Technical Report

UCAM-CL-TR-7
ISSN 1476-2986

Number 7

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

Local area computer communication networks

Andrew Hopper

April 1978

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500

http://www.cl.cam.ac.uk/

This technical report is based on a dissertation submitted April 1978 by the author for the degree of Doctor of Philosophy to the University of Cambridge, Trinity Hall.

Technical reports published by the University of Cambridge Computer Laboratory are freely available via the Internet:

http://www.cl.cam.ac.uk/techreports/

ISSN 1476-2986

PREFACE

I am indebted to my supervisor, Professor D.J. Wheeler, for initially introducing me to the subject of computer networks and for many interesting and wide-ranging discussions during my three years under his supervision.

I am also grateful to Professor M.V. Wilkes for advice and help given in numerous ways and for enabling me to visit the United States during the last year of my research which proved to be a very valuable and stimulating experience.

Dr J.W. Pitman of the Department of Pure Mathematics and Mathematical Statistics devoted much of his time to educating me in the complexities of queuing theory and Markov chains, for which I am grateful.

My thanks also go to Sheena Baptie for expertly typing the drafts and final copy of this thesis.

This research was funded by the Science Research Council.

* * * * * *

No part of this dissertation has been submitted for any other degree or diploma at any other institution.

The work described in this dissertation is to the best of my knowledge original, and was done by myself without collaboration. The early chapters contain a discussion of the background to the research in terms of the work done by others; but the criticism of this work is my own.

CONTENTS

CHAPTER 1

MOTIVATION AND AIMS

1.1 Introduction

In this thesis a number of local network architectures are studied,and the feasibility of a LSI design for a universal local network chip is considered. Local means within one or several buildings, or on one site. In this chapter the basic problems encountered in local network design are defined,and an overview of the thesis is given.

1.2 Technology

The development of computer systems has been greatly influenced by underlying shifts and developments in hardware technology. These two processes have not been independent: hardware is developed as there is a need for performance improvement in new systems, and new systems are designed to take advantage of hardware developments. In recent years these shifts have been tending towards design in LSI (Large Scale Integration) with up to 16K RAM (Random Access Memory) chips and microprocessors consisting of more than 2K gates.

A description of current hardware technology is outside the scope of this thesis,but a number of techniques can be mentioned as showing promise for the future. Of the MOS techniques (PMOS, NMOS, CMOS, SOS, DMOS) and bipolar techniques (DTL, TTL, ECL, MECL, I^2L), it seems that NMOS, TTL, and I^2L will compete in the future [FERRE77]. NMOS has moderate speed characteristics and a higher density (80 - 120 gates/mm^2) than I^2L and therefore a lower cost-per-gate. TTL, especially of the

low power Schottky variety, offers high speeds (5nsec/gate delay), is well known, and with improvements should stay competitive in the future. I^2L provides a compromise between bipolar speeds and MOS densities (I^2L density 100 - 200 gates/mm^2, speed 5 nsec/gate) and will probably be used in most areas with the exception of the very low-cost (PMOS) and very high-speed (ECL) applications. ECL is likely to flourish where very high performance is required (.5nsec/gate delay), for example in CPU design, and is likely to be developed in the form of uncommitted logic arrays (ULAs) so the preliminary manufacturing steps can be carried out before the final design is known.

The design of an LSI device has high initial costs. However, if these are amortised over a large production run, then the price of each unit becomes low. Thus, custom made LSI units have been developed for use in areas where the same device can be used at many points in the system. An area which only recently has been subject of study from this point of view is that of computer networking. Although developments in silicon technology over the past fifteen years have seen a threefold decrease in cost per gate, these developments have mostly benefitted CPU and memory hardware, while input/outpt and communications hardware have lagged behind. The time thus seems ripe for a move in this direction, especially now that communications technology has at last been moving forward with the advent of optical devices. These are used both as optical isolators, where the price has dropped to several dollars for units which operate a 2-3MHz, and as optical cables to replace conventional twisted pair or coaxial systems. The cost/performance improvement in communications technology by 1985 is forecast as twofold in performance and threefold in cost, giving a total improvement factor of six [HOB77].

1.3 Basic Elements

1.3.1 The spectrum of networking technology

Systems connecting a number of computers can be classified according to their memory/processor interconnection structure as shown in Fig. 1.1.

Network Type	Interconnection Structure
Global Computer Networks	Computers loosely coupled by links
Local Networks/Computer Modules	Some memory at processing units linked by high speed transmission lines
Multi-Processor Computers	Crossbar switch in the most general case.

Fig. 1.1 Computer Network Spectrum

Global computer networks grew as the need arose for communication between computers and remote terminals. In such loosely coupled systems the communicating devices can function as independent units and are connected by a network which can span a large area. As these systems developed, more and more computer-to-computer traffic resulted and sophisticated protocols were developed to enable connections to be set up quickly and automatically. These large systems improved the reliability and availability from the users point of view, but generally made inefficient use of the available computing power and were very costly. Examples are the Aloha net at the University of Hawaii [AB70],

which was originally developed to link a number of terminals to a central computer, and the ARPA network, which is a sophisticated and geographically distributed network linking machines of many different types [KL75b].

At the other end of the spectrum lie the multiprocessor computer systems where individual units are in close proximity to each other and share common memory. Such multiprocessor systems were initially developed to enable relatively inexpensive processors to share expensive peripherals such as discs. The software did not require alteration, except that the problem of simultaneity of access to the shared resource had to be solved. A concept which extended this was spooling (simultaneous peripheral operation on line). Such systems developed into the very tight multiprocessor configurations where each processor can access a common memory or set of memory modules through a crossbar switch. This can be considered as a network, although crossbar switches quickly become very complex when interconnecting large numbers of modules and thus are only useful for small numbers of processors and memory units. On the other hand, these tightly-coupled systems do satisfy some of the objectives of multiprocessing which are availability, adaptability, modularity, performance, and programmability. There has been a trend in recent years to improve reliability by decentralisation. Multiprocessor systems then fare well as they do not depend upon any particular memory module or processor and can be reconfigured at failure.

Between the two areas of remote computer networking and multi-processing lies the ill-defined area of local networking. It seems that a growing percentage of computer applications will gravitate to this kind of system because of its inherent advantages such as low cost, reliability, availability, adaptability to specific applications, and flexibility for growth. A number of local networks have been developed,

noteably the Ethernet [METC76] and DCS systems [FARB72] which will be described in more detail in the next chapter. Local networks tend to vary between systems which are designed to provide a communications capability for I/O sharing in a University or organisation, and others which attempt to provide more reliable and sophisticated computer systems for the user by dispersing the computer tasks over a number of distributed machines.

A scheme which attempts to totally decentralise the computing functions into a multiplicity of computing units and is best suited to applications where the load is highly variable is the Computer Module system built at Carnegie-Mellon University [FU73]. In such systems data has to be freely available to all modules,and thus all input/output is shared between processors. Such homogeneous processors and input/output controllers allow easy expansion and economic redundancy but require an efficient communications subsystem. These systems will be treated as part of the local network spectrum since the interaction between modules is not as critical as in a multiprocessor and could thus be implemented in a fast local network.

The differences between remote networks, local networks,and multiprocessors are summarised in Fig. 1.2.

	Global Network	Local Network	Multiprocessor
Separation	>10 Kilometers	10K-50 metres	<50 metres
Data rate	10-50 KHz	1 - 10 MHz	>10 MHz
Response time	1 sec	<.1 m.sec	.5 μsec
Port Architecture	asynchronous communication	shared address and interrupt	shared primary memory
Applications	terminal share load share file share	Distributed computing, high perf. fixed application, special by function	general purpose
Ability to grow	possible, but expensive	inexpensive, but algorithms do not exist	size and performance vary with cost and time

Fig. 1.2 Characteristics of Computer Networks

1.3.2 Characteristics of Local Networks

Local networking having been placed in the spectrum of network technology, the particular characteristics of local networks will be enumerated and compared with global networks to determine which factors make the two different. As local networking represents an intermediate area, it is difficult to define a precise set of criteria. The list that follows is the set of likely, rather than definitive, features a local network possesses.

(1) It is likely to have high bandwidth where the speed of the hardware does not represent a bottleneck. Various networks operating at speeds of up to 10 MHz have been proposed and built.

(2) It is likely to have a very simple connection topology so that routing problems are either non-existent or very simple.

(3) Messages in the local network have short life times, primarily as a result of the high data rates and low buffering requirements of such systems.

(4) Local networks tend to be optimised to the application. This is true even when the network is completely homogeneous and is constructed using one type of computer which is chosen for the application.

(5) Because the local network is cheap to interface to the outside world, the addition of extra nodes does not represent a great expense.

(6) There is little or no buffering in the communications subnet as local network nodes are simple and do not perform sophisticated control algorithms.

(7) The local network may be designed in such a way that there is a central controlling device. In a local network a central control node can communicate with other nodes quickly and efficiently and thus is economically feasible.

(8) Local networks tend to be inexpensive as often they are connecting simple devices, and thus high costs are not economically justifiable.

(9) Local networks are built around a very simple or non existent backbone network and communication can be directly at the host-to-host level.

The design goals of any network are to enable computers to share resources and to allow process to process communication. This is true whether the network is local or global. Furthermore, the network should have well defined operating characteristics, degradation properties, and should be incremental in nature. That is, it should be designed

in a modular fashion so that it can be expanded cheaply and easily. For local networks this can be achieved by designing simple, clean systems, where the communication and application functions are well partitioned and which behave deterministically under most conditions. The user requires a number of services from the network, which makes it necessary for a set of protocols to be defined. Such protocols exist whether or not the network is local; and although some optimisations can be made in the local case they are generally invariant for the two network types from the users-functionality point of view. Thus, to satisfy the user, the local systems will require most of the features of the global system (e.g. a layered protocol structure, virtual circuit connection, etc.). From the network designers point of view these protocols can be provided much more simply if the network is local. That is, routing and retransmission algorithms are simpler, as are some protocol time outs. For example, advantage can be taken of the specialised link types available and of communication at the host-to-host level. This means that a 16 bit parallel channel network can be devised with very high data rates and with no problems such as packets out of order or inadequate buffering at the network level.

One final requirement of both global and local networks is that they are capable of easily providing information about the traffic in the system. This can be achieved either by using a promiscuous node which looks at all passing traffic (as in Ethernet) or by special hardware units that trap erroneous packets. It is interesting to note that in the packet radio system as SRI as much money was spent on the monitor hardware as on the rest of the system.

Thus, most of the differences between local and global networks are due to physical size and are only governed by the environment of the network.

1.3.3 Users Requirements

From the users point of view the basic requirements of a network are that it should function in such a way that errors do not occur on transmission, that data arrives in the order it was transmitted, and that the user has some control over the movement of data within it. He also requires some guarantee of the security of data and an indication of the performance of the network under various loads. Thus the behaviour of the network should be specified under the following conditions.

(1) Performance of the network when it is working correctly and within its design limits. The user requires to know the delay in obtaining a service, consisting of the time to set up that service followed by the time to transmit the data. This includes delays due to buffering acknowledgement, etc.

(2) Performance of the network when it is not working correctly. The user requires to know what the response will be during degradation: which resources are given priority,and which are lost. He needs to know the response time during restoration and the mean time between failures when the network is working normally.

(3) Long term performance of the network. This includes both long-term data rate as opposed to that achievable in burst mode and the long-term failure characteristics.

1.4 Problems Encountered

1.4.1 Local network development

Numerous *ad hoc* techniques have been developed for interconnecting machines in a local network; however, for larger systems envisaged for

for the future a more rational approach to network design will have to be adopted. Some of the actions which must be taken before local networks become better understood and more widely used are listed below:

(1) Investigation of transmission problems in such networks. This includes hardware design,network design, protocol design,and error control.

(2) Development of cost effective LSI partitioning of such networks. This requires standards to be set to enable a chip or family of chips to be manufactured to perform some of the local network functions.

(3) Development of suitable control mechanisms. It is not clear to what extent control in a local network should be distributed or what kind of control algorithms should be used for optimum network performance.

(4) Determination of additional functions of local netoworks.

(5) Modularisation of such networks to enable the building of larger, perhaps global, systems.

1.4.2 Measures of Performance

The performance of a local network can be measured in terms of many factors. These include response time, throughput, delay, line utilisation, error rate, reliability, buffer requirements, processing requirements, behaviour at saturation,and error recovery time. Each network designer balances these parameters according to the application and it is difficult to compare different systems. However, the important parameters include cost, line utilisation,and reliability. These

are discussed below.

The cost of communication is a function of the distance between the communicating devices and the computing power required to enable the communication to take place. Local networks are less sensitive to the former of these costs than the latter. The network can be designed so that the host computer performs most of the computational functions of the network itself. This reduces cost but may require the host to perform sophisticated transmission algorithms. Alternatively, the local network can be designed along conventional lines with separate hardware at each node for network control functions. Such a network can be made completely transparent and autonomous to the user (the host being only aware of a data stream connection to another host) but is more expensive. Systems can be devised where the network not only performs transmission functions, but also some operating system functions. This enables a distributed computing system to be implemented more efficiently. These networks tend to be more sophisticated and expensive. The cost of a local network is thus influenced by the computational task it has to perform and more specifically by the point where the line between host computer functions and network functions is drawn.

Line utilisation is the proportion of time the transmission line is carrying useful data. This should be maximised, although on past experience, high line utilisations have not been easy to achieve. For global networks this figure has been rising from a level of about 10% for ARPANET to 60-70% for TELNET, but for local networks line utilisations are still low [KL76].

Local computer networks can improve reliability and availability from the users' point of view. Reliability is the probability of

completing a task that has already been started, and availability is the proportion of time that the system is available to the user and is related to response time. Reducing the mean time to repair to zero does not make the availability 100%. This is because it takes a finite time to set up a connection;and if the system breaks down, these connections are reinitialised and availability decreases. The reliability of a system decreases with the number of components in the system. This means that distributed computer systems seen as a whole are less reliable than centralised systems. However, from a users point of view, this is not true as most of the time he uses local resources and only occasionally requests a service from another node. As such services are replicated and are not dependent on any particular configuration, he is more likely to have his request granted. Thus,for the user the distributed system is more reliable.

1.4.3 Choice of Network Architecture

There are many choices available in the design of a network in terms of overall design and technology. Architectures are compared in terms of modulation (coding) schemes, the complexity of network interfaces, as well as measured performance parameters. The local network architectures include random access, broadcast, ring, loop, polling, direct-connection, circuit-switching,and shared-memory systems. Recently, the broadcast and ring schemes have been prominent,and these will be examined in more detail later. An interesting new technique is that of using optical systems. As fibre optic couplers have been developed and have become available at reasonable cost,such systems are now increasingly feasible. However, fibre optic systems are not always

suitable: for example, a broadcast system built around high-speed optical devices (200-300MHz) would have poor operating characteristics and require very long packets.

Different problems arise when the network is homogeneous and consists of identical units (e.g. Ethernet) and when it is constructed using dissimilar machines (e.g. DECNET, SNA). For non-homogeneous networks microprocessors can be used to perform some of the network-to-host interaction functions. However, as local networks tend to operate at high speeds, microprocessors can be a bottleneck and are not always suitable. When networks are homogeneous, addressing can be simplified by specifying network-wide address formats. Sets of such networks can be interconnected using devices which do not perform any protocol translation but only buffer the varying speeds of such networks. These devices can thus be simple and cheap.

Other network architecture issues which arise are what kind of error check facilities are provided and whether any special features such as a broadcast mode should be used. As local networks (with the exception of contention) can be made almost completely error free, the error check can be omitted in some cases. It is unwise to use a communication network without any error checking, but providing errors are rare, retransmission at a higher level is not inefficient. Some networks provide sophisticated error check facilities (Ethernet); and in some cases, the CRC check is used as a basic component in the packet structure (DCS). An important issue which has not been addressed by network designers is that of exploiting the broadcast facility of the bus and ring networks. Such a feature has been used to communicate between the source and one or two other nodes, but it is rarely used to communicate with all nodes in the system simultaneously. One of the

reasons for this is that the broadcast message can become corrupted, and additional protocols have to be defined to cater for this possibility.

The design of the local network has to be seen in the light of possible LSI implementation. In order to do this, the functions of a chip have to be defined, as well as the interfaces between the network and the host. Such a chip can be complex and require simple interfaces, or it can be simple and cheap with sophisticated interfaces to the communicating devices. Finally, the local network has to be capable of allowing logistic problems of network administration and protection of data by encryption to be resolved.

1.4.4. Simulation and Analysis

The performance of a local network can be measured by employing suitable simulation and modelling techniques. The analytical techniques include queuing theory analysis and Markov theory analysis. Such analytic models are generally some way removed from the real system due to simplifying assumptions and give only general results. On the other hand simulations tend to be more specific and thus give more application dependent results. Local networks will be studied using theoretical techniques in later chapters. The problems encountered and assumptions taken will be discussed in detail then.

1.5 Points of Investigation and Outline of Dissertation

In this dissertation the tradeoffs in implementing a local network using a number of architectures are investigated. This is done both by considering the performance characteristics and hardware requirements of the systems. The networks chosen for comparison

include ring and broadcast schemes. The ring systems consist of the empty slot system [HOP78], the permission token system [FARB73], the register insertion system [HAF74b], and a pre-allocated bandwidth system. The broadcast schemes consist of the Aloha system [AB70] and the Ethernet system [METC76]. LSI partitioning of the network hardware is considered, as well as the feasibility of a single, variable architecture, general-purpose,multi-network chip. Hardware support for protocol implementation is also examined.

Chapter 2 surveys local networks and gives a detailed description of the ring network at the Computer Laboratory, Cambridge. Chapter 3 compares ring systems,and Chapter 4 considers the stability and performance of broadcast systems. Chapter 5 presents a design and considers the feasibility of manufacture of a general purpose local network chip. In Chapter 6 the Cambridge ring is redesigned to provide hardware support for protocol implementation. Finally Chapter 7 presents some novel local network architectures incorporating simple packet routing mechanisms.

CHAPTER 2

LOCAL NETWORK SURVEY

2.1 Introduction

In this chapter a survey of local networks is presented. Initially, early systems are considered, followed by descriptions of a number of local networks in operation today. These include the University of California Irvine DCS, the Xerox Corporation Ethernet, as well as the ring system at the Computer Laboratory, Cambridge.

2.2 Ring networks

2.2.1 Early systems

An early example of a ring scheme is the IBM 2790 data communications system [STE70] where the data is transmitted to, or received from, a single loop supervisor. The 2790 system is based on fixed assignment of time-slots to channels. Each slot is assigned in turn to a different channel, the total data rate being 514.67 Kbits/sec. The primary use of the 2790 system is limited to data collection.

Another example of an early ring system is the permission token scheme proposed by Farmer and Newhall [FARM69]. Each terminal is allowed to transmit an arbitrary length message when in possession of a token. The token is placed at the end of the message to pass to the next node downstream. Such a scheme is particularly suited for sources of a bursty nature; however, only one source can transmit at a time. The token is implemented with the aid of a polarity violation modulation technique. A digit is transmitted as a pair of pulses with opposite polarities,and

a violation of this principle indicates start of message (SOM) or end of message (EOM). The single bit which follws the EOM is the token and is either taken off by the node which transmits its data or is passed on unaltered. A ring monitor (Honeywell 516) is used to detect when the token becomes corrupted or when more than one token exists.

Further work on rings was done by Pierce, who proposed that a ring be divided into a number of fixed size slots [PI72a]. Each slot is marked full by the transmitter and empty by the receiver. Such a scheme allows an arbitrary amount of parallelism but has the disadvantage that hogging may occur; and if messages are longer than the slot size elaborate disassembly, sequencing and reassembling facilities have to be provided. Pierce further proposed a hierarchy of rings with buffering devices transferring messages between neighbouring rings. A single supervisor monitors each ring to ensure it does not become blocked with undeliverable data. Pierce's system was implemented by Kropfl [KR72] and also by Cocker [CO72], who connected two laboratory computers (Honeywell 516) and achieved a maximum transmission rate of 50 Kbits/sec. To obtain simplicity at the gateway between rings, an addressing scheme with routing based on the Hamming distance between nodes has been developed by Graham [GR71].

A system where the controller is much more sophisticated and performs error control functions and coordinates terminals operating at different speeds is the Bell Laboratories Spider network [FRAS74a]. This network can handle up to 64 duplex data transmissions at the same time and makes use of two packet types. Large packets are used for data, and small packets for network control signals. The ring is divided into a number of slots, each slot being able to hold one packet of each type. The Spider network determines the route for all packets in a message in

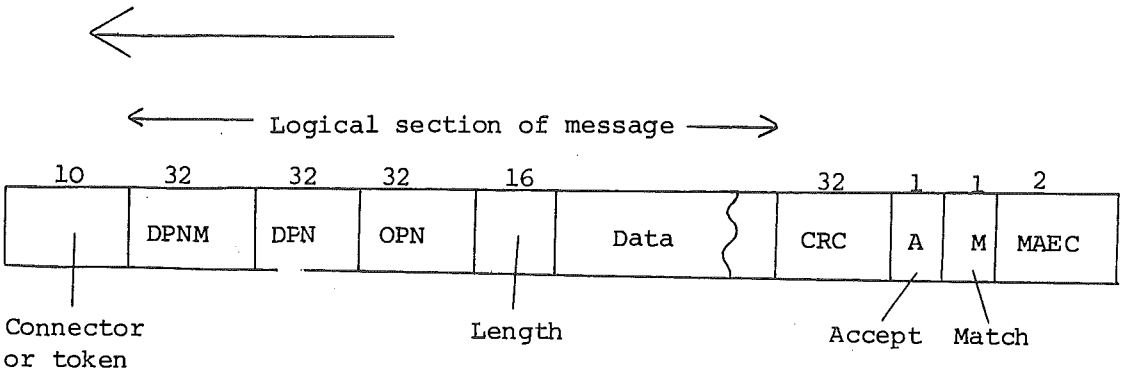
advance of transmission, which allows the number of bits in the address field to be reduced. This has the advantage that there are fewer special error and control states, but there is an overhead to changing the communication path.

