun. 1989.

42. Hansen F. and Meno F. I.
Mobile fading Rayleigh and lognormal superimposed,
IEEE Trans. on Veh. Tech., Vol.VT-26, No.4, pp.332-335, Nov. 1977.

43. Hon L. K. and Lee H. W.
Performance analysis of the CSMA/CD and movable-slot TDM protocols,
IEEE Int. Conf. on Commun., Vol.1, pp.575-579, June 1987.

44. Hon L. K. and Lee H. W.
Simulation studies of the movable-slot TDM protocol,
IEEE Proc. of Symp. on the Simulation of Comput. Networks, pp.239-46, 1987.

45. Ibe O. C.
Assessing ISDN performance for packet networks,
Data Commun., Vol.16, No.8, pp.147-152, July 1987.

46. IEEE 802.3 standard for local area networks: Carrier sense multiple access with collision detection (CSMA/CD), access and physical layer specifications,
Wiley-Interscience, New York, 1985.

47. IEEE 802.4 standard for local area networks: Token-passing bus, access method and physical layer specifications,
Wiley-Interscience, New York, 1985.

48. Jakes W. C.
Microwave mobile communications,
John Wiley & Sons, New York, 1974.

49. Jayant N. S. and Christensen S. W.
Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation proceedure,
IEEE Trans. on Commun., Vol.COM-29, pp.101-109, Feb. 1981.

50. Johnson D. H. and O'Leary G. C.
A local access network for packetised digital voice commuication,
IEEE Trans. on Commun., Vol.COM-29, No.5, pp.679-688, May 1981.

51. Keiser G. E.
Local area networks
McGraw-Hill, Computer science series, 1989.

52. Kim B. G.
Characterization of arrival statistics of multiplexed voice packets,
IEEE J. on Select. Areas in Commun., Vol.SAC-1, No.6, Dec. 1983.

53. Kim B.G. and Choi D.
A delay-based window access protocol in a CSMA/CD network
IEEE Proc. TENCON, Vol.3, pp.851-855, Aug. 1987.

54. Kim M. B. and Lin H. P.
Modelling and simulation of channel access protocols for integrated networks,
Proc. Symp. on the Simulation of Comput. Networks, pp.118-122, Aug. 1987.

55. Kleinrock L. and Tobagi F. A.
Packet switching in radio channels: Part I - Carrier Sense Multiple Access modes and their throughput-delay characteristics,
IEEE Trans. on Commun., Vol.COM-23, pp.1400-1416, Dec. 1975.

56. Kleinrock L.
Queueing Systems Vol. 1,
John Wiley & Sons, 1975.

57. Kleinrock L.
Queueing systems Vol. 2,
John Wiley & Sons, 1976.

58. Klemmer E. T.
Subjective evaluation of transmission delays in telephone conversations,
Bell Syst. Tech. J., Vol.46, pp.1141-1147, July 1967.

59. Kraimeche B., and Schwartz M.
Analysis of traffic control strategies in integrated service networks
IEEE Trans. on Commun., Vol.COM-33, No.10, pp.1085-1093, Oct. 1985.

60. Kurose J. F. and Schwartz M.
A family of window protocols for time constrained applications in CSMA networks,
IEEE Proc. INFOCOM, pp.405-413, April 1983.

61. Kurose J. F., Schwartz M. and Yemini Y.
Controlling window protocols for time constrained communication in a multiple access environment,
IEEE 8th Int. Data Commun. Symp., pp.75-84, Oct. 1983.

62. Kurose J. F. and Mouftah H. T.
Computer-aided modelling, analysis and design of communication networks,
IEEE J. on Select. Areas in Commun., Vol.6, No.1, pp.130-145, Jan. 1988.

63. Kurose J. F., Schwartz M. and Yemini Y.
Controlling window protocols for time-constrained communication in multiple access networks,
IEEE Trans. on Commun., Vol.36, No.1, pp.41-49, Jan. 1988.