A ring scheme has independently been proposed by Hafner and by Reames based on inserting a shift register into the communications path [HAF74b, RE75a]. The packet to be transmitted is placed in the shift register before being transmitted to the next station. A packet can thus be inserted between two other packets on the ring. This means that bandwidth is distributed evenly between all users, and that the delay round the ring is proportional to traffic. A number of techniques can be devised for removing a shift register from the ring, some of which allow the register to be used again before the previous packet has been removed.

2.2.2 University of California, Irvine Distributed Computer System (DCS)

The Irvine ring is used for communication between computers in a distributed computer system. The ring interface allows process-to-process communication by use of an associative name table at each node. A new design called the Local Network Interface (LNI) has been proposed, which incorporates additional address fields for masking purposes, and which can be implemented in LSI [MO77].

The LNI connects a host to a communications medium and allows some changes in message format and transmission protocol. It is to be implemented on a single LSI chip, supported by line drivers and power supplies. The final design speed for the LNI is 1-5 MHz.



DPNM - Destination Process Name Mask

DPN - Destination Process Name

OPN - Originating Process Name

CRC - Cyclic Redundancy Check

MAEC - Match/Accept Error Check

Fig. 2.1 DCS Packet Structure

The structure of a DCS packet is shown in Fig. 2.1. Each LNI has a Name Table (NT) with two fields: the name field and the mask field. Each of these fields is 32 bits long,and the NT can hold a variable number of entries. As a message passes an LNI,all entries in the NT are compared in parallel with the address fields of the message. If one of the entries in the NT corresponds to the name in the message, an attempt is made to copy the data. The mask fields can be used to override components of the name, which can then be subdivided into a number of fields. There are two control bits: the Match bit (M) and the Accept bit (A). These control bits are set by an OR operation at each LNI. The logical section of the message is protected by a CRC check, and there is a separate error check field for the MA bits (MAEC). As the MAEC field is typically much shorter than the CRC field,the delay through the LNI is minimised.

The transmission system allows only one LNI to add a message to the ring at a time. This is achieved by using a unique permission token (eight ones), which is used in the same way as in Farmers scheme [FARM69]. If no token arrives before a time-out takes place, a new token is generated. If more than one LNI times-out at the same time, several new tokens are created. These will either destroy themselves by overwriting, or two messages and two tokens will be output onto the ring. In the latter case, the LNI will notice that the CRC of the message following the token (which is the way it identifies its own message) is different to the one it originally transmitted and delete both messages. As the time-outs are set to different values, the ring recovers. The LNI is designed as a finite state machine built around a number of programmed logic arrays (PLAs). There are 18 byte input and output buffers, and the delay through each LNI is about 1 microsec. The maximum distance between LNIs is 1Km, and the ring is optically couped to each LNI.

The Irvine ring provides very powerful addressing facilities, and thus allows efficient process to process communication. This is achieved at the expense of additional hardware in the LNI and of additional control fields in the packet. Such a system is not suited for applications where the network only provides a communication mechanism. Also, the scheme for token regeneration is complex; and as maximum packet length is 64 Kbits, the effects of hogging may reappear.

2.2.3 University of Cambridge, Ring project

A system which the author has been concerned with is the ring project at the University of Cambridge. The data ring at Cambridge was designed to provide a high-speed, low-error rate communications path between computers and other devices in the Computer Laboratory.

These devices are connected through the ring on an individual basis , and as yet there are no global high level protocols to provide automatic call establishment. The primary uses of the ring are for equipment sharing and file dumping [WI75, HOP78].

2.2.3.1 Ring Organisation

The original design was based on the register insertion principle, but in due course it was realised that a more attractive system would be one based on the empty slot principle. In its simple form the empty slot system suffers from hogging. This defect can be overcome if each packet makes a complete revolution of the ring and is not marked empty until it has passed the original source. With this scheme the interaction with the ring at each node is minimised and reliability is improved. As performance characteristics of the register insertion and empty slot systems are very similar, the latter was adopted as the basis for the Cambridge ring.

The structure of the ring is shown in Fig. 2.2. Repeaters are used to regenerate the signal at each node and can operate autonomously from the stations which perform the logic functions for transmission and reception of packets. Each station is interfaced to its host via a specially built access box. The access box tends to be sophisticated for a simple device such as a line printer and simple for an intelligent device such as a computer which can perform most of the required logic functions internally. There is a unique station called the monitor station,which is used for setting up the slot structure during turn-on, for monitoring the ring and clearing lost packets, and for accumulating error statistics.

The packet structure is shown in Fig. 2.3 and is chosen to allow the maximum timing tolerance and minimum delay at the transmitter and

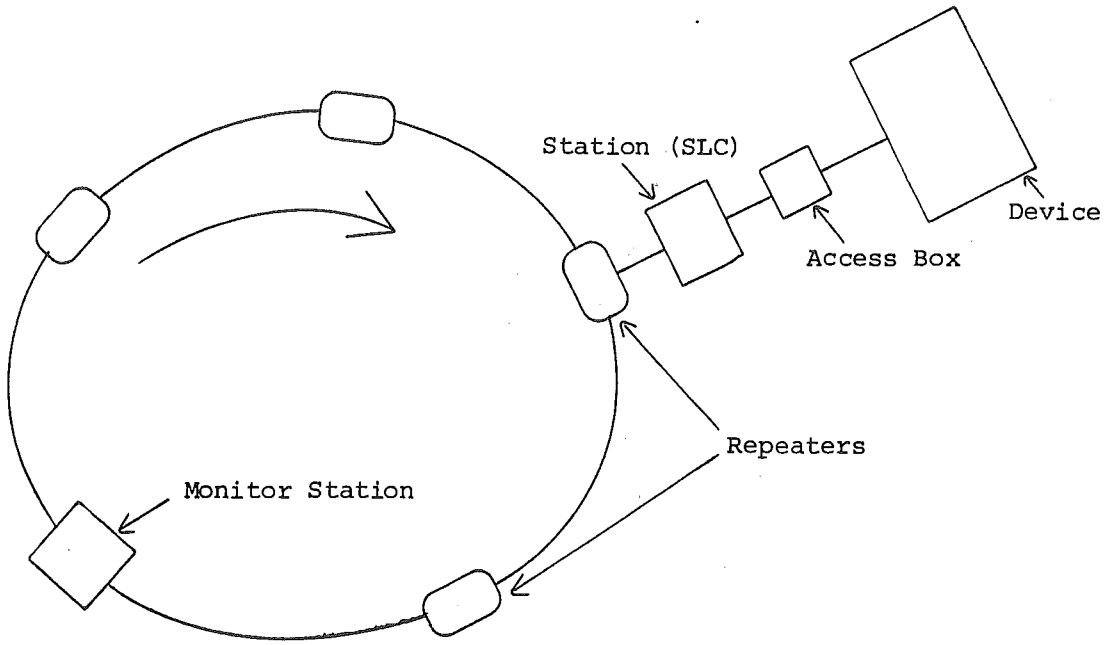


Fig. 2.2 Ring Structure

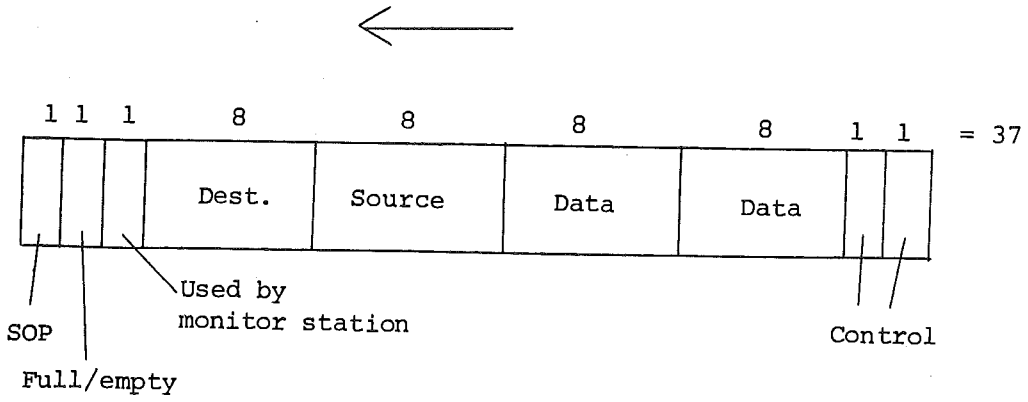


Fig. 2.3 Packet Format

receiver. The leading bit is always a one, and is followed by a bit to indicate whether the slot is full or empty. Now follows a control bit used by the monitor station to mark as empty packets which are circulating indefinitely due to an error in the full/empty bit. This is followed by 4 eight-bit bytes, the first two of which are used for destination and source addresses and the last two for data. Finally, there are two control bits used for acknowledgement purposes.

When a station has a packet ready for transmission in its shift register, it waits until the beginning of the next slot. It now reads the full/empty bit and at the same time writes a one at the output. If the full/empty bit was a zero it transmits the packet. If, however, the full/empty bit was a one, the slot is already occupied and the algorithm is repeated at the following slot. This scheme minimises the delay at each node, which can be less than one bit time.

The transmitted packet makes its way to the destination, where the control bits are set on the fly to indicate accepted, busy, or rejected. It now returns to the source where the slot is marked empty. If the packet returns with the control bits unchanged, it was not recognised by any destination. Each station automatically computes the total number of slots in the ring and can thus clear the full/empty bit immediately.

It can thus be seen that on transmission the packet is delayed until an empty slot is found, but then the transmission is rapid. This is in contrast to the register insertion scheme where the delay round the ring can be large, but the initial delay before the register is inserted is small. As the destination does not explicitly signal when it is ready to receive the next packet, the ring can become clogged with packets returning marked busy if devices with varying speed characteristics are being interconnected. In order to overcome this the following algorithm

is incorporated in the hardware to delay responses other than accepted. If a source transmits the same packet twice, and both times it returns marked busy, then it is not allowed to retransmit it again until some time later. This additional delay is 2 x ring delay x traffic density for the first retransmission, and 15 x ring delay x traffic density for succeeding tries. Thus the number of busies is decreased and performance is improved. As the data field is short, line utilisation may be poor; however, multi packet messages can be received asynchronously.

Each station possesses a station select register which is initialised by the host. This register can be set to accept or reject all packets addressed to it, or to receive from one source only. When combined with a time-out mechanism, it can be used to allocate resources on the ring.

2.2.3.2 Error Recovery

There are no CRC or parity checks on the transmitted packet, however, a copy of the information is retained at the source and is compared with the returning packet. This provides a powerful error detection facility but does not indicate that the packet was correctly copied at the destination.

If one of the SOP bits is corrupted or the full/empty bit becomes full, then this will be detected and corrected by the monitor station. If full becomes empty, then the packet might be ignored at the destination, but this will be detected by the source. Similarly, the transmitter will detect if the monitor station bit becomes corrupted in such a way that the slot is marked empty. An error in the address fields may cause the packet to be delivered incorrectly or be assigned to the wrong source. An error in the response bits may have a more serious effect as it will not be detected by the transmitter, which might

repeat the packet or assume it was received correctly when this was not the case. Under some circumstances such errors can propagate, but generally they are detected by the source or monitor station within one ring delay. Unnoticed errors are rare.

Each station continually determines the slot count by using the gap digits and packet leader digits. The gap digits are set to zero, and at least one such digit must be present in the system.

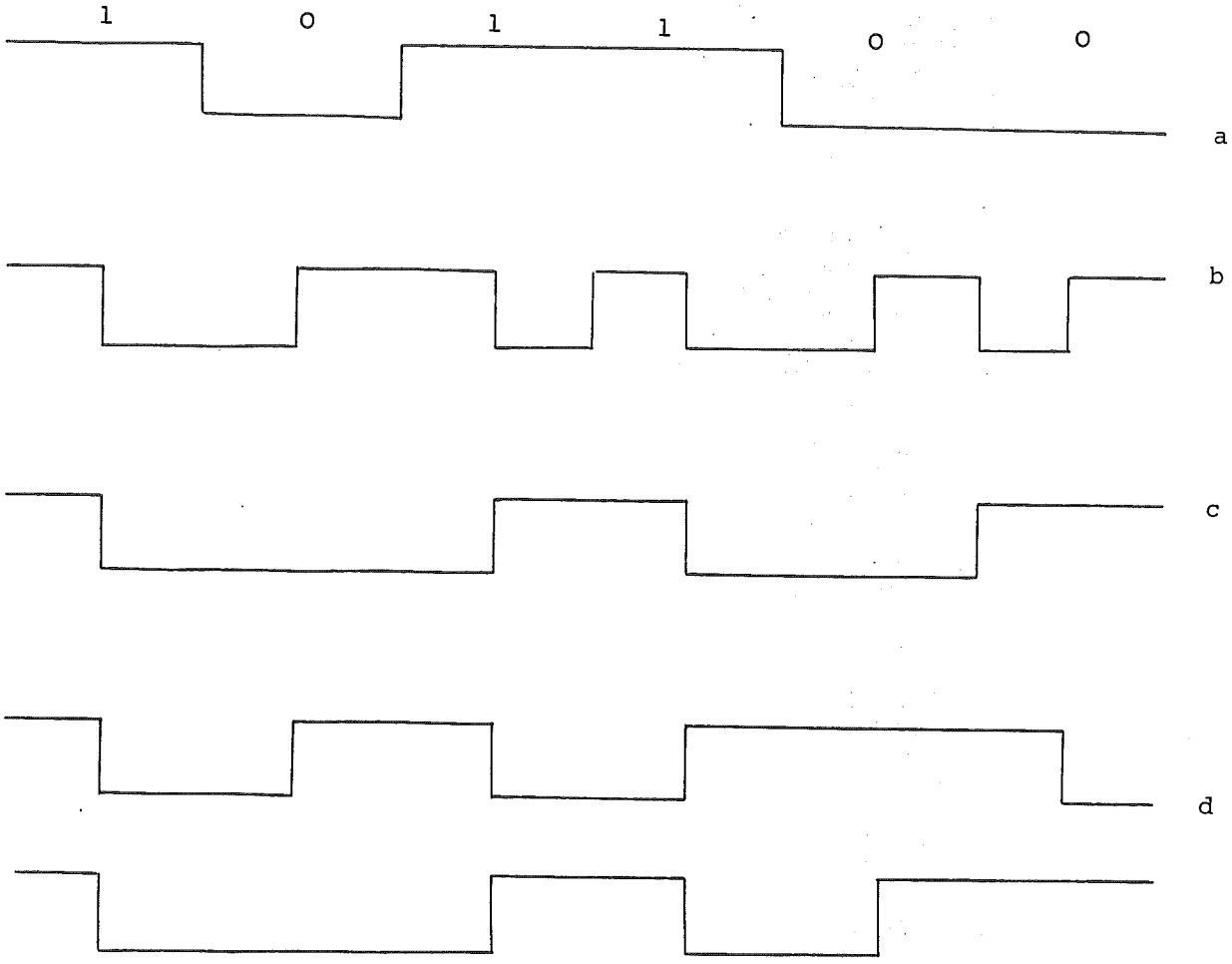
Additional error detection facilities are provided by the monitor station which can issue test packets, store erroneous ones, and provoke a response from any station. This response is independent of the returning data so that such a test can be carried out when the ring is broken.

2.2.3.3 Hardware

The ring is built using TTL technology and operates at 10 MHz, with a maximum distance of 200 meters between repeaters. Higher rates would be readily attainable with faster logic. The signals are transmitted along twisted pairs of the type normally used for duplex operation of teletypes. Transformers are used throughout for isolation and common mode rejection. As the repeaters have to operate reliably, whether they are connected to a station or not, they are powered directly from the ring. This power is injected into the system by a number of power units.

Each station is fully duplex so it can transmit and receive concurrently and independently. The number of bits delay at a station is a fraction of a bit, and the minimum ring delay is about 4 microseconds.

A number of modulation techniques were considered, and some of these are shown in Fig. 2.4. The four wire scheme was chosen as it is suitable for a pair of twisted wires and has no ambiguity about the



a. pulse for 1 no pulse for 0

b. phase modulation

c. p.m./2 (delay modulation)

d. four wire system

Fig. 2.4 Modulation Techniques

start of a digit (unlike phase modulation). A change on both pairs indicates a one and a change on only one pair indicates a zero, each pair being used alternately. The advantages of the four wire system can be summarised as follows:

(1) it is DC balanced

(2) there is very little encoding or decoding delay

(3) both pairs are treated identically thus minimising skew

(4) a choice of clock techniques is available

(5) it is easy to provide a control signal for a phase locked loop (PLL).

(6) it is easy to detect some errors (e.g. no change on either pair)

(7) it cannot move half a digit out of phase

(8) it is easy to transmit power without elaborate filters.

2.2.3.4 Discussion

The Cambridge ring was designed in an environment where many different types of machines exist and where the disruption to their operating systems has to be minimal. This differs significantly from systems where the designer has a free hand to develop host software according to his wishes, and especially from systems which connect a large number of identical machines. Furthermore, it was a design task to make the system as inexpensive as possible,and it was thus kept simple. Nevertheless, many options are left,and some of these are discussed below.

In a simple scheme each source transmits to only one destination at a time. This can be extended to allow the multiplexing of packets to different destinations. If this is done, an algorithm has to be developed for matching the speed of transmission and reception and to ensure no sources are continually blocked. For devices with similar speed

characteristics this can easily be done by employing a round-robin scheme. Where the communicating devices operate at different data rates, a speed number could be associated with each destination. This number is updated when the delay does not match the estimate. Other algorithms have been developed which achieve this in different ways.

Under some circumstances the services of a particular node might be required by a number of stations at the same time. In the Cambridge system such requests are arbitrated on a random basis,and thus some sources experience additional delay before successfully transmitting. Where the application demands more precise performance characteristics, such requests should be queued.

2.2.4 Ring System Conclusions

Rings allow a number of users to demand share a single communications link. As one of the most important parameters in network design is delay, the rings are normally operated at low loads. This has the advantage that most of the irregularities inherent at high traffic levels do not occur.

Rings allow easy implementation of distributed switching and control functions and are thus well suited for use in distributed computer systems. There are also economic advantages to ring systems: they can be designed to be completely modular and incremental in nature and thus do not require a large initial investment.

The reliability of a ring system poses a problem as the failure of any element will disable the entire network. This can be overcome by incorporating a parallel standby ring, and a number of such schemes have been proposed [ZA74]. Such techniques are not suitable for rings with a small number of stations as the probability of failure of the reconfiguration units is greater than the probability of their improving

network reliability. In any case, ring failures will often be less serious than a breakdown in a centralised system in terms of fault location and repair times.

2.3 Broadcast Networks

Broadcast networks are characterised by the fact that a number of nodes may attempt to transmit at the same time, which results in all transmissions being corrupted. Such networks have been developed primarily to exploit the broadcast and multi-access capabilities of radio channels, although they can be implemented using a conventional cable system. The radio broadcast channel can be designed using satellite or ground radio. In the former case the round trip delay is approximately 0.27 secs, and in the latter the propagation delay is much smaller (microseconds). For a satellite channel, by the time a packet reaches the receiver, the transmission is past history, and no control action can be taken. When the propagation delay between transmitter and receiver is small, the transmission can be aborted early on if this is desirable.

2.3.1 Aloha Systems

The broadcast network was first proposed by Abramson for interconnecting terminals to a central computer at the University of Hawaii using a UHF radio channel [AB70]. One channel is used for transmitting data to the computer, and another for transmitting acknowledgements back to the terminals. Each terminal transmits and stores one packet at a time and then initiates a time-out for the returning acknowledgement. The central computer monitors incoming traffic and can detect if a collision has taken place by using a suitable error check, in which case

it does not send the acknowledgement. If no acknowledgement is received at the terminal before the time-out, the packet is retransmitted. In order to avoid two terminals timing-out and colliding repeatedly the time-outs for each are set to different values. An alternative scheme is for each terminal to choose the retransmission interval from a random distribution. A terminal uses no part of the channel when it is idle, but can utilise all the bandwidth during a burst. However, the maximum theoretical channel utilisation is 18%, and it has been shown that the system is unstable at saturation traffic levls [FAY77].

An Aloha network with users transmitting at different power levels has been proposed, where if a collision takes place, the most powerful user transmits successfully [RO73]. This is achieved by employing the capture effect of receivers, a transmission being received correctly if it is sufficiently more powerful than the others. This results in a higher theoretical channel capacity.

In a pure Aloha system a transmitted packet of length T is vulnerable to collision for a time 2T. If channel time is divided into fixed length slots, then packets either collide and completely overlap or are transmitted successfully. Thus, the vulnerable period is reduced to T, and theoretical channel capacity is doubled. This technique is more complicated than the pure Aloha scheme as a global check for synchronisation of user packets has to be provided. The slotted Aloha scheme is unstable for high traffic inputs, and throughput goes to zero.

2.3.2 Carrier Sense Multiple Access (CSMA)

When the propagation delay between source and destination is small the CSMA system increases the theoretical channel capacity. Before transmission the channel is sensed, and if it is occupied the transmission

is deferred until some time later. If the channel is sensed idle, then the transmission proceeds and is vulnerable to interference for a time equal to the propagation delay between the two most distant points in the system. If a collision takes place, it is detected by both transmitters, and the packets are retransmitted later. Once a transmission has been established, it continues without interruption [METC76].

A number of protocols, some of which considerably increase throughput, have been proposed for the user's action after sensing the channel [KL75a]. However, like Aloha systems, the CSMA system is characterised by the throughput tending to zero for large values of channel traffic.