64. Lea C. T. and Meditch J. S.
A channel access protocol for integrated voice/data applications,
IEEE J. on Select. Areas in Commun., Vol.SAC-5, pp.939-946, July 1987.

65. Lee W. C. Y.
Mobile communications engineering,
McGraw-Hill, New York, 1982.

66. Lee W. C. Y.
Mobile cellular telecommunications systems,
McGraw-Hill, New York, 1989.

67. Li S. and Majithia J. C.
Performance analysis of a DTDMA local area network for voice and data,
Comput. Networks, Vol.8, pp.81-91, 1984.

68. Lin S.
An introduction to error-correcting codes,
Prentice-Hall Inc., U.S.A., 1970.

69. Lin S., Costello D. J. and Miller M. J.
Automatic repeat request eeror-control schemes,
IEEE Commun. Magazine, Vol.22, pp.5-17, Dec. 1984.

70. Loshe E.
The role of the ISO in telecommunications and information systems standardisation,
IEEE Commun. Magazine, pp.18-24, Jan. 1985.

71. MacDonald V. H.
Advanced mobile phone services: the cellular concept,
Bell Syst. Tech. J., Vol.58, No.1, pp.14-41, Jan. 1979.

72. Mahmoud S. A., Wai-Yip C, Riordon J. S. and Aidarous S. E.
An integrated voice/data system for VHF/UHF mobile radio,
IEEE J. on Select. Areas in Commun., Vol.SAC-1, No.6, pp.1098-1111, Dec. 1983.

73. Marini M.
Improved performance in local area network integrating voice and data,
Europ. Trans. Telecommun. Relat. Technol. (Italy), Vol.1, No.1, pp.43-8, Jan.-Feb. 1990.

74. Matsumoto Y., Takahashi Y. and Hasegawa T.
Performance comparison of two CSMA/CD systems with prioritised retransmission,
Electron. Commun. Jpn. 1, Commun. (USA), Vol.72, No.9, pp.1-13, Sep. 1989.

75. Maxemchuk N. F.
A variation on CSMA/CD that yields movable TDM slots in integrated voice/data local networks,
Bell Syst. Tech. J., Vol.61, No.7, pp.1527-1550, Sep. 1982.

76. Meditch J. S. and Lea C. T.
Stability and throughput in Virtual Time CSMA,
Comput. Networks ISDN, Vol. system 10, No.1, pp.19-26, Aug. 1985.

77. Mercado J. De, Hafez H. M. and DaSilva J. S.
Packet error rates for fading land mobile data channels,
INTELCOM`80 4th Int. Conf. Creating the Future Today, pp.131-136, May 1980.

78. Metcalf R. M. and Boggs D. R.
Ethernet: Distributed packet switching for local computer networks,
Commun. Ass. Comput. Mach., Vol.19, No.7, pp.395-404, July 1976.

79. Michelson A. M. and Levesque A. H.
Error control techniques for digital communication,
John Wiley & Sons, U.S.A., 1985.

80. Mier E. E.
Tying together telephones and typewriters,
Data Commun., Vol.9, pp.51-67, April 1980.

81. Minoli D.
Issues in packet voice communication,
Proc. IEE, Vol.126, No.8, pp.729-740, Aug. 1979.

82. Mittal K. K.
An efficient contention based multiple access scheme for integrated voice and data networks,
IEEE Int. Conf. on Commun., pp.950-956, June 1983.

83. Molle M. L.
Unifications and extentions of the multiple access communication problem,
PhD. dissertation, Dep. Comput. Sci. CSD Rep. 810730 (UCLA-Eng.-8118),
Univ. California, Los Angeles, July 1981.

84. Molle M. L. and Kleinrock L.
Virtual time CSMA: Why two clocks are better than one,
IEEE Trans. on Commun., Vol.COM-33, pp.919-933, Sep. 1985.

85. Montgomery W.
Techniques for packet voice synchronisation,
IEEE J. on Select. Areas in Commun., Vol.SAC-1, pp.1022-1028, Dec. 1983.

86. Munoz-Rodriguez D. and Cattermole K. W.
Multiple criteria for hand-off in cellular mobile radio,
IEE Proc., Vol.134, Pt.F, No.1, pp.85-88, Feb. 1987.

87. Murase A. and Imamura K.
Idle-signal casting multiple access with collision detection (ICMA-CD) for land mobile radio,
IEEE Trans. on Veh. Tech., Vol.VT-36, No.2, pp.45-50, May 1987.

88. Murase A. and Imamura K.
Idle-signal casting multiple access with data slot reservation (ICMA-DR) for packet radio communication,
IEEE Trans. on Veh. Tech., Vol.38, No.2, pp.50-54, May 1989.

89. Musser J. M., Liu T. T. and Boggs G. J.
A local area network as a telephone local subsriber loop,
IEEE J. on Select. Areas in Commun., Vol.SAC-1, pp.1046-1053, 1983.

90. Nelson R. and Kleinrock L.
Rude-CSMA: a multihop access protocol,
IEEE Trans. on Commun., Vol.COM-33, No.8, pp.785-799, Aug. 1985.

91. Nutt G. J. and Bayer D. L.
Performance of CSMA/CD networks under combined voice and data loads,
IEEE Trans. on Commun., Vol.COM-30, No.1, pp.6-11, Jan.1982.

92. Paradells J.
A simulation study of local area networks under combined voice and data traffic,
8th European Conf. on Electrotechnics, Stockholm, pp.247-251, June 1988.

93. Parsons J. D. and Gardiner J. G.
Mobile communication systems,
Blackie & Son, Glasgow, 1989.

94. Petr D. W., DaSilva L. A. and Frost V. S.
Priority discarding of speech in integrated packet networks,
IEEE J. on Select. Areas in Commun., Vol.7, No.5, pp.644-656, June 1989.

95. Pickholtz R. L. (ed.)
Local area and multiple access networks,
Computer Science Press, 1986.

96. Pritsker A. A. B.
Introduction to simulation and SLAM II, 3rd. ed.,
John Wiley & Sons, New York, 1986.

97. Prycker M. D.
Evolution from ISDN to BISDN: A logical step towards ATM,
Comput. Commun., Vol.12, No.3, pp.141-146, Jun. 1989.

98. Rozanski G.
Integrated services local-area networks. I: Performance evaluation of medium-access control protocols,
Electron. Telecommun. Lett. (Poland), Vol.4, No.1, pp.17-24, Mar. 1989.

99. Shanmugan K. S.
An update on software packages for simulation of computer systems (links),
IEEE J. on Select. Areas in Commun., Vol.6, No.1, pp.5-12, Jan. 1988.

100. Sharrock S. M.
A framed movable-boundary protocol for integrated voice/data in a LAN,
Proc. ACM Sigcom Symp: Commun., Arch. and Protocols, pp.111-119, Aug. 1986.

101. Sheikh A. U. H. and Hafez H. M.
Digital transmission over mobile channels: measurements and evaluation,
IEEE 36th Veh. Tech. Conf., Dallas, pp.46-51, May 1986.

102. Sklar, B.
Digital communications,
Prentice-Hall, New Jersey, 1988.

103. Stallings W.
Data and computer communications,
Macmillan, (New York), 2nd ed., 1988.

104. Suda T.
Performance evaluation of a packetised voice system - simulation study,
IEEE Trans. on Commun., Vol.COM-32, No.1, pp.97-102, Jan. 1984.

105. Tafazolli R., Richharia M. and Evans B. G.
A flexible multiple access scheme for mobile communications,
IEE Colloq."University research in mobile radio", Digest 1990/143, Nov.1990.