2.3.2.1 Xerox Corporation Ethernet

A local network based on the CSMA principle and built using coaxial cable as the transmission medium has been built at Xerox PARC. Ethernet interconnects a large number of small homogeneous ALTO computers, each with its own disc and VDU. These machines are linked by a number of cables which form a tree structure. A gateway computer is located at the root of the tree to perform routing and buffering functions. There exists only one path between any source-destination pair, and the cables can be extended from any point providing this rule is adhered to. In Ethernet the network specific hardware is simple, most of the communication task being performed in the microprogram of the ALTOs. Packets are restartable at the source, the absence of an acknowledgement causing the transmission to be repeated. This principle is carried on throughout the levels of the protocol structure so that if a discrepancy occurs at any level no action is taken as the appropriate procedure is carried out at the source. The reason for this approach is that the CSMA system is inherently error-prone, and it is up to the processes in the source and destination to take the necessary precautions to achieve

the desired degree of reliability.

A station is connected to the Ether by means of a tap and a transceiver. The tap is designed in such a way that it can be connected without disruption to the network, and the transceiver is designed to fail without polluting the Ether. The maximum cable length is 1 KM and the system operates at 3 MHz. It is capable of interconnecting up to 256 stations, of which about 125 are in use. Phase modulation is used for transmitting data; this allows the carrier to be easily detected as there is at least one pulse on the cable for each bit-time. A typical Ethernet packet is shown in Fig. 2.5.

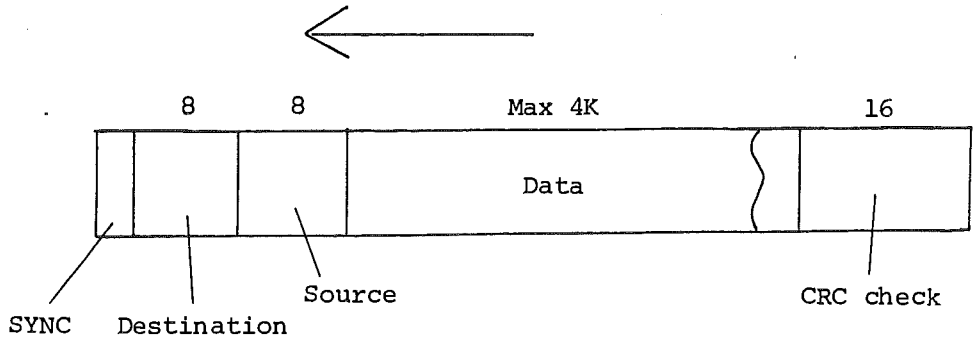


Fig. 2.5 Ethernet Packet Structure

The structure and functions of the hardware and software are divided into a number of levels. The lowest level encompasses the transceiver hardware and the error check generator. Above this lies the physical-link level which is concerned with retransmission, serialisation and deserialisation of data, speed matching, sequencing,and acknowledgements. This level is efficient and has few message formats. At the next level there is the logical-link software for multiplexing several user data streams into a single data stream. It is at this level that internetwork protocols such as TCP are implemented. At the highest dialogue level messages are sent over a logical link between users ; TELNET and file-transfer protocols (FTP) are implemented, as is a mailing system. The high level protocols consume the greatest amount of computational time.

It takes 40-50 simultaneous disc transfers for the system to become significiantly loaded, and thus no special retransmission algorithms are implemented in hardware. However, the following algorithm is employed in microcode to increase Ether efficiency. Retransmission intervals are multiples of a slot, and every time a transmission attempt ends in a collision, the controller delays for an interval of random length, with a mean twice the previous interval. Under light load conditions the mean is close to one, and retransmissions are prompt. As the traffic increases a backlog of packets develops and the retransmission intervals increase, the channel efficiency remaining high.

In the Ethernet system the cost and complexity of the hardware is minimised by performing most of the transmission functions in microcode. This means that the transmission algorithms can be easily changed, but the attached computers must be sufficiently sophisticated and homogeneous to allow this. The transceiver provides a collision-detect signal, which can be up to 20 DB weaker than the transmission and has to

be attached to the Ether in such a way that no reflections are produced. A break in the Ether, or a mismatched tap, disable the system.

An attractive feature of the Ethernet system is the use of ALTO computers to locally fulfil most of the user's needs, with the exception of very big tasks and I/O. These are catered for by special server nodes which can thus be optimised for their tasks. As transmission speeds increase the Ethernet (and CSMA) systems will become less attractive, since by the time a collision is detected, a large part of the packet will have been transmitted.

2.4 Other local network proposals and developments

There are a number of broadcast techniques which resolve the instability problem under overload conditions [HEI76]. These fall into two categories: dynamic control procedures for Aloha type systems and channel reservation schemes. Dynamic control schemes require each user to take action to prevent channel saturation when the backlog of packets reaches a certain level. Reservation schemes are suitable for systems where each message is composed of several packets since the access request is made for complete messages.

Local network techniques have been used for linking very large machines. This generally entails the design of a sophisticated station which can communicate at data channel speeds. Examples are the Goddard Space Flight Center network which links four IBM360s, a CDC-7600, a Cray-1, and a number of smaller satellite machines and has station costs in the region of $25,000. Network Systems Corporation market a similarly priced system based on coax cable operating at 15 MHz which has been installed for fast peripheral (disc) sharing at a number of locations [NBS77].

At the other extreme a number of microprocessor based systems are being built. These can be very cheap, as in the Queen Mary College C-net [WES77], or suitable for applications where a large number of microprocessors are interconnected for control applications [DOW77]. An example of a process control application is the Ford CSMA network which monitors the operation of machines on an assembly line.

Other local networks include the Mitrenet system [HANK77], which has a network language for interprocess communication and synchronisation and is run on a single RF coaxial cable which also supports other networks, telephone and TV channels; another is a broadcast network of MIT.

CHAPTER 3

A PERFORMANCE COMPARISON OF RING COMMUNICATION NETWORKS

3.1 Introduction

In this chapter a number of models for ring communication networks are developed. These will be used to provide a better insight into the way the networks behave and to compare their performance characteristics. This is done by describing the communications system in terms of a network of queues. The models describe both the functions of the logical units of the networks and the different transmission protocols, as well as measure the average delay and traffic handling capabilities of the networks. The analysis takes into account the storage capacity (or delay) at each node, scheduling restrictions, and retransmissions due to errors.

The networks compared are the register insertion system, the empty slot system and the permission token system. A further ring architecture, called the pre-allocated bandwidth system, is also evaluated, and its areas of application are outlined. These networks are implemented on the station logic chip described in Chapter 5.

3.2 Previous work

The main parameter of interest when studying a computer network is delay. Once this has been calculated, the throughput can be obtained either by explicit equations or by Little's theorem. The parameters which influence delay are:

(1) the station activity and how steady (homogeneous) it is

(2) the amount of buffer storage on the ring

(3) the use of protocols and structure of the ring

(4) errors and error recovery procedures

The delay experienced by a packet can be split into two components:

(1) the time before the packet is output onto the ring

(2) the delay around the ring itself

Other ring performance characteristics are the line utilisation (a function of data time, busy time, and idle time), and the effects of clustering, of the monitor station, of special packets, and of a complete ring break. Some of the above properties are best studied by use of a simulation rather than an analytical model as they are too complex to handle analytically. However, a sufficiently complicated model for the ring can be developed to show the relative importance of all major parameters.

3.2.1 Previous work on ring systems

Previous queuing theory work on ring systems can be divided into two categories :

(1) the analysis of the general cyclic queue problem

(2) the direct analysis of ring communication systems

The problem of a cyclic queue was first studied by Koenigsberg [KO58]. In his system a fixed number of customers requested service at a number of service centres in cyclic order with no external arrivals or departures. When the service times for the servers are equal, the probability of any system state can be obtained by combinatorial methods. When each service center has a different service-time the solution is more complicated, and Koenigsberg calculated the analytical formulae for up to five service centers. The number of customers queued at a center was found, along with the delay at the center, the mean cycle time, and the probability that a stage is idle.

Finch [FIN59] extended these ideas to include cyclic queues with

feedback. He found unique solutions for two types of cyclic queue discipline. In both cases an external source feeds one node with customers. In the first system customers move to the next node on completion of service, except for one node from which they can move to any node. In the second system, as well as moving to the next node, customers can return to the queue from which they have just departed. Both Koenigsberg and Finch use the product solution to a birth-death queuing system as a starting point for their solutions. This leads to calculation of the marginal probability that there are n customers at the j'th stage, the average number of customers at the j'th stage, and the average number of customers waiting. Finch also points out that in the case of a single server, the two types of feedback are the same, and that it is then possible to treat the case of random arrivals and general service time in a manner similar to that of a M/M/1/K queue.

Cyclic queues with restricted queue lengths were studied by Gordon and Newell [GO67a]. They showed that the closed cyclic systems considered were equivalent to open systems in which the number of customers is a random variable. A two-stage system was studied, the stochastic equations being comparable to those of a finite-capacity, single server queue. Next, systems with small and large number of customers, were considered. Results show that for a system with few customers the restricted queue length at each node has little effect, and that with many customers blocking dominates, and has a similar effect to queuing.

The above results are primarily applicable to the register insertion system (and to simple store and forward networks). A parallel can be drawn between service time and the delay introduced at each node by the variable length shift register. However, this does

not emulate the real system well since the shift register is of finite length, and the analysis is difficult except in special cases.

In other ring networks the delay round the ring is deterministic and is proportional to its length, the queue forming at the input (Fig. 3.1). The models developed in this chapter treat the rings in this way, the register insertion system being shown to be logically equivalent.

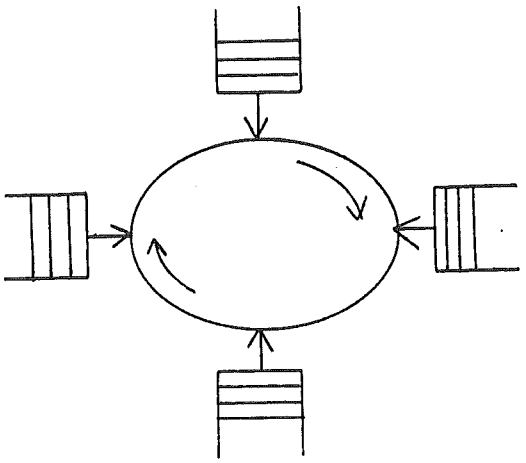


Fig. 3.1 Queue Structure for Ring Network

Hayes and Sherman studied the traffic behaviour of a Pierce loop system (i.e. a system with fixed-size slots circling round a loop and with no hog prevention technique) [HAY71]. They were interested in the effect of buffering on message delay, and the model incorporates an exponential on-off input process to take into account the bursty nature of data sources. The exponential on-off source is approximated in two ways: the first, by assuming messages with exponentially distributed length arrive at Poisson rate, and the second, by assuming that the source outputs at a constant rate equal to the average number of bits/sec

output by the exponential on-off model. The first of these approximations is suitable for sources that are not very active, the second being better suited to active ones. Using this model mean line idle and busy times were obtained, leading to the calculation of throughput and delay.

Another paper by Hayes and Sherman [HAY72] is concerned with the study of data multiplexing techniques for user populations whose source characteristics are of inquiry/response type. Thus ,source models for users, as well as for computer responses, are developed which approximate an inquiry/response context. The three systems compared are polling, random access,and Pierce loop. It is found that only the polling system is sensitive to synchronisation delay,which takes place every time a user station transmits to the central facility. The three schemes have similar performance characteristics, random access showing slightly lower delays than the other two.

Kaye [KAY72] studied a loop similar to the token system and gave examples with no propagation delays. He showed that for low loads,the probability of a message arriving before the previous one has been processed is low and thus developed approximations.

3.2.2 Input buffer structures

In this chapter two kinds of input buffer structures are modelled. In the first it is assumed that the input buffer at each node is of infinite length, and that all arriving packets are stored and transmitted in due course. When the arrival rate equals the service rate,the number of packets waiting for transmission is infinite. In the other buffer structure buffer size is limited to one packet. Any packets arriving while the buffer is full are turned away and do not effect the arrival

process. Thus, the arrival rate must always exceed the service rate.

3.3 The infinite size input buffer

A number of functions for the performance of the networks with an infinite size buffer at the input are derived below. In all cases the fixed electronic and cable delay around the ring is given by NBe.

$$NBe = Nb_{is} + \sum_{i=1}^{N} b_{il}$$

where b_{is} is the delay in bits through station i

b_{il} is the electronic delay of the lines between station i and i+1

and N is the number of stations.

A summary of the notation used is given in the Appendix.

3.3.1 The register insertion system basic performance function

In this scheme each station can insert at most one packet into the ring. This packet makes its way to the destination and then back to the source where it is removed. Two cases for a subsequent transmission are considered: (a) when the next packet can be loaded and transmitted instantaneously, and (b) when this takes a finite number of bits delay E. Any bits stored in the lines when the system is idle are not utilised and form gap digits. Let us consider the delay from the time a packet arrives ready for transmission at the head of the input queue, to the time it arrives back at the source and is removed allowing another transmission to take place. This is the service delay of the ring and when divided by the ring speed, gives the service time of the ring (per packet).

Service delay can be divided into three components.

(1) The electronic delay of the lines and the transmission and reception of own packet

(2) The delay due to transmission only being permissable between other packets

(3) The delay due to the transmitted packet passing through other inserted registers

Let Tx be the mean arrival rate of bits to a node from the outside world and Tr be the speed of the ring, and let

$$z = \frac{Tx}{Tr}$$

and p(z) represent the mean probability that a station has a packet available for transmission in its buffer (this packet is transmitted at the next opportunity).

The first register insertion system to be evaluated is that with instant replacement of old packets in the shift register. Type 1 delay is given by

$$NBe + Ps \qquad 3.1$$

where Ps is the packet size. If the packet to be transmitted arrives before the previous has been received back, no alignment delay is experienced. If, however, it arrives after the previous one has been processed, it is assumed that the mean value of this delay is Ps/2. The probability of encountering a packet at the moment of transmission is given by the ratio of number of packet bits on the ring to the total number of bits stored in the ring and is

given by

$$\frac{p(z)NPs}{p(z)NPs + NBe}$$

The probability that the packet arrives after the previous has been processed is 1-p(z), thus type 2 delay is given by

$$(1 - p(z)) \frac{Ps^2\ p(z)}{2(Ps\ p(z) + Be)} \qquad 3.2$$

Type 3 delay is governed by the number of other registers inserted into the ring multiplied by the delay in each and is given by

$$p(z)\ (N-1)\ Ps \qquad 3.3$$

Thus,the basic performance function for the register insertion system with instant replacement of packets is given by combining equations 3.1, 3.2,and 3.3:

$$B_D = NBe + Ps + \frac{(1-p(z))Ps^2\ p(z)}{2(Ps\ p(z) + Be)} + p(z)(N-1)Ps \qquad 3.4$$

If packets cannot be replaced in the transmission register instantly, type 2 delay is given by

$$\frac{Ps}{2} \frac{p(z)Ps}{Ps\ p(z) + Be}$$

as the probability of encountering another packet on transmission is never zero. Thus,the basic performance function for non instant replacement of packets becomes

$$B_D = NBe + Ps + \frac{p(z)Ps^2}{2(p(z)Ps + Be)} + p(z)(N-1)Ps + E \qquad 3.5$$

Due to the effect of E, the maximum value of p(z) is no longer 1 but is given by the ratio of time to output with instant replacement to time to output with instant replacement plus E:

$$p(z)^E_{max} = \frac{B_D}{B_D + E} \quad , \quad p(x) = 1$$

$$p(z)^E_{max} = \frac{2N(Ps + Be)^2}{2N(Ps + Be)^2 + 2E(Ps + Be)}$$

3.3.2 The empty slot system basic performance function

This model is based on the empty slot system built at Cambridge and described in section 2.1.3. Let Ga be the number of gap digits not used for transmitting data and Q be the number of slots.

$$Q = \frac{NBe - Ga}{Ps} \qquad Q \geq 1$$

The delay experienced by a packet between arriving at the transmitting shift register and being completely removed from the ring (service delay) consists of three components;

(1) the electronic delay of the lines and the time to fill and empty the slot

(2) the delay from the moment of arrival until the next full/empty bit

(3) the busy period before acquiring an empty slot

Type 1 delay is given by equation 3.1. It is assumed type 2 delay is on average Ps/2 plus the overhead due to gap digits. This delay is only experienced if the new packet arrives before the previous

one is processed and is given by

$$(1 - p(z)) \frac{Ps}{2} \frac{NBe}{(NBe - Ga)} \qquad 3.6$$

Type 3 delay is dependent on the number of other stations transmitting at any one time and carries an overhead due to gap digits. It is obtained by multiplying the proportion of full slots by the length of a cycle and the delay per full slot. This assumes the full slots tend to cluster which holds well for Q<N. Type 3 delay is given by

$$p(z) \ (N-1) \ \frac{NBe \ Ps}{(NBe - Ga)} \qquad 3.7$$

Thus the basic performance function for the empty slot system is obtained by combining equations 3.1, 3.6, and 3.7:

$$B_D = NBe + Ps + (1-p(z))\frac{Ps \ N \ Be}{NBe - Ga} + \frac{p(z)(N-1)Be \ Ps}{NBe - Ga} \qquad 3.8$$

3.3.3 The permission token system basic performance function

The delay due to the finite size of the token is disregarded. The service delay as previously defined again consists of three components :

(1) the fixed delay of the lines and the time to transmit and receive the packet

(2) the time to acquire the token due to its variable position at transmission request time

(3) the delay in acquiring the token due to other transmissions

Type 1 delay is given by equation 3.1. In an empty system the token is delayed on average by NBe/2 bits before arriving at the transmitting station. It is assumed that this is the average delay if a second

transmission is requested before the first one is processed. Thus, type 2 delay is given by

$$(1 - p(z)) \quad \frac{NBe}{2} \qquad 3.9$$

The delay to the token due to other stations transmitting (type 3) is the same as for the register insertion system and is given by equation 3.4. Thus, the basic performance function for the token system is obtained by combining 3.1, 3.4, and 3.9:

$$B_D = NBe + Ps + (1 - p(z)) \frac{NBe}{2} + p(z)(N - 1)Ps \qquad 3.10$$

As it is not logically necessary for the transmitted packet to be received back before the next transmission is attempted, a tandem queue model for the token system can be developed. Such two stage systems have been studied: however, as it is impossible for the token to be received before the transmitted packet, the results are the same.

3.3.4 The pre-allocated bandwidth system basic performance function

This system is included to illustrate the differences between systems where the bandwidth is pre-allocated: for example, where each node has its own slot and systems where a single station can utilise the available bandwidth almost completely. Fixed allocation systems require simpler hardware and have been implemented for linking peripheral devices to a central machine [STE70]. The two fixed allocation systems modelled are (a) where each node is in possession of its own slot and (b) where a single slot is shared between all nodes. In the latter

case the slot size and ring delay are constant. Such schemes work well when traffic is heavy and homogeneous but generally are very inflexible.

The service delay for the single slot system varies between

$$\frac{N(Be + Ps)}{2}$$

when traffic from the user is very low, and twice that when the user transmits at every available opportunity. This is due to the service delay being defined as the delay between the two points in time when the packet arrives and when it is completely serviced. The small Ps component at low traffic levels is ignored. Thus, the basic performance functions for the single slot pre-allocated bandwidth system are

$$B_D = (1 + p(z)) \left(\frac{N(Be + Ps)}{2} \right) \qquad Ps \geq NBe$$

$$B_D = (1 + p(z)) \left(\frac{NBe^2(N + 1)}{2} \right) \qquad Ps \leq NBe \qquad 3.11$$

It can be shown that the basic performance function for the pre-allocated bandwidth system with a single slot per station is given by

$$B_D = (1 + p(z)) \frac{NBe}{2}$$

3.3.5 Line Utilisation Equations

Line utilisation (S) is defined as the ratio of data traffic to total traffic. For the register insertion system the data on the

line is p(z) NPs, and the total length of the line is p(z) NPs + NBe. Thus, line utilisation for register insertion is given by

$$S = \frac{p(z)\ Ps}{p(z)\ Ps + Be} \qquad 3.12$$

For the slot system there are two cases. When the actual number of packets on the lines p(z)N is less than the number of slots Q, the utilisation is given by the data on the lines divided by the ring length plus the time to clear the packet (with its gap overhead). All packets are cleared in parallel. Thus ignoring the slight discontinuity when N and Q are not of the same parity, the utilisation is given by

$$S = \frac{p(z)\ NPs}{NBe + Ps + Ga/Q} = \frac{p(z)\ NPs\ (NBe - Ga)}{NBe\ (NBe - Ga + Ps)}, \quad p(z)N \leq Q \qquad 3.13$$

If $p(z)N \geq Q$, the minimum delay between successive transmissions is no longer one revolution (NBe). For N/Q revolutions the data transmitted is p(z)NPs. The total traffic is thus the data plus the associated gap overhead $G\frac{N}{Q}$ plus the overhead in passing on the slot $\frac{BeN}{Q}$. Thus utilisation is given

$$S = \frac{p(z)\ NPs}{p(z)\ NPs + \frac{p(z)\ Ga\ N}{Q} + \frac{BeN}{Q}} = \frac{p(z)\ (NBe - Ga)}{Be(p(z)N + 1)}, \quad p(z)N \geq Q \qquad 3.13$$

Note when $p(z) = \frac{Q}{N}$ the two equations are equal; and when $p(z) = \frac{Q}{N} = 1$ both reduce to $\frac{N}{N+1}$.

The token system is logically identical to the register insertion system, and the utilisation is given by equation 3.12.

The pre-allocated bandwidth system line utilisation is directly

proportional to traffic because the divesor of the utilisation equation is invariant with load. Thus, for the single slot system

$$S = \frac{p(z)\ Ps}{Be(N + 1)} \qquad Ps \leq NBe$$

$$= \frac{p(z)\ Ps}{Ps + Be} \qquad Ps \geq NBe \qquad 3.14$$

and for the slot per station system

$$S = \frac{p(z)\ NPs}{QPs + Ga} = \frac{p(z)\ Ps}{Be}$$

3.3.6 Delay equations

In order to calculate the total delay experienced by a packet, it is necessary to determine the value of $p(z)$ for each system. This is done by considering the line utilisation equations. For the register insertion system satuation point is reached when

$$S_{max} = \frac{Ps}{Ps + Be}$$

Thus, the maximum data rate into the system is

$$\sum Tx_{max} = \frac{Ps\ Tr}{Ps + Be}$$

and the maximum data rate per source is

$$Tx_{max} = \frac{Ps\ Tr}{N(Ps + Be)}$$

and

$$z_{max} = \frac{Ps}{N(Ps + Be)}$$

In order to calculate p(z) it is assumed that the probability of a station transmitting is directly proportional to the data rate into the system, and that when the system is saturated ,all stations are transmitting at every available opportunity. This assumption will be made in the same way for all systems so that their relative performance can be evaluated. Thus the probability of a station transmitting is given by

$$p(z) = \frac{\bar{n}}{N}$$

where $\bar{n}$ is the mean number of packets transmitted in one revolution. By the linear assumption, for the insertion system with instant replacement

$$\bar{n} = \frac{zN^2\ (Ps + Be)}{Ps}$$

$$\therefore \quad p(z) = \frac{zN(Ps + Be)}{Ps} \qquad 3.15$$

and for non instant replacement

$$p(z) = \frac{zN(p(z)^{E}_{max}\ Ps + Be)}{Ps} \qquad 3.16$$

By similar arguments it can be shown that the probability of transmission for the other systems is given by

for slot

$$p(z) = \frac{zNBe\ (NBe - Ga + Ps)}{Ps(NBe - Ga)} \qquad N \leq Q$$

$$p(z) = \frac{zNBe(N + 1)}{NBe - Ga} \qquad N \geq Q \qquad 3.17$$

for token

$$p(z) = \frac{zN(Ps + Be)}{Ps} \qquad 3.18$$

and for the pre-allocated system with one slot:

$$p(z) = \frac{zNBe\ (N + 1)}{Ps} \qquad Ps \leq NBe$$

$$p(z) = \frac{zN(Ps + Be)}{Ps} \qquad Ps \geq NBe$$

It is now possible to calculate the total delay experienced by a packet. In each of the systems this delay consists of two components: (a) the deterministic delay due to the fixed length of the transmission lines, and (b) the random delay due to other traffic. As the load increases, the coefficient of variation of the random component of delay changes and at saturation equals zero. Two sets of delay equations are developed, the first assuming that this random component has coefficient of variation zero, and the second assuming that the coefficient of variation is one. This defines an envelope for the delay equations within which the true curve lies. Each source is approximated by a Poisson source with a mean arrival rate of λ packets/sec.

The M/D/1 queuing system (coefficient of variation zero) is treated as a special case of the M/G/1 system. The server utilisation is given by

$$\rho = Bs\lambda \quad \text{where } Bs = \frac{B_D}{Tr}$$

$$\text{and } \lambda = \frac{zTr}{Ps}$$

The mean number in the queue at each node is given by

$$L_q = \frac{\rho^2}{2(1 - \rho)} = \frac{(Bs\lambda)^2}{2(1 - Bs\lambda)}$$

the steady state number in the system by

$$L_s = L_q + \rho = Bs\left(\frac{Bs\lambda^2}{2(1 - Bs\lambda)} + \lambda\right)$$

and the steady state time in the system by

$$D = \frac{L_s}{\lambda} = Bs\left(\frac{Bs\lambda}{2(1 - Bs\lambda)}\right) \qquad 3.19$$

Where the random component of the service time is assumed to be Poisson (coefficient of variation 1), the mean of the random components is denoted by $\frac{1}{\mu_1}$ (all terms in basic performance functions with p(z) components), and the constant value of the fixed component is denoted by $\frac{1}{\mu_2}$ (all other terms). Because the two service time components are independent the mean of their sum and first moments can be obtained directly by summation. Thus, by considering the M/G/1 queue, it can be shown that the mean steady-state time in the system is given by

$$D = \frac{2(\mu_1 + \mu_2)(\mu_1\mu_2 - \lambda(\mu_1 + \mu_2)) + \lambda(2\mu_1^2 + \mu_2^2)}{2\mu_1\mu_2(\mu_1\mu_2 - \lambda(\mu_1 + \mu_2))} \qquad 3.20$$

3.4 The single packet input buffer model

In this model new estimates for the probability of a station transmitting p(z) are developed. These are obtained directly since the input queue is restricted to a single packet. Packets which arrive while the buffer is full are lost.

3.4.1 The register insertion system delay equation

Mack [MA57a] has studied the problem of the efficiency of N machines unindirectionally patrolled by one operative when walking times and repair times are constant, and his model can be applied directly to the token system. Because the token and register insertion systems are identical from the point of view of delay, Macks result applies for both.

Let p_n be the probability that n stations transmit in a single revolution of the ring. Mack has shown that

$$p_n = p_o \binom{N}{n} \sum_{k=0}^{n-1} (a^k b - 1)$$

$$\text{where} \quad a = \exp\left(\frac{\lambda Ps}{Tr}\right)$$

$$b = \exp\left(\frac{\lambda NBe}{Tr}\right)$$

$$\text{and } p_o \text{ is obtained from } \sum_{n=0}^{N} p_n = 1$$

Hence, the mean number of messages transmitted in one revolution of the ring is

$$\bar{n} = \sum_{n=0}^{N} np_n \qquad 3.21$$

The service delay for the ring is given by

$$NBe + \bar{n}\, Ps$$

and the probability that at least one packet has arrived, and thus that the transmitting register is full is

$$p(z) = 1 - \exp\left(-\frac{\lambda}{Tr}(NBe + \bar{n}\, Ps + E)\right) \qquad 3.22$$

This value of p(z) is substituted in the appropriate basic performance function (3.4 or 3.5),and the delay is calculated by considering the M/G/1 queue as before.

3.4.2 The slot system delay equation

For the slot system p(z) can be derived if the line busy period is known, and this is calculated in a similar way to Hayes and Sherman's model [HAY71]. Line intensity is defined as the ratio of the mean of line-busy and line-idle periods. This implies that the active and idle periods are independent and stationary in the mean. The traffic being generated by a station is Tx, thus the total traffic passing through any node is

$$J = \sum_{i=1}^{N-1} Tx$$

The intensity at that node is given by

$$I = \frac{J}{Tr' - J} = \frac{Tx(N - 1)}{Tr' - Tx(N - 1)}$$

where the bandwidth available for data transmission is given by

$$Tr' = \frac{Tr(NBe - Ga)}{NBe}$$

The intensity at any station due to transmissions from a single other station is given by

$$I_i = \frac{Tx}{Tr' - Tx}$$

Thus the average duration of the idle period due to station i is

$$\frac{1}{\alpha_i} = \frac{Ps}{Tr\ I_i} = \frac{Ps(Tr' - Tx)}{Tr\ Tx}$$

In order to calculate the duration of the total idle period, it is assumed that the duration of the individual idle periods are exponentially distributed. It can be shown that the duration of the total idle period is then given by

$$\frac{1}{\alpha} = \frac{1}{\sum_{i=1}^{N-1} \alpha_i}$$

and as a station can only transmit one packet at a time, this assumption holds well. Thus, the output idle period is exponentially distributed with mean

$$\frac{1}{\alpha} = \frac{Ps(Tr' - Tx)}{(N-1)\ Tr\ Tx}$$

Finally, the busy period of the line is given by

$$\frac{1}{\alpha} = \frac{I}{\beta} = \frac{Ps(Tr' - Tx)}{Tr(Tr' - Tx(N-1))} \qquad 3.23$$

It is now possible to calculate p(z) by combining the fixed line transmission time, the busy period, the overhead due to gap digits, and the time to transmit the packet.

$$p(z) = 1 - \exp\left(\frac{-\lambda}{Tr}\left(NBe + \left(\frac{Ps(Tr' - Tx)}{Tr' - Tx(N-1)}\right)\left(1 + \frac{Ga}{QPs}\right) + Ps\right)\right) \qquad 3.24$$

The delay is derived as before.

3.4.3 The token system delay equation

As the token system is logically identical to the insertion system, the equation for p(z) is given by

$$p(z) = 1 - \exp\left(-\frac{\lambda}{Tr}(NBe + \bar{n}\,Ps)\right) \qquad 3.25$$

3.4.4 The pre-allocated bandwidth system delay equations

As the time between successive service quota is fixed, p(z) is given by

$$\begin{aligned} p(z) &= 1 - \exp\left(-\frac{\lambda N}{Tr}(Ps + Be)\right) \quad Ps \geq NBe \\ &= 1 - \exp\left(-\frac{NBe}{Tr}(N + 1)\right) \quad Ps \leq NBe \end{aligned} \qquad 3.26$$

for the single slot system and by

$$p(z) = 1 - \exp\left(-\frac{\lambda NBe}{Tr}\right)$$

for the slot per station system.

3.4.5 Line utilisation equations and packets turned away

For the single packet input buffer model the line utilisation equations are unchanged from the infinite size packet buffer model. As packets are turned away, $p(z) \to 1$ when $N \to \infty$ or $\lambda \to \infty$, and throughput when $Tx = Tr$ is less than for systems with an infinite input buffer.

The number of lost packets is given by

$$L_n = \lambda D$$

the proportion of lost packet is given by

$$P_l = \frac{\lambda D}{1 + \lambda D}$$

and thus the total traffic turned away by

$$T_l = \frac{\lambda D \, Tx}{1 + \lambda D} \qquad 2.27$$

3.5 Evaluation of Systems

The primary difference between the two buffer models is that whereas for the infinite size buffer, the probability of a station transmitting increases linearly with load, for the single buffer model, this probability increases exponentially. This in turn influences the length of the busy period of the line, and hence delay. By defining a performance envelope (coefficient of variation 1 and 0), the analysis concentrates on the effect of alignment delays. The width of this envelope increases with the number of nodes, and thus the models are not suitable for very large systems.

In this section traffic loads for the Cambridge ring are estimated, and the ring systems are compared.

3.5.1 Traffic estimates

Data for the traffic generated by the various devices which will be connected to the Cambridge ring has been collected, and an estimate

has been made for the additional traffic generated by the presence of the network itself. It was decided to emulate a large system as the demands on the network are thus maximised. The system chosen consisted of a large number of free-standing I/O devices including teletypes (150), VDU's (20), line printers (4), and graph plotters (2); and a number of computers including a 370, PDP11's (4), MOD1's (2), a Nova, and the CAP. The traffic estimates were made by combinatorially computing the probabilities of transmissions occurring at the same time, the program taking into account the bursty nature of data sources.

It was found that on average the ratio of the data entering the system to the data handling capability of the system (where the latter was calculated for the register insertion system) did not exceed 10% for a 16 bit data packet and 14% for an 8 bit data packet. In both cases the control section of the packet was 20 bits long. Although this does not indicate the performance of the system under peak load conditions, it does show that most of the time such a network will operate under low loads. This is further supported by the observation that in the Ethernet system the speed of transmission is not governed by the bandwidth of the network but rather by the efficiency of the software driving it.

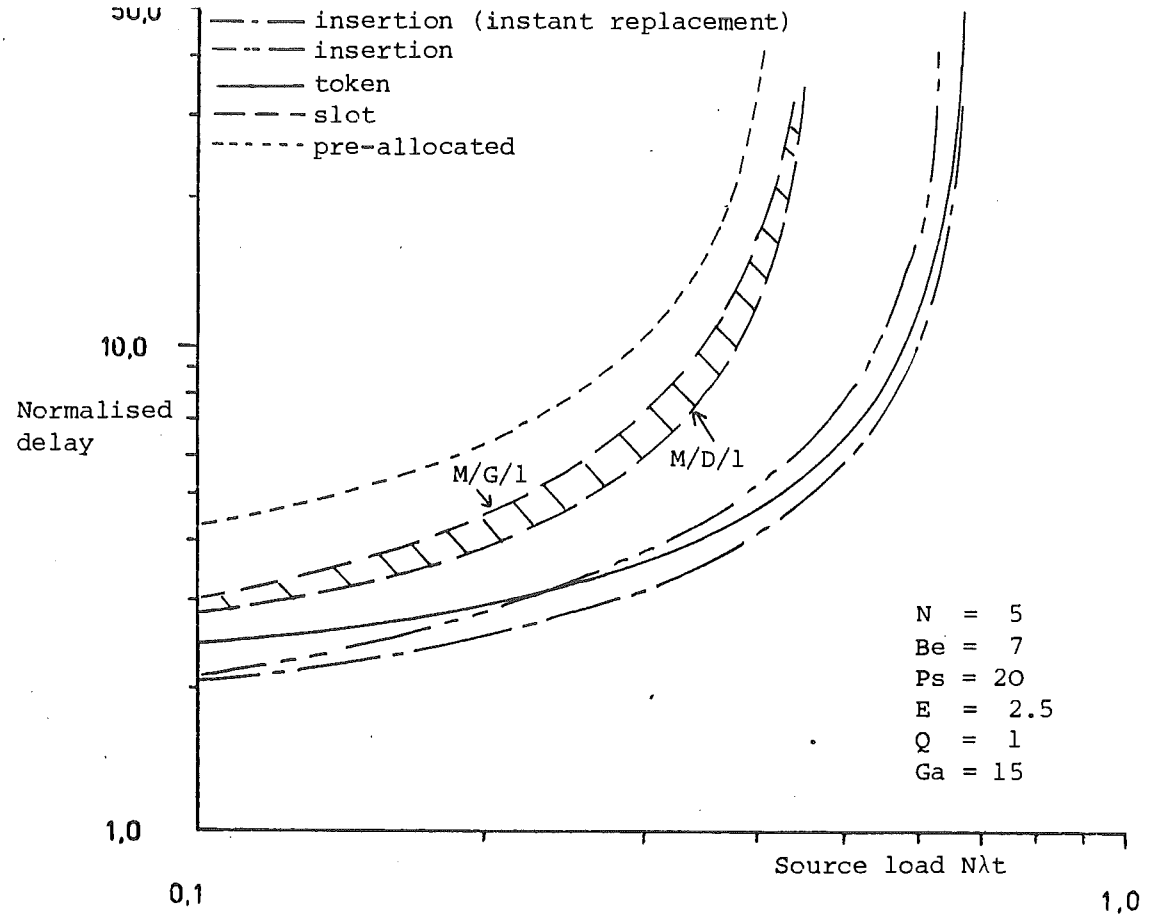
3.5.2 A comparison of delay characteristics

The delay is normalised and plotted in terms of ring delays per packet (that is in terms of the time for one bit to travel once round an empty ring) against the average load on the system $N\lambda t$, where t is the time to transmit a packet ($t = Ps/Tr$).

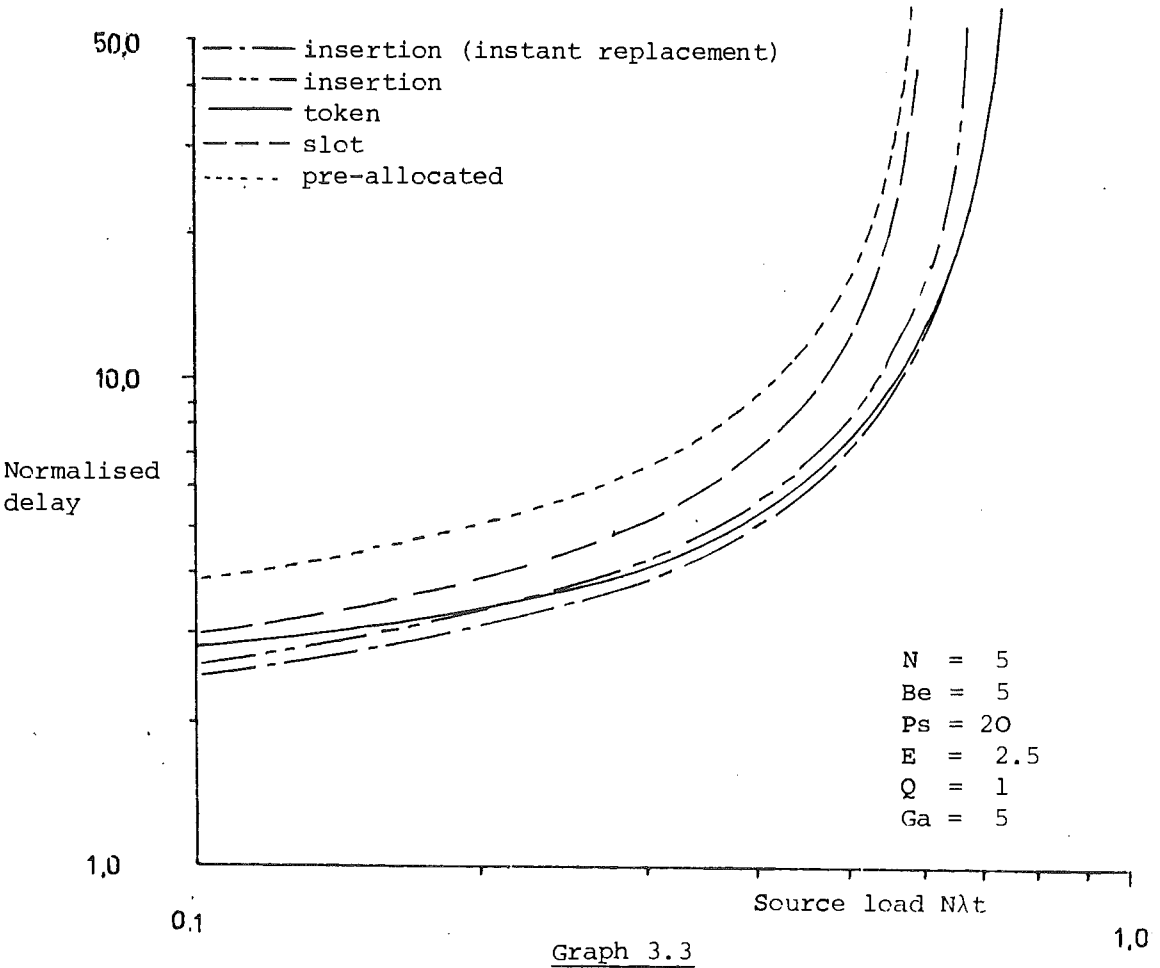
The envelope defined by the equations for the M/G/1 and M/D/1 queuing systems is considered below. The average delay for the M/D/1

system is always less than for the M/G/1 system, and this is shown in Graph 3.2 for the infinite size input buffer model. As N increases the envelope becomes wider due to the maximum and minimum values of the random components of delay moving further apart. It can be seen that the envelope is widest in the central region, as both queuing systems originate and saturate at the same points along the load axis. It is also at these extremes that the coefficient of variation of the service time tends to zero, whereas in the central region it increases from this value. All subsequent plots of delay are for the M/G/1 system ,and it should be borne in mind that they are thus optimistic in this central region. However, it is not expected that this will effect any of the systems to a greater extent than the others ,and thus relative comparisons are valid.