106. Thrower K. R.
Mobile radio possibilities,
J. IERE, Vol.57, No.1, pp.1-11, Jan./Feb. 1987.

107. Tobagi F. A. and Kleinrock L.
Packet switching in radio channels: part II- The hidden terminal problem in carrier sense multiple access and the busy-tone solution,
IEEE Trans. on Commun. Vol.COM-23, No.12, pp.1417-1433, Dec. 1975.

108. Tobagi F. A. and Kleinrock L.
Packet switching in radio channels: Part III - Polling and (dynamic) split-channel reservation multiple access,
IEEE Trans. on Commun., Vol.COM-24, No.8, pp.832-844, Aug. 1976.

109. Tobagi F. A.
Analysis of a two-hop centralised packet radio network-part I: Slotted ALOHA,
IEEE Trans. on Commun., Vol.COM-28, No.2, pp.196-207, Feb. 1980.

110. Tobagi F. A.
Analysis of a two-hop centralised packet radio network-part II: Carrier sense multiple access,
IEEE Trans. on Commun., Vol.COM-28, No.2, pp.208-216, Feb. 1980.

111. Tobagi F. A. and Hunt V. B.
Performance analysis of Carrier Sense Multiple Access with Collision Detection,
Comput. Networks, Vol.4, pp.245-259, 1980.

112. Tokinaga S.
Analysis of demand-assignment schemes with adaptive reservation on bus-type LANs for integrated communications,
Electron. Commun. Jpn. 1, Commun. (USA), Vol.72, No.8, pp.1-10, Aug. 1989.

113. Turner J. S. and Wyatt L. F.
A packet network architecture for integrated services,
Global Telecomm. conf. record, San Diego, California, Vol.1, pp.2.1.1-2.1.6, Nov. 1983.

114. Turner J. S.
New directions in communications (or which way to the information age?),
IEEE Commun. Magazine, Vol.24, No.10, pp.8-15, Oct. 1986.

115. Turner J. S.
Design of an integrated services packet network,
IEEE J. on Select. Area in Commun. Vol. SAC-4, No. 8, pp.1373-1379, Nov. 1986.

116. Velduis R. N. J.
A method for the restoration of burst errors in speech signals,
3rd European Signal Process. Conf., The Hague, pp.403-406, Sep. 1986.

117. Weinstein C. J.
Fractional speech loss and talker activity model for TASI and for packet-switched speech,
IEEE Trans. on Commun., Vol.COM-26, No.8, pp.1253-1257, Aug. 1978.

118. Welch P.
The statistical analysis of simulation results,
Academic, New York, S. Lavenberg ed., 1983.

119. Willer M.
Making waves (packet radio),
BYTE (USA), Vol.14, No.13, pp.363-366, Dec. 1989.

120. Willet M. and Martin N. E.
LAN management in an IBM framework,
IEEE Network, Vol.2, No.2, pp.6-12, Mar. 1988.

121. Wizgall M., Weiss W. and Stier W.
The ISDN approach for mobile radio,
Int. Switching Symp., Stockholm, Vol.6, pp.161-167, June 1990.

122. Wolfberg N. E.
The Ethernet handbook, 2nd ed.,
North-Holland, Amsterdam, 1983.

123. Wu C. T. and Dhadesugoor V. R.
An adaptive multiple access protocol for integrated voice/data local area networks,
IEEE GLOBECOM'83, San Diego, Vol.2, pp.959-963, Dec. 1983.

124. Wu C. T. and Dhadesugoor R. V.
Performance analysis of a dual channel local area network (DC-LAN) for integrated voice and data transmission,
IEEE Int. Conf. on Commun., Vol.1, pp.585-590, June 1987.

125. Yong W. R.
Advanced mobile phone services: introduction, background, and objevtives,
Bell Syst. Tech. J., Vol.58, No.1, pp.1-14, Jan. 1979.

126. Zhang K. and Pahlavan K.
An integrated voice/data system for mobile indoor radio networks,
IEEE Trans. on Veh. Tech., Vol.39, No.1, pp.75-82, Feb. 1990.