The infinite size input buffer model is presented first. Graphs 3.2 and 3.3. show the delays for a system with 5 stations, with an associated line delay of 7 and 5 bits respectively. It can be seen that as the system load increases ,there comes a point at which the rings saturate that is ,the delays tend to infinity. This defines the maximum data rate into the system. The highest delays are exhibited by the pre-allocated bandwidth system, but are not much different than for the other systems as there are only 5 nodes, and the fixed delay between service quanta is small. Next in terms of delay is the slot system with higher delays than either the insertion or the token systems due to the large proportion of gap digits (20%). It is also for this reason that it saturates at a lower value, equal in this case to the pre-allocated bandwidth system. Next are the insertion systems with and without instant replacement and the token system. The difference between the



Graph 3.2

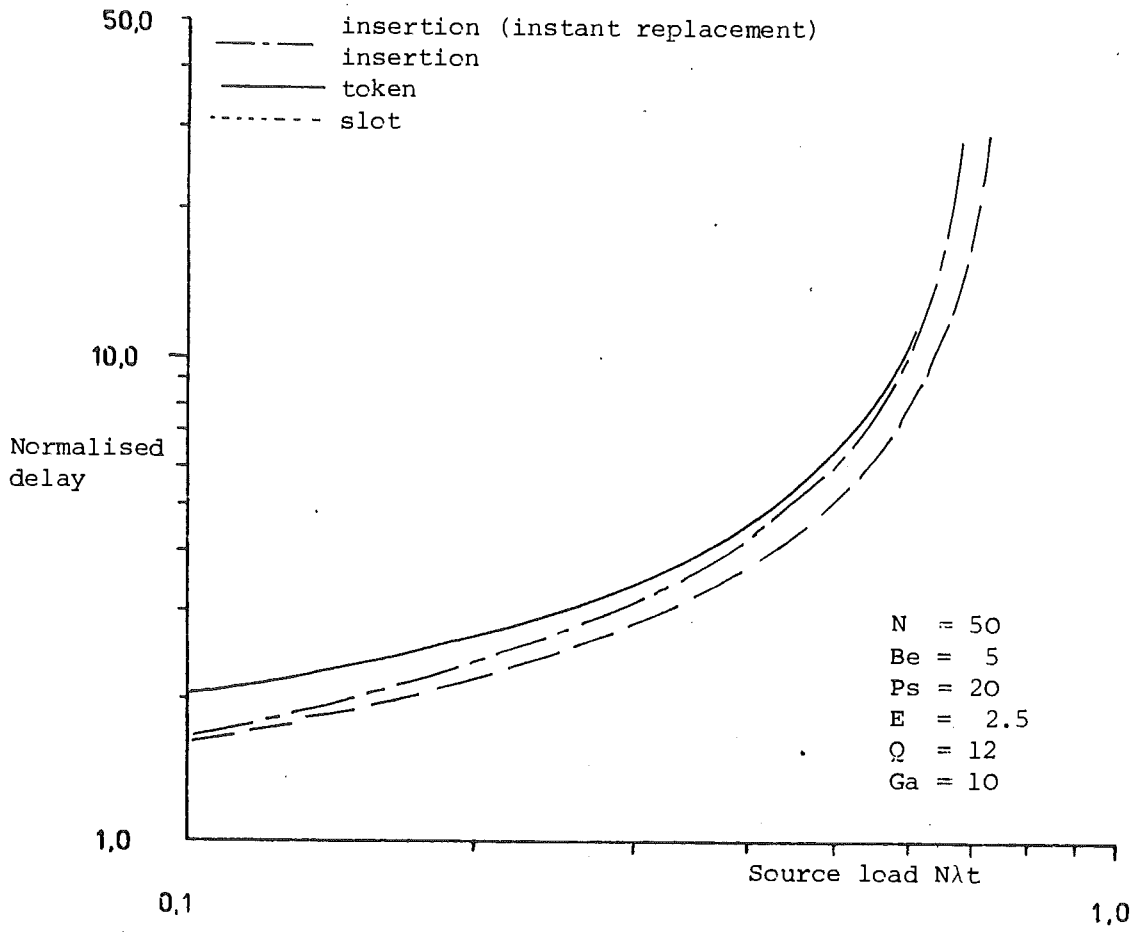


Graph 3.3

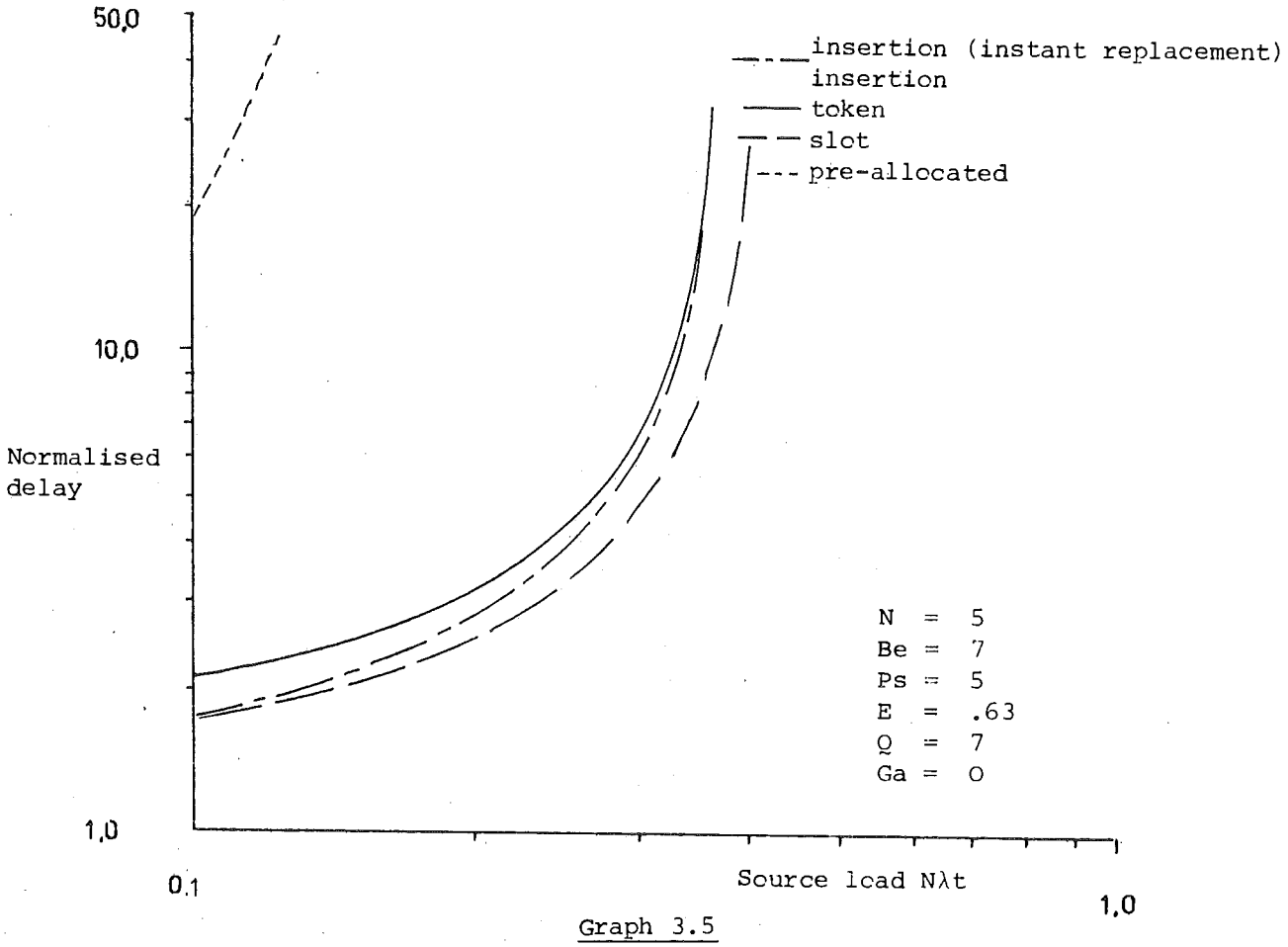
two insertion systems is small and increases with load. For low loads the token system behaves in a similar way to the slot system, and the alignment delay tends to half a ring delay. However, as load increases the token system behaves more like an insertion system (with instant replacement), and both saturate at the same point. Thus, there is a cross-over point at which the token system becomes superior to the insertion system with packet replacement time E.

Graph 3.4 shows the system of Graph 3.3 with 50 stations. Each station has an associated delay of 5 bits. Thus the minimum ring delay has increased. The pre-allocated bandwidth delays are high and do not appear on the graph. The insertion systems with and without instant replacement show little variation. The token system behaves in a similar way to insertion for high loads but shows higher delay for low loads due to larger non alignment delays. The slot system has lowest delays as for large N it is the most efficient, and gap digits form a smaller portion of the ring(4%). This is particularly brought out in the infinite size buffer model as p(z) is proportional to efficiency. At low loads the slot system is similar to insertion since the non alignment delay are governed by Q and are lower than for token.

As the relative values of Ps and Be change the saturation points move, and this is shown in Graph 3.5. Because Ps is small (relative to Be), the saturation point for the insertion and token systems has moved to the left compared to Graph 3.2, whereas for slot it remains approximately constant. Thus, there is a point at which the insertion and token systems become worse than the slot system even for small N and (not shown) some gap digits. For the pre-allocated system the delays are high as there is only one packet on the ring, and 85% of the bandwidth



Graph 3.4



Graph 3.5

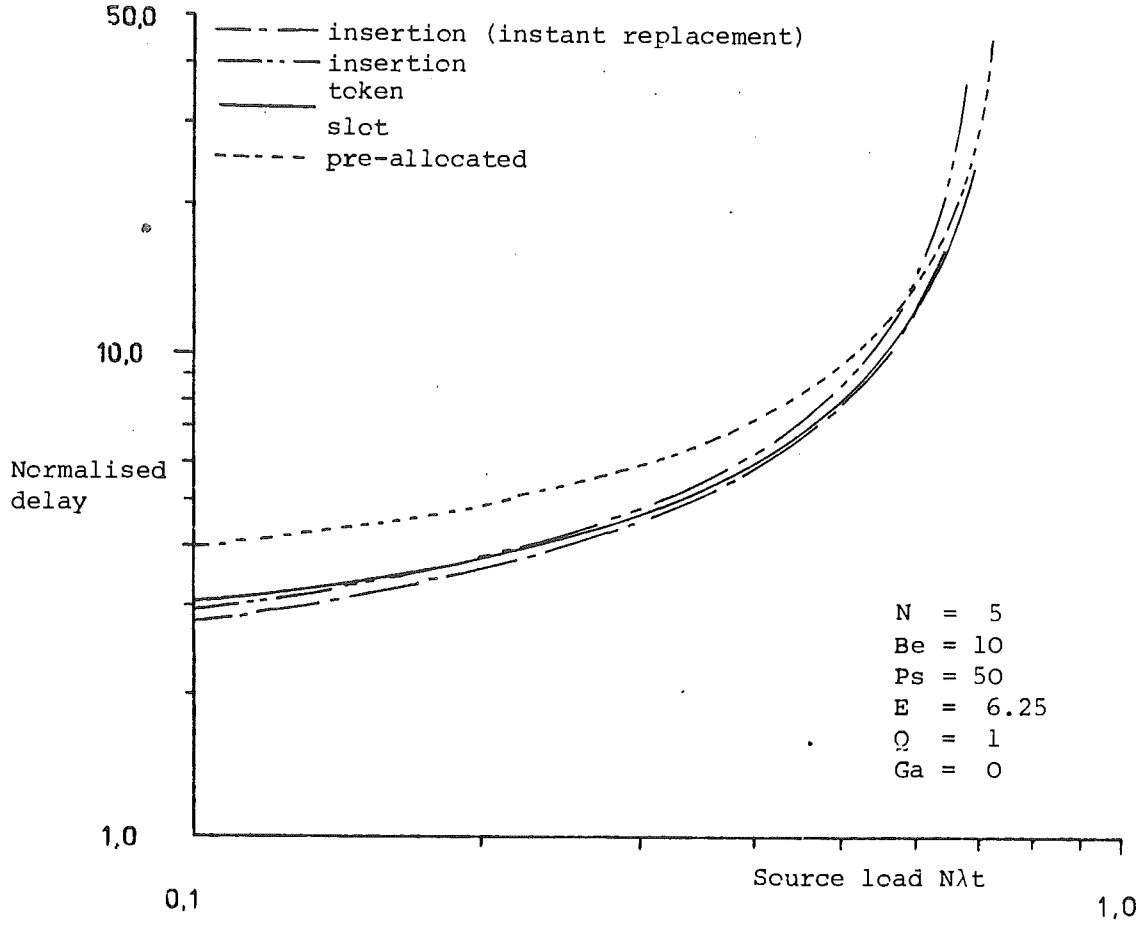
is wasted. In contrast as Ps increases this system performs better, and Graph 3.6 shows a system where all rings have similar delay characteristics.

Let us now consider the one packet input buffer model. Under some circumstances this model exhibits finite delay when $N\lambda t<1$, since the load consists of accepted packets and lost packets. The saturation point is higher, and corresponding values of p(z) and the average delay are lower compared to the previous model. Graph 3.7 shows the delays for a system identical to that of Graph 3.2. The most significant differences are the lower delays for the slot system. This is because for lower values of p(z) the overheads due to gap digits are smaller, and delay decreases. A further effect of p(z) taking lower values is that the effects of alignment delays are worse for the token than for the other systems. This is shown in Graph 3.8 for a 50 station ring, where the token system is always inferior to the others. For the single packet buffer model the slot system is never better than insertion, and for some configurations, all systems behave in a very similar way (high load region in Graph 3.8). In Graph 3.9 the cross over point between token and insertion is shown, and the slot system behaves like the token system as there is one slot on the ring and no gap digits.

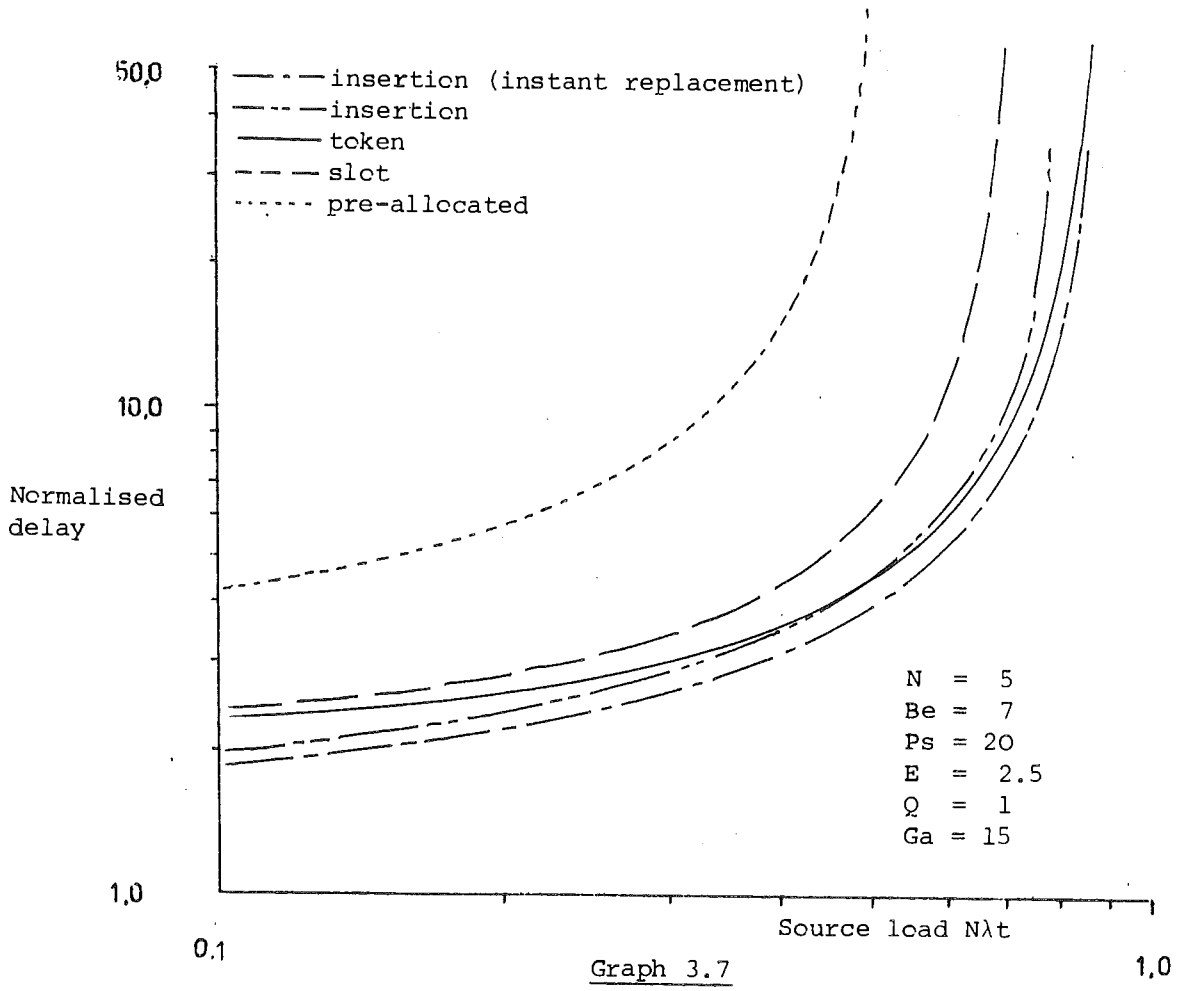
The delay models have shown that the ring systems behave in a similar way and that for any particular configuration the parameters govern which system exhibits lowest delays.

3.5.3 A comparison of line utilisation characteristics

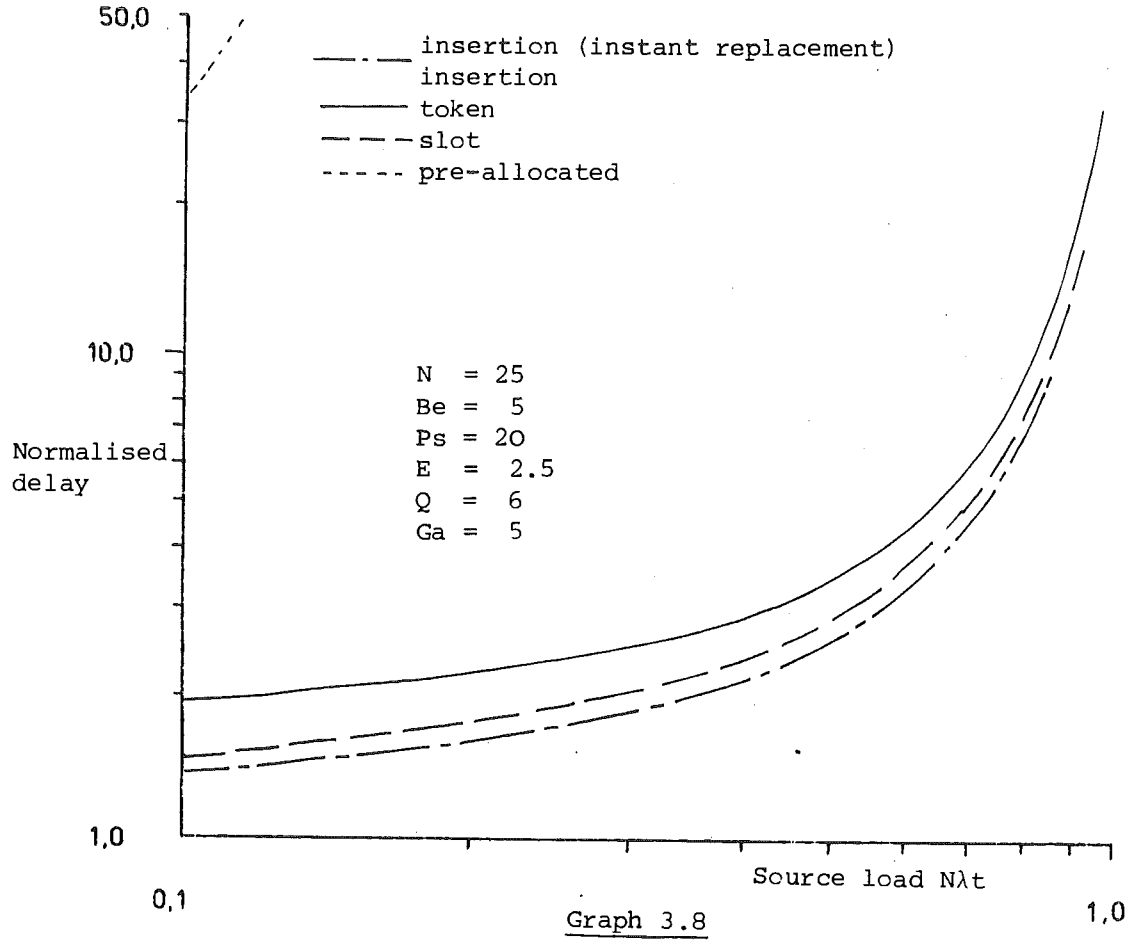
When comparing line utilisations traffic is generated by N homogeneous sources. Graph 3.10 shows line utilisation for the infinite size buffer model when packet size is large. The pre-allocated system



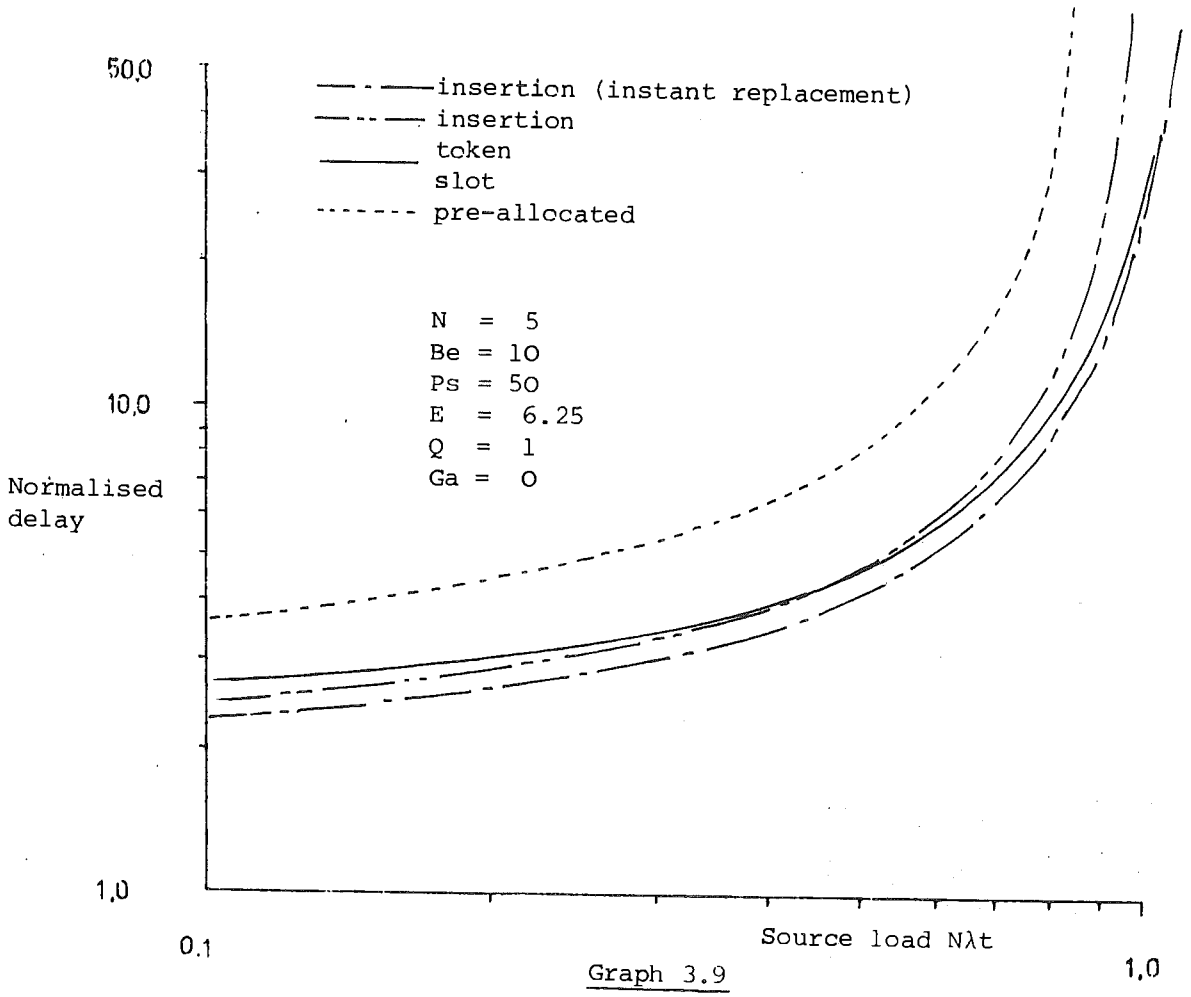
Graph 3.6



Graph 3.7



Graph 3.8



Graph 3.9

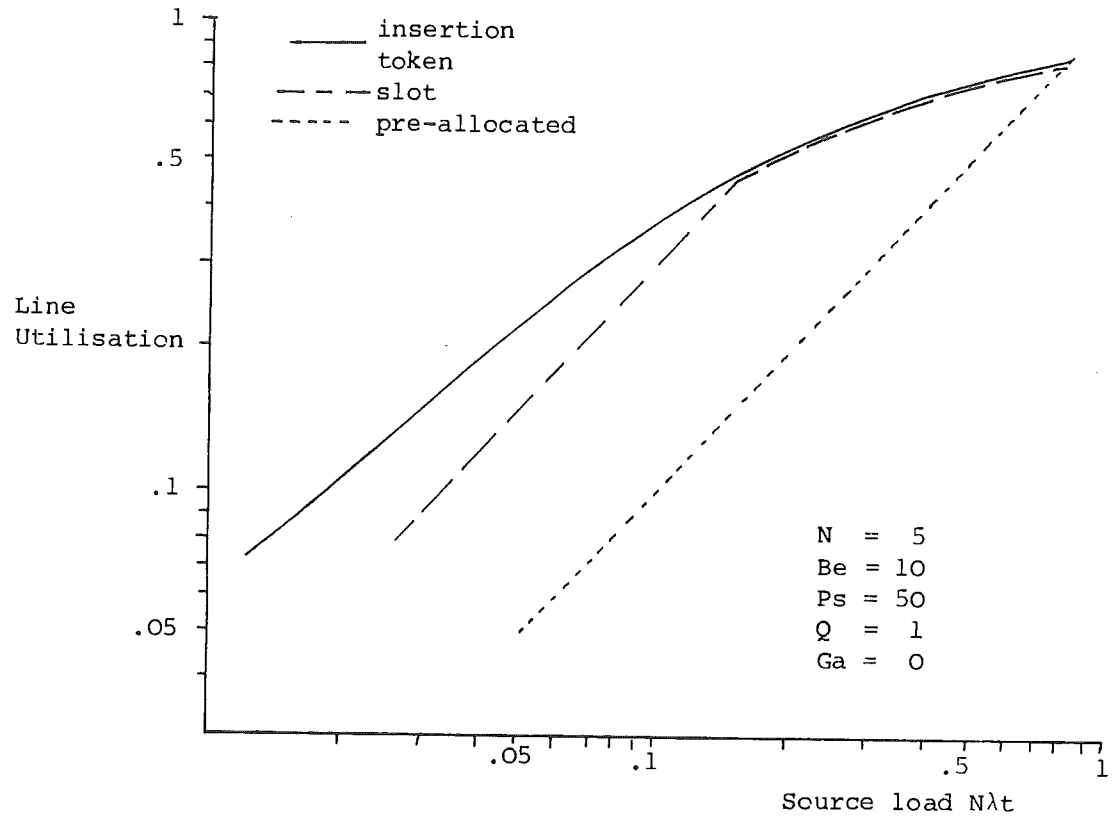
performs well under high load conditions, and since NBe = Ps all systems saturate at the same point. For low loads the slot throughput line has the same slope as the pre-allocated system (due to packets having to return to the source), whereas for p(z)N>Q the three rings are identical. The insertion replacement delay E has little effect on efficiency, and in all cases the token and insertion systems are equivalent.

Graph 3.11 shows a similar system to that of graph 3.10, but with the slot scheme carrying an overhead of a number of gap digits. It thus saturates at an earlier point, and the efficiency curve has a lesser slope at higher traffic levels than insertion.

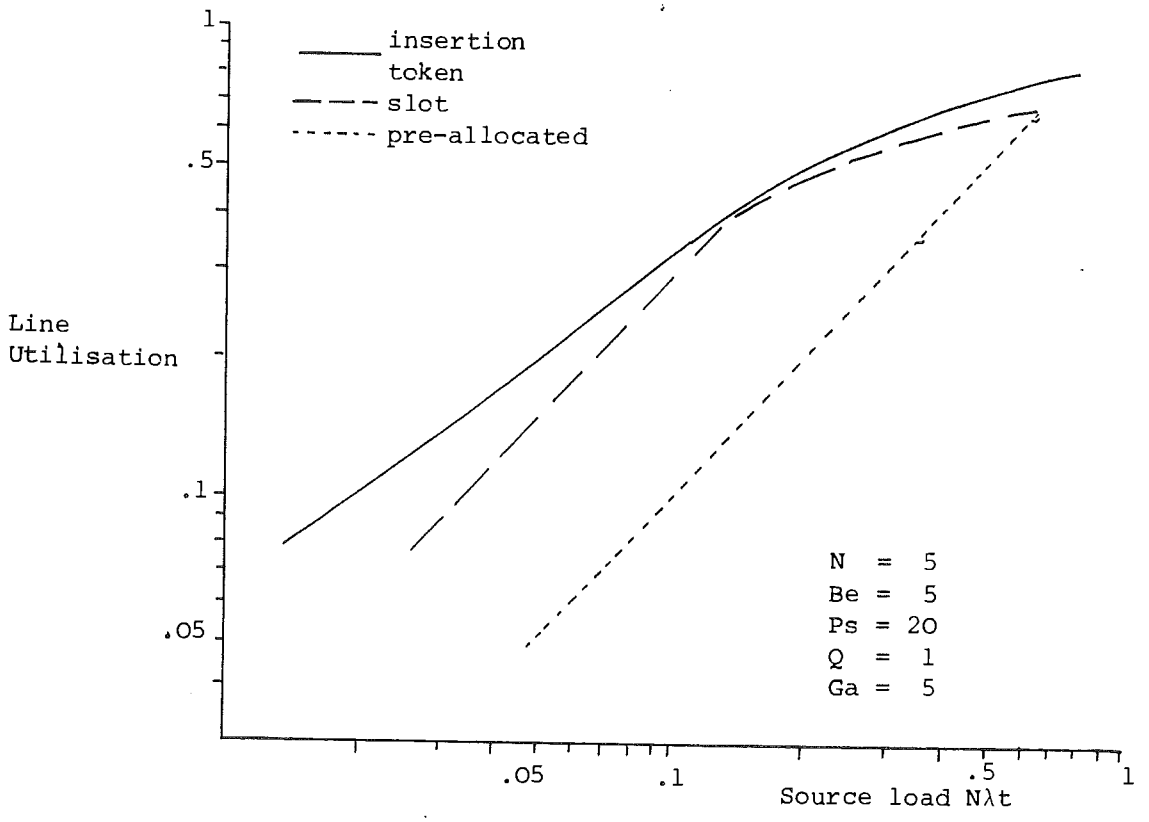
Graph 3.12 shows a system with 7 slots and 5 stations. For the slot system the sources cannot drive the ring beyond a certain point; however, due to the adverse ratio of Ps to Be, the insertion and token systems nevertheless saturate at a lower level.

Finally, Graph 3.13 shows a system with 25 nodes and very small electronic delay per node. This means that only a small number of slots can exist on the ring, and thus any gap digits have a noticeable effect. So although the slot system is more efficient than insertion at some traffic levels, there comes a point where it is overtaken by insertion. A similar graph for a larger number of stations would show slot to be superior even at maximum inputs.

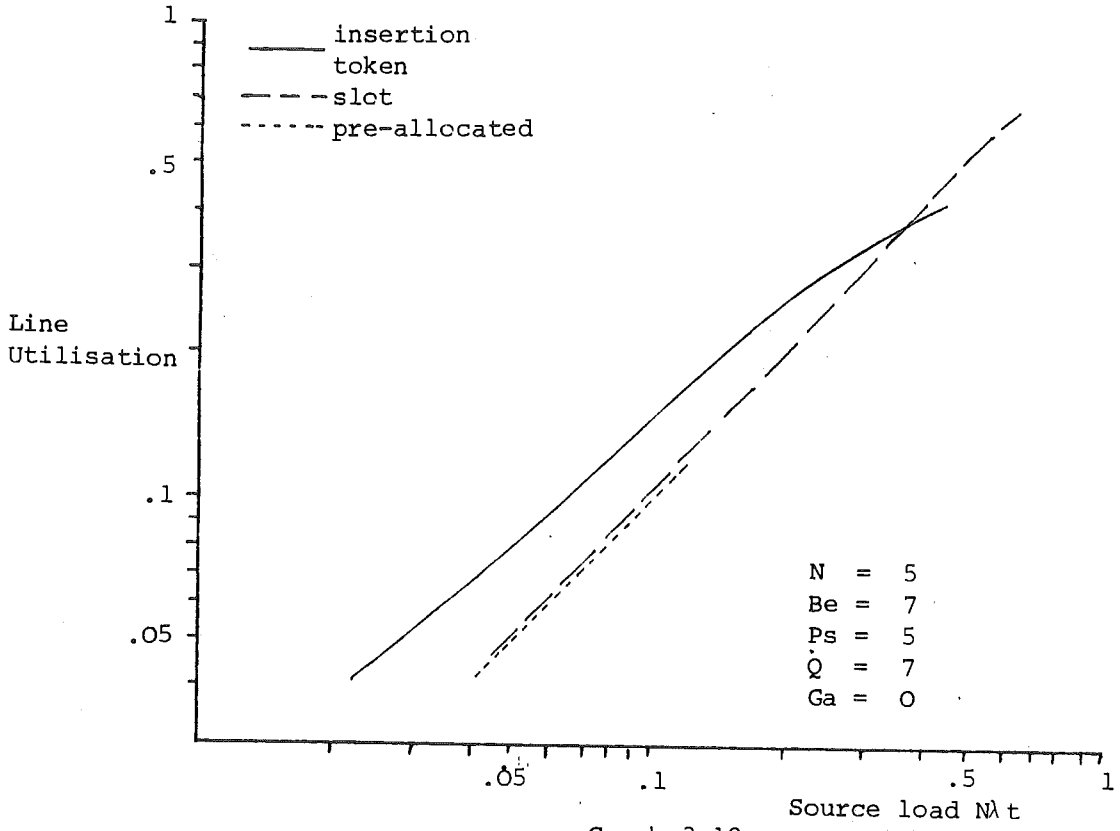
For the one packet buffer model the efficiency curves can be extended beyond the point $N\lambda t=1$, but the further they are drawn to the right, the more packets are lost, and thus the weaker the model. Graph 3.14 shows a system comparable with Graph 3.12. The maximum efficiencies are numerically close for both models, although the path



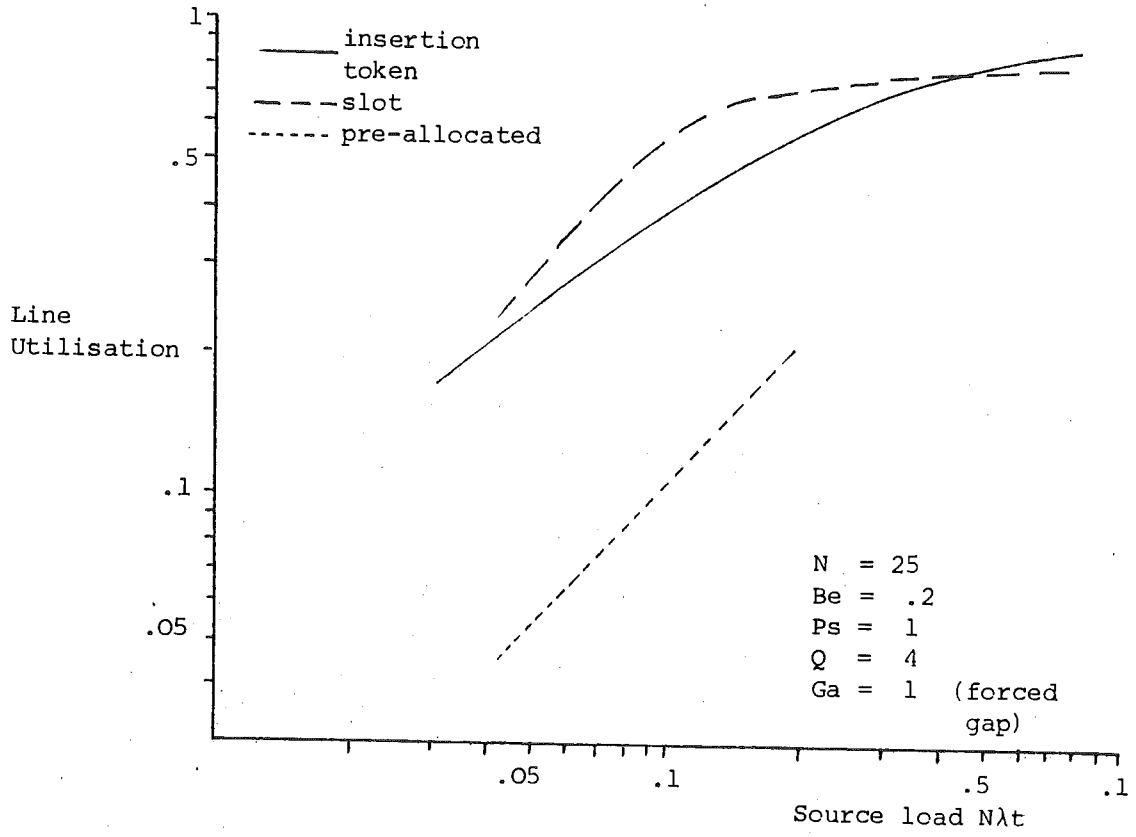
Graph 3.10



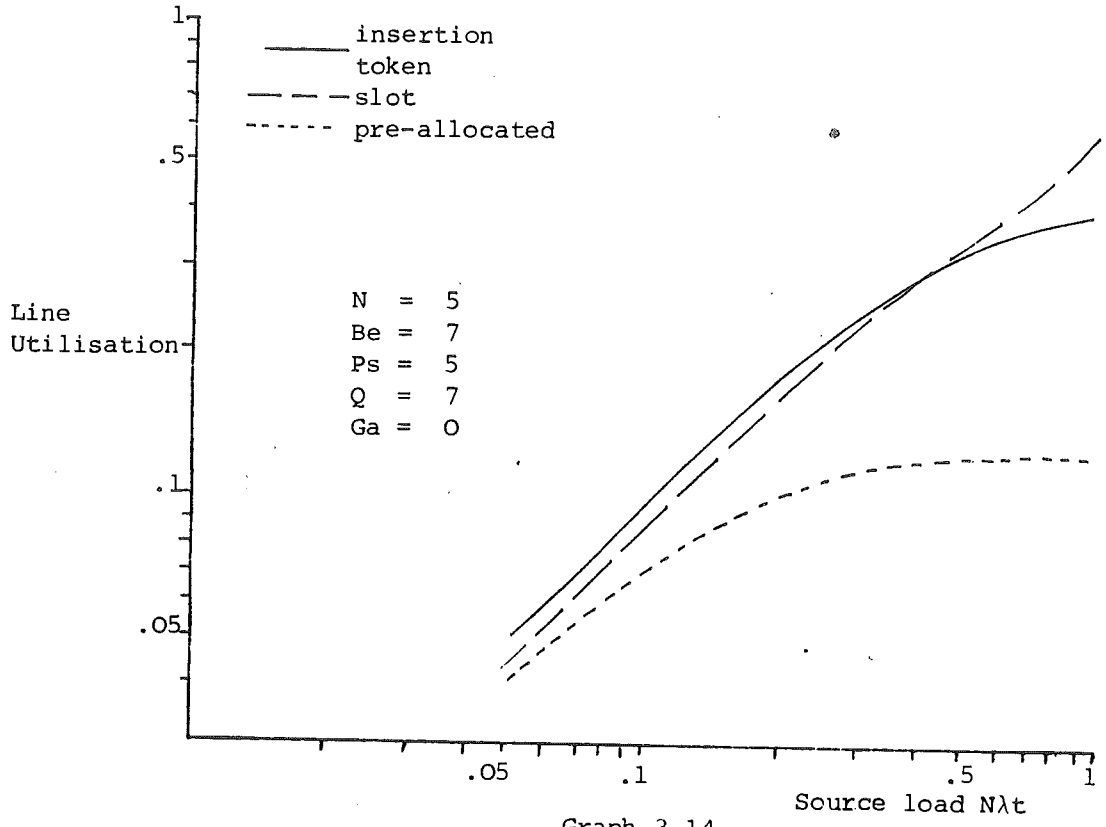
Graph 3.11



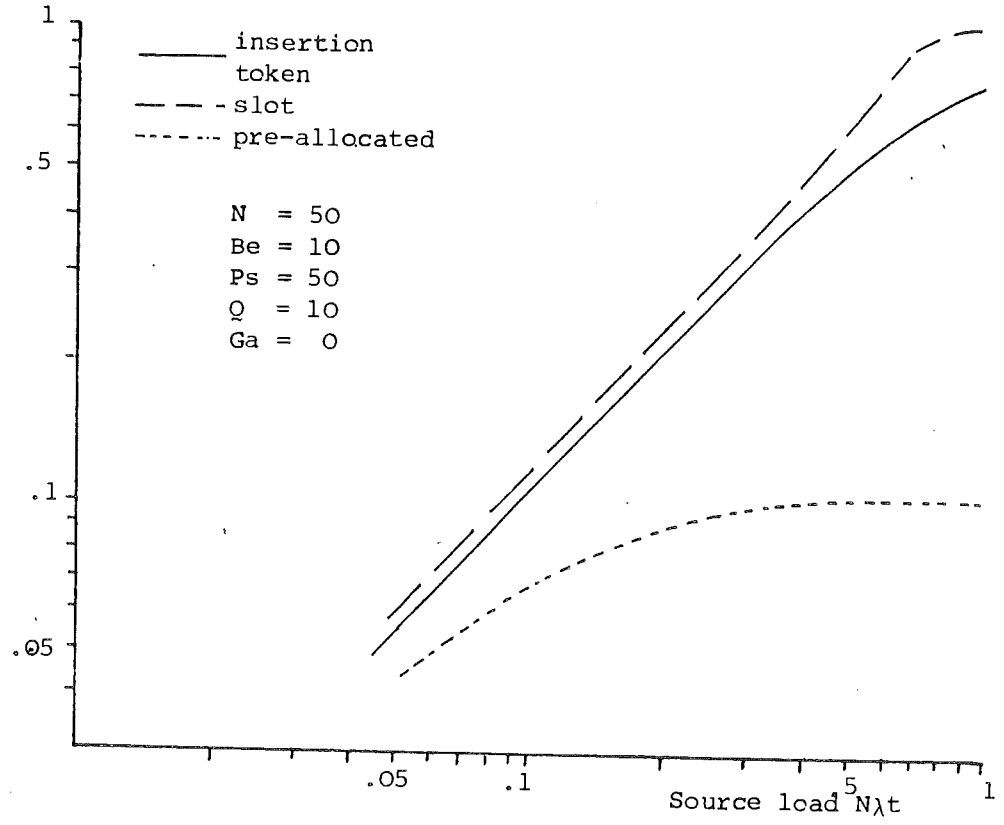
Graph 3.12



Graph 3.13



Graph 3.14



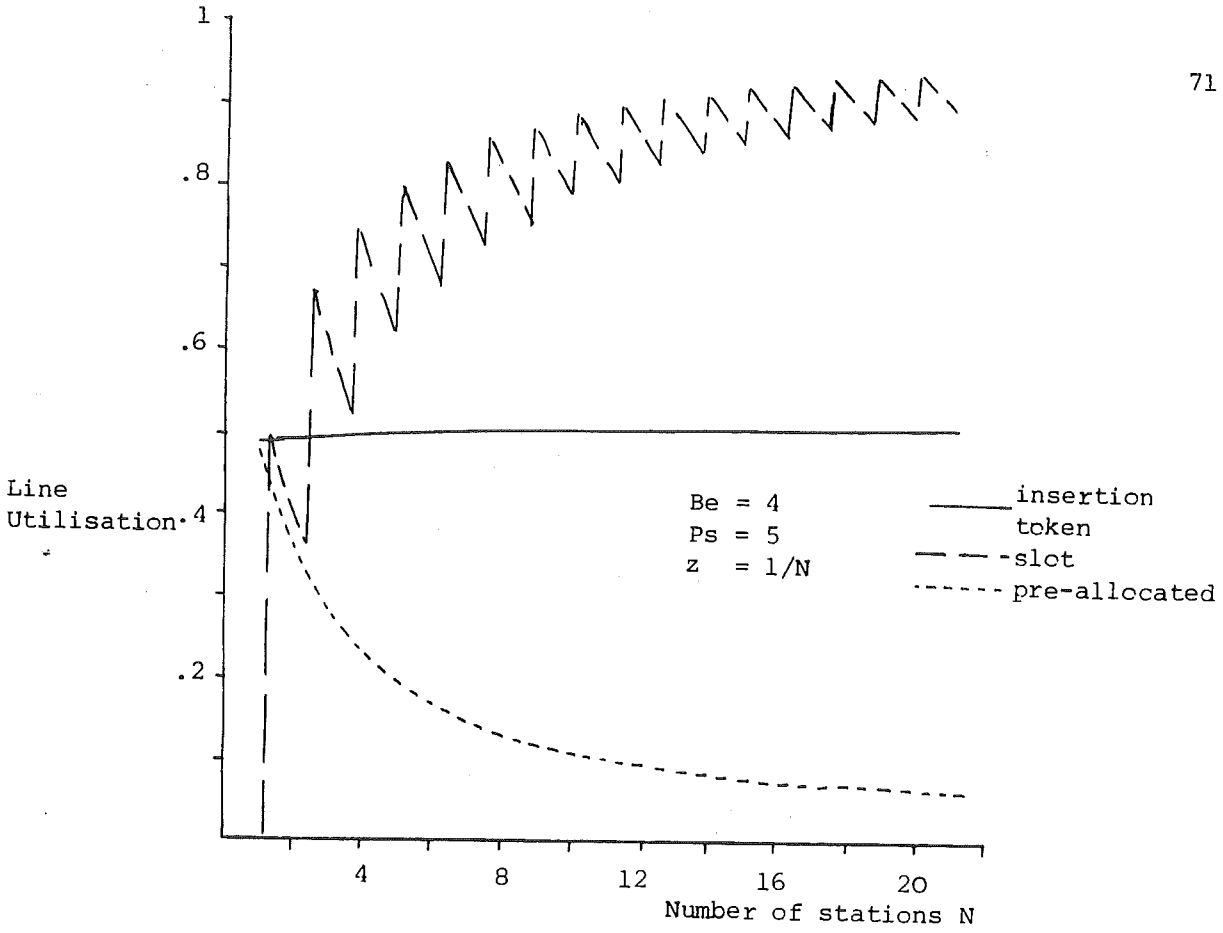
Graph 3.15

taken in reaching these points is lower for the one packet buffer model. Graph 3.15 shows a system with a large number of stations where the performance of the slot and token systems is initially similar, but at high traffic levels the slot system is better.

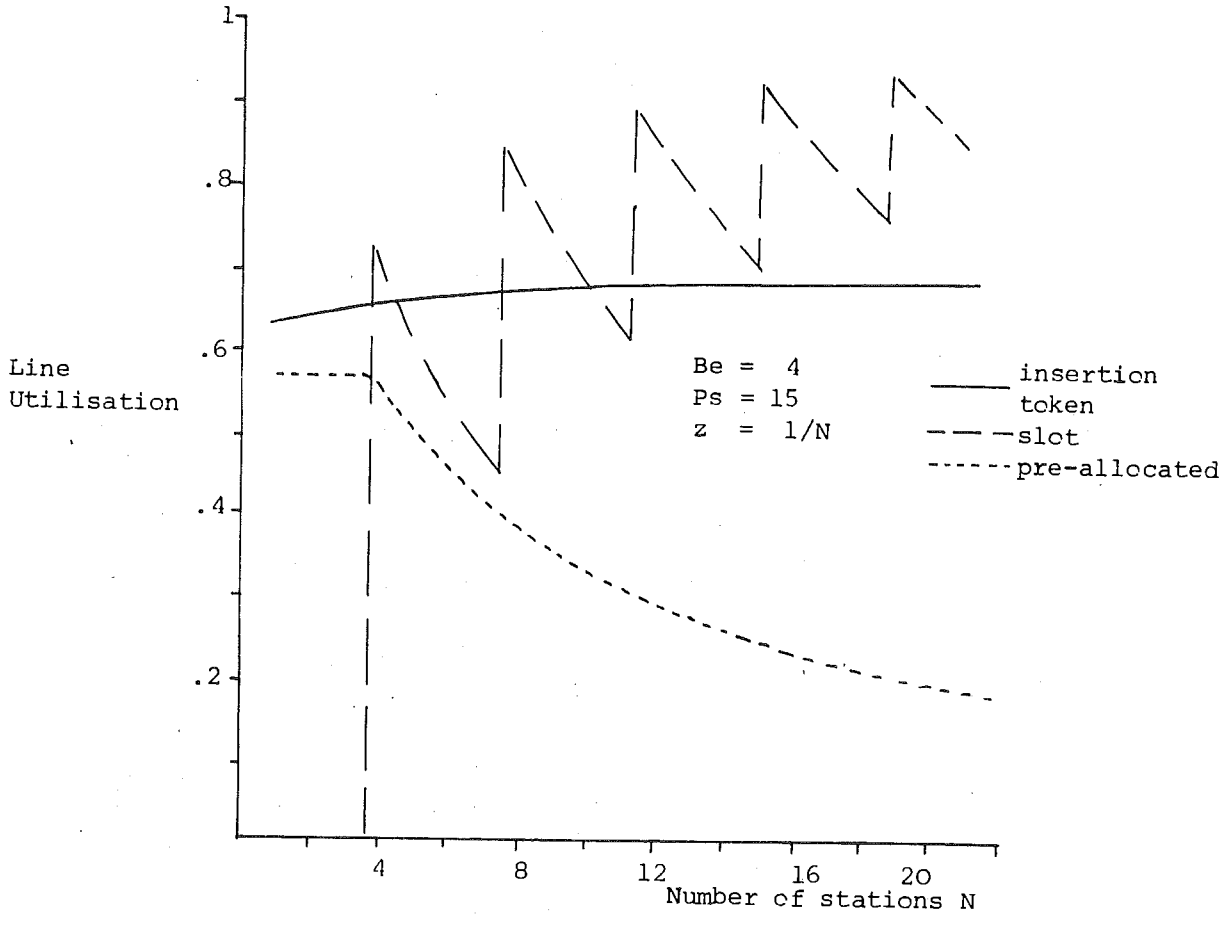
Taking an overall view of the efficiency models it can be seen that whereas for the insertion and token systems the efficiency is invariant with N, for the slot system (with Q<N) efficiency is a function of N. Thus as N increases, there comes a point (at a given load) where the slot system becomes better than the others. The pre-allocated bandwidth system is always poor, although it can perform well under high loads. In a real system this would be less true as the pre-allocated system is inflexible to non-homogeneous traffic sources.

3.5.4 The effect of the number of stations and packet size

The effect of the number of stations N and the packet size Ps on efficiency will now be considered. Because both buffer models behave in a similar way, the one packet buffer model with the input traffic constant at $z = \frac{1}{N}$ is used. Graph 3.16 shows the effect on efficiency of increasing N. As each N contributes a fixed amount of electronic delay the efficiency of the insertion and token systems is invariant with N. The pre-allocated system uses a single fixed size packet, and thus as NBe increases the available bandwidth decreases. The slot system cannot exist below some value of N since Q would then be less than 1. As N increases beyond this point, the efficiency of the slot system decreases until again QPs = NBe. At this point the number of gap digits is zero, and efficiency is maximised. As N increases further, the efficiency decreases until the next point at which QPs = NBe. Thus, the efficiency



Graph 3.16



Graph 3.17

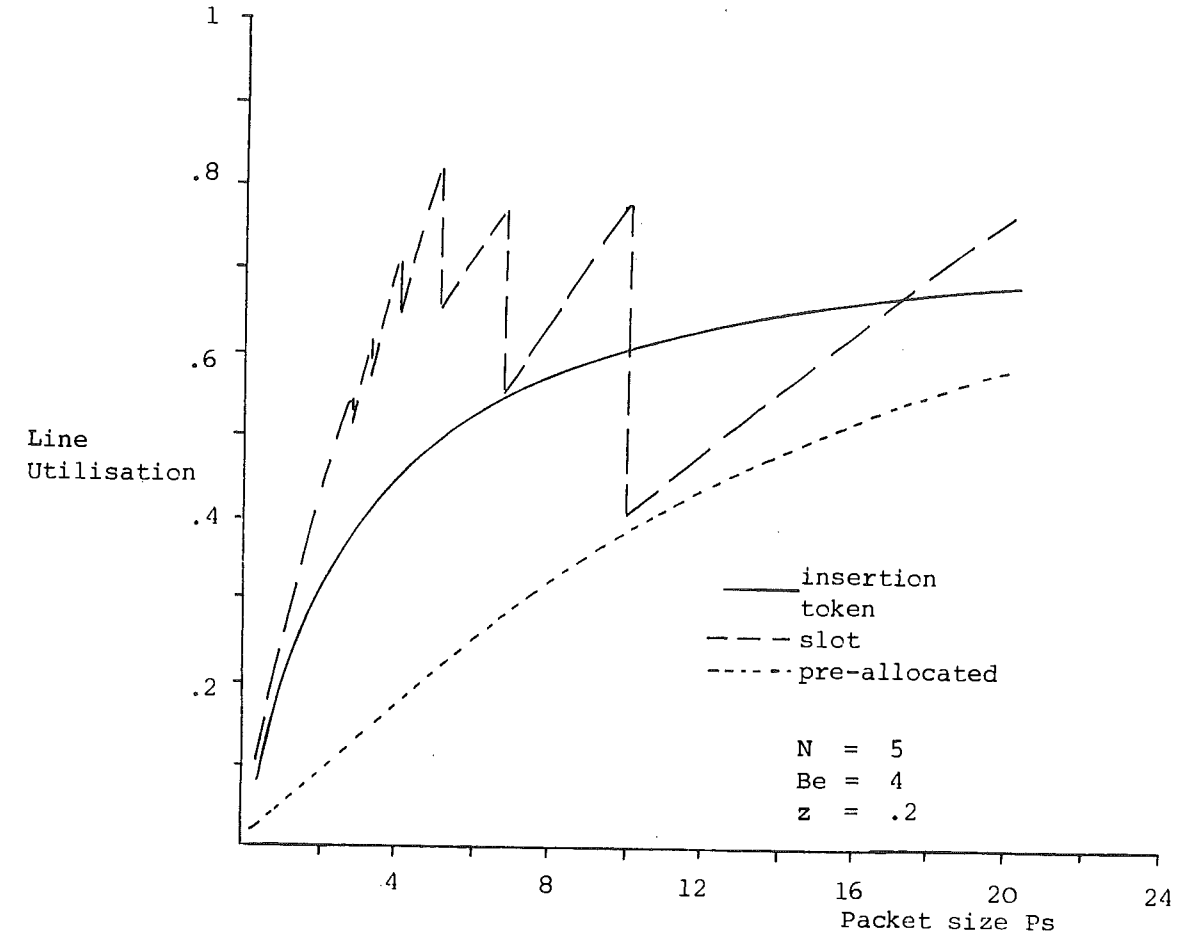
has a number of discontinuities, which have a smaller effect as N increases. It can also be seen that as N increases, the efficiency of the slot system tends to one. Graph 3.17 shows the same effect for larger packets: the efficiency of the token and insertion systems is higher, and the step function for the slot system has a greater effect.

The effect on efficiency of parameter Ps is shown in Graphs 3.18 and 3.19 for small and large number of stations respectively. As Ps increases, the efficiency of the insertion token, and pre-allocated systems improves. The efficiency for the slot system, on the other hand, undergoes greater and greater variations because the gap digits form a greater proportion of the ring. It should be noted that as Ps moves beyond some value, the slot system can no longer exist, and that for large N (Graph 3.19) the initial variation in efficiency is smaller.

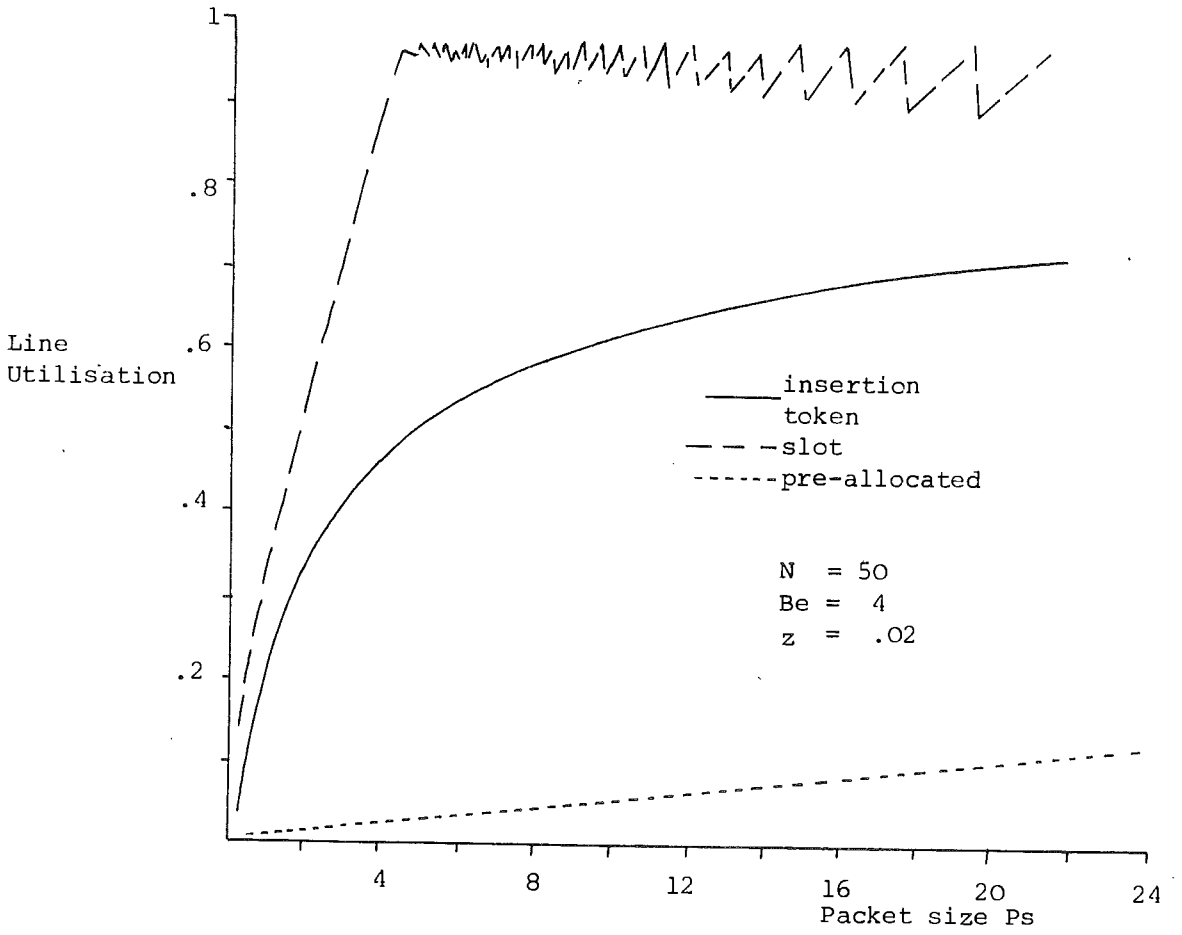
Finally, the effect of increasing N, where each additional station increases both the electronic delay and the system load, is examined. This is shown in Graph 3.20 where each station contributes 5% of the traffic. All systems show an improvement in efficiency with increasing N, the pre-allocated system falling off when the overhead due to wasted bandwidth becomes greater than the additonal traffic from new stations. The other systems tend to their expected efficiency values, the slot scheme exhibiting two types of discontinuities, due to the effect of gap digits and due to restrictions on transmission because of ring size.

3.6 The dominant user

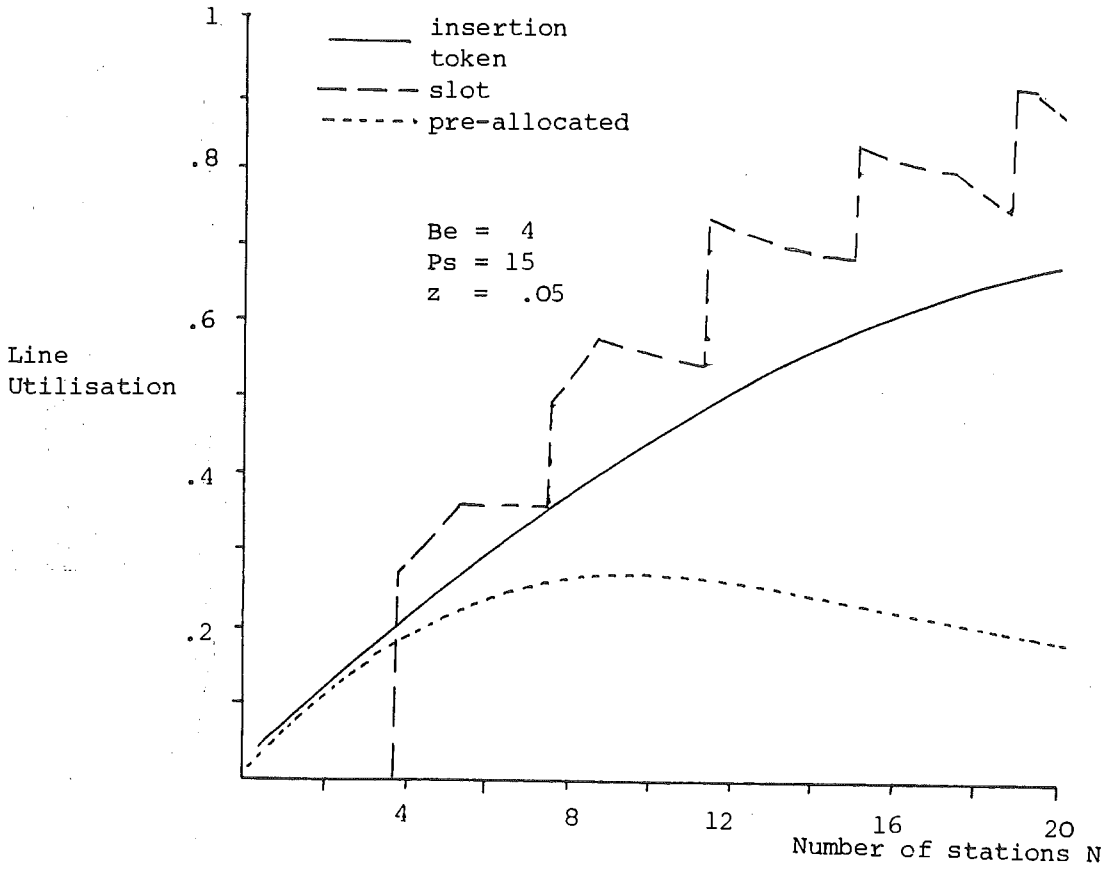
Non-homogeneous traffic is modelled by dividing users into two groups, with N_i and N_j users in the first and second groups, with mean



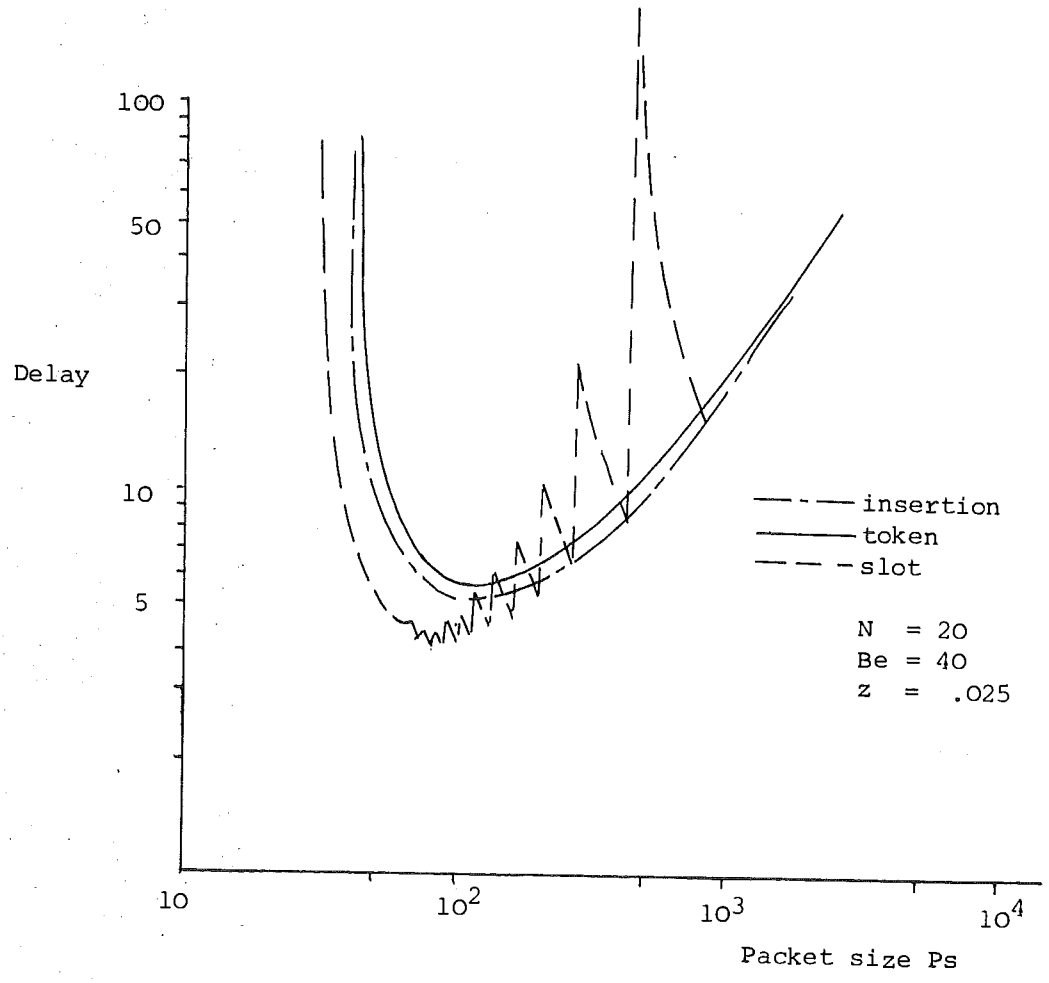
Graph 3.18



Graph 3.19



Graph 3.20



Graph 3.21

transmission rates of Tx_i and Tx_j bits/sec respectively. The proportion of traffic generated by group i is

$$\frac{Tx_i}{Tx_i + Tx_j}, \text{ and group j } \frac{Tx_j}{Tx_i + Tx_j}$$

As all stations have an equal opportunity to transmit, the bandwidth available to any user is proportional to the group transmission rate, up to a maximum determined by the maximum efficiency of the system. Thus

$$Tr_i = \frac{Tr\ N_i\ Tx_i}{N_i\ Tx_i + N_j\ Tx_j}\ S_{max}$$

and

$$Tr_j = \frac{Tr\ N_j\ Tx_j}{N_i\ Tx_i + N_j\ Tx_j}\ S_{max}$$

where

$$N_i\ Tx_i + N_j\ Tx_j \leq Tr\ S_{max}$$

Making the same assumptions as for the infinite buffer model, the probability that a station transmits in a cycle is given by

$$p(z)_i = \frac{N_i\ Tx_i}{Tr_i\ S_{max}}$$

$$p(z)_j = \frac{N_j\ Tx_j}{Tr_j\ S_{max}}$$

These equations are identical and can be represented by p(z)

$$p(z) = p(z)_i = p(z)_j = \frac{N_i\ Tx_i + N_j\ Tx_j}{Tr\ S_{max}} \qquad 3.28$$

The basic performance equations can now be written and are the same as before with all p(z)N terms replaced by

$$p(z)N \equiv p(z)\ (N_i + N_j) \qquad 3.29$$

and the new $p(z)_i$ and $p(z)_j$ replacing the single p(z) components.

For the pre-allocated bandwidth system the bandwidth available to a group is only proportional to the number of members in that group. Thus for these systems

$$Tr_i = \frac{Tr\ N_i\ S_{max}}{N_i + N_j}$$

$$Tr_j = \frac{Tr\ N_j\ S_{max}}{N_i + N_j}$$

and the probabilities of transmission are given by

$$p(z)_i = \frac{Tx_i(N_i + N_j)}{Tr\ S_{max}}\ ,\ p(z)_j = \frac{Tx_j(N_i + N_j)}{Tr\ S_{max}}$$

Having obtained the basic performance equations, the delays can be calculated as before, there being a separate equation for the i group and for the j group. This is because the non-alignment delays are different for the two groups. The line utilisations are obtained as before.

3.7 Fixed address field overhead and errors

In this section effects which increase the load on the ring are considered.

3.7.1 The effect of fixed size control fields

The analysis so far has not considered the fixed overhead in each packet due to control fields. As packet size varies, the length

of the control field changes, and small packets can carry a large overhead. Let the number of leader bits (including all control bits) be H, and let Ps' be the number of data bits in the packet. Thus, total packet size is given by

$$Ps = Ps' + H$$

Since the arrival rate of packets to the system is governed by the input data rate, the mean packet arrival rate is given by

$$\lambda = \frac{z\ Tr}{Ps'}$$

The paramter p(z) is obtained as before, and for the register insertion systems and the infinite buffer model is given by

$$\begin{aligned} p(z) &= \frac{zN(Ps + Be)}{Ps'} \\ p(z)^{E} &= \frac{zN(p(z)^{E}_{max}\ Ps + Be)}{Ps'} \end{aligned} \qquad 3.30$$

Similarly the efficiency is given by

$$S = \frac{p(z)Ps'}{p(z)Ps + Be}$$

It can be seen that as $\frac{H}{Ps} \rightarrow 0$ the effect of the control bits vanishes. The p(z) and efficiency equations for the other systems are derived in the same way, those for insertion being the same as for token.

3.7.2 The effect of packet size on performance

To measure the effect of packet size on performance, the input traffic is kept constant (in terms of bits per second), and the packet

size is varied. The infinite size input buffer is used so no packets are lost.

As the packet size is incremented, the arrival rate in terms of packets per second decreases, and the effect of non-aligment delays changes. These delays have a greater effect for large packets, as does the fixed delay component. However, the delay due to other traffic decreases with increasing packet size. For very small Ps the delay per packet tends to infinity. As packet size is increased there comes a point where the delay per packet is mainly governed by the fixed time to clear the packet (Ps/Tr), and thus with increasing Ps the delay per packet increases linearly. This indicates an optimum value for packet size,and this is shown in Graph 3.21 where the optimum packet size is approximately 100 bits.

3.7.3 Data stored in rings and errors

The error rate is governed by the type of transmission medium in use (wire or stages in a shift register). The register insertion system is implemented by combining the two types of transmission media. The token system requires at least one shift register stage per node, while the slot system can be implemented with only an OR gate in the ring path. It will be assumed that the error rate contribution of the OR gate is negligable, and that if an error occurs,the whole packet is retransmitted. The probability of two errors occurring in one packet is ignored. The error rate for wire is defined as e_w, and the error rate for shift registers as e_r. These error rates are measured in terms of data stored and thus incorporate the increase in error rate with increasing transmission speed. The system error rates are given

by

$$\text{for insertion} \qquad Er = N(Be - 1)\, e_w + e_r + p(z)\, Ps\, e_r$$

$$\text{for slot} \qquad Er = NBe\, e_w$$

$$\text{for token} \qquad Er = N((Be - 1)e_w + e_r)$$

It can be shown that the p(z) equations for the infinite size buffer model are given by

$$\text{for insertion} \qquad p(z) = N(Ps + Be)\,(Tx + NPs(e_w(Be - 1) + e_r))$$

$$\text{for slot} \qquad p(z) = \frac{NBe(NBe - Ga + Ps)(Tx + Ps\, NBe\, e_w)}{Tr\, Ps\, (NBe - Ga)} \qquad N \leq Q$$

$$= \frac{NBe\,(N + 1)(Tx + Ps\, NBe\, e_w)}{Tr(NBe - Ga)} \qquad N \geq Q$$

$$\text{and for token} \qquad p(z) = \frac{N(Ps + Be)(Tx + PsN(e_w(Be - 1) + e_r))}{Ps\, Tr}$$

Thus although the token and slot systems are affected by errors only linearly, the insertion system is succeptable to a further deterioration due to the data stored in shift registers.

3.8 Towards real systems

Invariably a model of a computer network has to make simplifying assumptions. In this chapter these have been made in such a way that the relative performance of a number of such networks can be analysed. However, in order to better evaluate a real system ,these assumptions can be changed. This does not alter the flavour of the results but makes them more representative of a particular network.

The chief assumption in comparing the networks has been that in all cases packets are of fixed size. This is true for the slot system but need not be for the others. The register insertion system can handle variable length packets but at a high cost in hardware and control algorithms. The token system, however, is well suited to transmit packets of any size, although if the maximum is very large, the effects of hogging reappear. Thus if messages are expected to be large and highly variable, the token system gives lowest delays and highest efficiency. Each message only carries one set of address and control bits, and there are no overheads in disassembling and reassembling the messages into packets.

Studies of computer-user statistics have shown that the input traffic to a network can be approximated by an exponential on-off model. Hayes and Sherman [HAY71] have developed an equation for the number of packets generated when a message approximated by an exponential on-off model arrives at a network. Such an arrival process poses a number of difficulties, as the number of packets in a message is a random variable, and a buffer large enough to hold this number has to be provided. Thus the point at which arriving packets are lost is shifted from the single buffer model. It can be assumed that the buffer is infinitely large and that all packets in a message arrive instantaneously. The system can then be treated as a bulk-arrival queue.

The models in this chapter treat the input to the system as a Poisson process with parameter λ. Thus,arrival and service processes are independent. This is not applicable to interactive systems where arrivals to a queue occur in batches, and no new arrival occurs until the previous entry has been completely serviced. An arrival

process which models interactive traffic is where the time interval from the completion of service of one message, to the arrival of the next is exponentially distributed with mean $1/\lambda$. This loads the system in a different way : that is ,the inputs are more bursty ,and the performance equations take a different form. Such input models place more emphasis on examining the stability and transient performance of the ring. As the networks studied in this chapter do not suffer from hogging, and a quantum of service is guaranteed after some maximum delay, their behaviour is stable. However, a transient at the input does degrade the performance to other nodes.

Finally, issues of setting up calls can be evaluated. It was stated above that if a variable length message is split into a number of fixed size packets, the overhead due to address bits increases. This need not be so if the call is set up. A model can be devised where the first packets of a transmission are treated as control, and the rest as data. There is a function which determines at which point setting up a call becomes feasible.

CHAPTER 4

STABILITY AND PERFORMANCE IN BROADCAST NETWORKS

4.1 Introduction

In this chapter some types of broadcast networks are explored. Initially the pure slotted Aloha system is studied, the primary consideration being to stabilise the network under all load conditions. This is achieved by introducing additional hardware which monitors the system state. Delay and throughput equations for such a system are derived and verified by simulation. Further algorithms for improving performance are studied, the analysis being carried out using Markov chain models and Monte Carlo simulations.

The Carrier Sense Multiple Access (CSMA) network is also considered, and a new method of implementing it akin to a ring system is presented. It is shown that the hardware problems associated with this implementation are simpler.

Finally, ring and slotted Aloha networks are compared, and it is shown that under some circumstances they have similar performance characteristics.

4.2 Previous work

The Aloha network can be modelled by assuming that the total traffic entering the channel G (consisting of new traffic and blocked traffic) is an independent process generated by an infinite population of users. If throughput is denoted by S and p_o is the probability that no additional packets are generated after transmission (for the

duration of a slot), then

$$S = G\,p_o, \quad p_o = e^{-G}$$

This has a maximum value of $S_{max} = .368$ when $G = 1$. In the unslotted Aloha system the vulnerable period is twice as long thus $S_{max} = .189$ when $G = 0.5$. If the system consists of N users each transmitting in a slot with probability x then

$$S = Nx(1 - x)^{N - 1}$$

and for $N = 2$, $S_{max} = 0.5$ when $x = 0.5$ [KL75b]. Thus, the channel has a finite capacity, and the rate of new packet generation must not exceed this value. If the population of users is not homogeneous but consists of a number of groups, then the maximum throughput may increase as a large user can utilise almost all the bandwidth, and cannot clash with himself.

The throughput analyses above disregard delay, and it can be shown that they are only valid when the average delay is infinite. The delay is a function of the retransmission parameter L and the ratio G/S (mean number of transmissions until success). These throughput models hold well when L is relatively small, and for L>20 the error tends to disappear. There is a tradeoff between delay and throughput: that is, throughput can be improved by increasing L (and therefore mean delay). However, beyond L = 20 the price for extra throughput is high. Minimum delays are experienced when there is only one user as the system behaves like a M/D/1 queue.

Metzner [METZ76] has shown that the maximum throughput of a slotted Aloha network can be improved from .36 to .53 by dividing the

users into two groups, one transmitting at high power and the other at low power. Whenever a high-power user clashes with a low power user, the weaker transmission does not cause interference, and the packet is transmitted successfully. The point at which throughput reaches .53 is when the higher-power users are transmitting more frequently than low-power ones. This can be extended to a number of power classes, convergence being rather slow so that 18 power classes are required to read a throughput of .9. It is interesting to consider that if no two transmitters are spaced equally apart and the receivers have perfect capture characteristics, then 100% utilisation is achieved due to the natural hierarchy (or even N/2 x 100%).

As well as showing poor channel utilisation, Aloha systems can become unstable under some conditions. Kleinrock and Lam [KL75b] have studied this behaviour; and by using simulations and fluid approximations they have shown that the Aloha channel becomes saturated if the set of transmitting stations is very large, independently of the arrival rate of packets to the channel. That is,the number of blocked terminals becomes arbitrarily large. This problem has been treated theoretically by Fayolle [FAY77], who shows that even for a finite number of users, the effective throughput of the system can tend to zero if the population of users is sufficiently large. This is done by showing (using probability balance equations) that the Markov chain representing the number of blocked terminals is not ergodic.

Fayolle also studied and gave stability conditions for two categories of channel control policies: those that restrict access to the channel from free users and those that control the retransmission rate of blocked ones. In the case of retransmission controls it is shown that only policies which restrict the rate of retransmission from blocked terminals in a slot to $H_1 = 1/n$, where n is the number of

blocked terminals and H_1 the probability of one of them transmitting in a slot, yield a stable channel. It is further shown that the optimum retransmission policy which maximises throughput is

$$H_1 = \frac{F_0 - F_1}{nF_0 - F_1} \qquad n \geq 1$$

where F_0 and F_1 are the probabilities of zero, and one packet arriving from the free terminals, for the infinite source model. Thus, if the source is Poisson, the optimum retransmission probability for blocked terminals is

$$H_1 = \frac{1 - \lambda}{n - \lambda} \qquad \lambda = \frac{F_1}{F_0} \qquad n \geq 1$$

which when substituted in the throughput equation reduces to 1/e as $n \to \infty$.

Carrier sense multiple access networks have been studied by Kleinrock [KL75a], who has derived throughput equations for a number of systems in terms of the applied channel traffic G and the normalised propagation delay A. These include the persistence system where on finding the channel free a terminal transmits with probability ε and delays by one propagation delay with probability $1 - \varepsilon$. It is shown that by using this scheme channel capacity can be improved for the unslotted case, although this is also a strong function of A (the ratio of packet length to maximum propagation delay). Under almost all circumstances the slotted non-persistent CSMA system has the best performance characteristics.

4.3 Algorithms for stabilising and improving the performance of the slotted Aloha channel

The approach adopted for stabilising the Aloha network is to find an optimum solution for a system with two nodes, and by blocking some transmissions iteratively reduce large systems to a system with two nodes as rapidly as possible.

Slots in an Aloha type network can be divided into three categories: those which contain data, those that are empty (gaps), and those in which a clash has occurred. A normal Aloha network is constructed from stations which can distinguish between data slots and clash slots. It is a relatively simple task to arrange that a station can also recognise gap slots. It is also arranged that there are two counters in each station which are used for keeping the system state and which are initially set to zero. One is incremented every time the station detects a clash on the broadcast medium. The other counter increments only at the clashes that have occurred due to the station transmitting. That is,it is incremented every time the station transmits and clashes. Once the station has transmitted successfully,this counter is reset to zero.

The most important rule in stabilising the channel is that a station is only allowed to transmit if its station clash count is greater or equal to the system clash count. Consider a system with only two stations. When these two clash,it is arranged that each chooses to retransmit in one of the following two slots (L = 2) with probability 0.5. Thus,with probability 0.5 the retransmissions split, and with probability 0.25 each they clash again at the first and second slots,and the algorithm is repeated. If the stations split, then the one that transmitted in the first slot is free to transmit a new packet in

the second slot. However, because the station clash count has been reset to zero, while the system clash count is still at one, any such transmission is blocked. An increment to the system clash count indicates that at least two stations have clashed. Thus, this count should be decremented after the second data transmission. However, as the second station might become inactive (e.g. a fault) without transmitting, this counter is decremented after the first two gap or data slots. It is possible to arrange that the choice for retransmission is made between more than two slots or that the exponential distribution is used, and both these possibilities are explored later.

When the number of nodes in the system is greater than two, the algorithm behaves in the same way, except that the system clash count is decremented after the second data or gap slot following a clash and at every data or gap thereafter until the next clash. As an example, it is assumed a clash of 128 stations has taken place. The system clash count and all station clash counts are incremented by one, and on average 64 stations will choose to transmit in the next slot. When they do, the appropriate counts are incremented, the other 64 stations are now blocked from transmitting, and the algorithm is repeated. As the system clash count increases, more and more of the stations are blocked from transmitting, until in due course only two remain, and the system behaves as described above. Once these two stations have transmitted successfully the other two of the group of four transmit, and so on up and down the blocked stations, until the system clash count reduces to zero again. This description has ignored the possibility of the same group of stations clashing repeatedly. However, as a choice of retransmission slots is made after each clash, in due course more and

more stations become blocked and the clashes are resolved.

It can be seen that the algorithm operates by forming arriving traffic into groups, each group being dealt with before the next is allowed to transmit. In the normal (lightly loaded) state packets are transmitted with little or no delay.

The stability criteria derived by Fayolle states that the blocked terminals must transmit at most with probability $\frac{1}{n}$ in each slot. This condition is satisfied for the two node system. As the number of nodes increases, the overall transmission rate from the blocked nodes tends to decrease (due to compulsory gap digits when changing groups); thus, the system is stable. Fayolle also showed that the optimum retransmission rate is less than $\frac{1}{n}$; thus, the algorithm operates in a favourable way, and an improvement might be obtained by changing the probability distribution for choice of retransmission slots.

The above scheme will be called algorithm A, and a number of improvements called algorithms B, C, and D are now discussed. They all restrict access from the blocked terminals in the same way as the basic algorithm, and thus stabilise the channel. To describe algorithm B a system with two nodes and L = 2 is again considered. When a clash is followed by two data slots, the delay to the transmissions is at a minimum. If a clash is immediately followed by another clash and only then by the two data transmissions, there will be a gap following these data transmissions for the system clash count to return to zero. If the original clash is immediately followed by a gap, then there will be a clash in the next slot with probability 1, and again there will be a gap slot to reduce the system clash count. As the probability of a clash after the sequence clash-gap is 100%, there is no reason why this gap cannot be treated in the same way as a clash,

with the stations repeating the algorithm and choosing new slots. However, as in this case by making the choice again the system is not being divided into groups, there is no need to increment the system clash count as would normally be the case when choosing retransmission slots. If the system consists of more than two nodes, these arguments still apply since the pattern clash-gap can only result in a certain clash As N increases the probability of this pattern tends to decrease for a given traffic level and system clash count.

So far, the hardware used for implementing the stabilising algorithms consisted of two counters per station, plus some simple logic. For algorithm C some additional units are incorporated which store the system state for some period of time in the past. It is thus possible to adjust the system to operate optimally when certain traffic patterns occur. For example, with two stations the longest delays in algorithm B are experienced when the pattern clash-clash occurs. If it were arranged that the system count is not incremented after this pattern, the delay would be reduced. However, in the presence of a larger number of nodes the second data transmission would probably be interrupted due to stations from the lower level transmitting and interfering. If it is arranged that the system clash count is not incremented following the pattern gap-clash-clash, the delay when arbitrating between two stations is reduced. However additional clashes can occur when changing groups as the system clash count is kept artificially low. Algorithm C has been simulated, and it was found that performance gains are only marginal.

A more promising approach is to restrict the transmission rate of stations which have been blocked and which are freed by a descending system clash count. Algorithm D introduces a parameter Φ which represents the probability of a station transmitting when the system clash count

has been reduced to its level. When the station clash count is zero, all stations may transmit, and some will be blocked in the normal way. Two successful transmissions (or gaps) will eventually take place, and the system clash count will be reduced. The freed stations transmit with probability Φ, and if a clash occurs, a new group is formed which is smaller than an equivalent group in algorithm A and whose size is a function of Φ (and the traffic level). Thus, in order to create a new subgroup, fewer clashes have taken place. On the other hand, there is a bulge in the number of stations blocked at the lower levels; however the algorithm repeats and the system will at worst operate as with algorithm A. If when the system clash count drops a group of two stations is freed, then for $\Phi = 0.5$ the system operates exactly as after a clash (without the clash taking place), and for larger Φ it tends to the straight system (algorithm A). It can be seen that some stations now experience a larger delay, but since throughput has been improved, the average delay has probably not changed significantly.

The algorithms described in this section all have the same property that they stabilise the Aloha channel and improve its performance. However, for heavy loads the system can nevertheless on average be in a state where the system clash count is high, and where each transmission experiences a high (but finite) delay. This is because while transmissions are being ordered, new packets are likely to be arriving ready for transmission in each station, and when the system clash count reduces to zero, it immediately climbs up again. This can be remedied by restricting the access of new packets to the system under heavy load conditions. Thus, new packets are transmitted with probability ε. This probability is close to 1 for low loads and smaller otherwise. With this scheme the average value of the system clash count is reduced, and the system is not continually in a blocked state at high loads. A system combining

the use of probabilities Φ and ε gives a good solution but is also complex.

A simpler scheme for stabilising the Aloha channel would be to only use the system clash count (SCC) register. Both blocked and free stations ($\Phi = \varepsilon$) would transmit with probability 2^{-SCC}, which would stabilise the channel but not form the blocked stations into groups.

In the following sections analytic models for the Aloha system are derived and simulations are used to show the correct functioning of the above algorithms.

4.4 Modelling the stabilised Aloha channel

In this section an analytical model for the stable Aloha channel is presented for a system with two nodes. The delay and throughput are calculated, as is the mean time to resolve a clash. Results of simulations for a system with a larger number of nodes are then given, and it is shown that they possess similar characteristics to the two node model.

4.4.1 The analytical model

The mean time to resolve a clash is considered first: that is, the time from a clash until both nodes have transmitted successfully. This is initially calculated for the geometric approximation to the exponential distribution with parameter y. It is then derived for a general distribution, the uniform distribution being treated as a special case. Let W_1 and W_2 be random variables representing the retransmission slot chosen by stations 1 and 2 respectively. Let $\bar{y} = 1 - y$, then

$$P(W_1 = W_2) = \sum_{n=1}^{\infty} P(W_1 = W_2 = n)^2$$

$$= \frac{y^2}{1 - \bar{y}^2}$$

As

$$P(W_1 = W_2) + P(W_1 \neq W_2) = 1$$

$$P(W_1 \neq W_2) = \frac{2y\ (1 - y)}{1 - \bar{y}^2} \qquad 4.1$$

Let x be the probability that a free station transmits in a slot ($\bar{x} = 1 - x$). Both stations transmit successfully if the free station does not interfere with the one which chose the higher W.

$$P_{suc} = \frac{\bar{x}\ 2y(1 - y)}{1 - \bar{y}^2}$$

When a clash occurs the algorithm for retransmission is repeated. Thus the process is memoryless, and the mean number of trials before success is $1/P_{suc}$. The mean value of the exponential distribution is $\frac{1}{y}$; and because of the memoryless property of the exponential distribution, this is the delay component both for clash and for split. Thus, delay until success is

$$D_{suc} = \frac{1}{y\ P_{suc}} = \frac{2 - y}{\bar{x}\ 2y(1 - y)} \qquad 4.2$$

As the probability of the free station interfering increases ($x \to 1$), the delay tends to infinity.

Let us now consider a general system where the probability of each station choosing slots 1, 2, 3 .... is f_1, f_2, f_3 ....

Now

$$P(W_1 = W_2) = \sum_{n=1}^{\infty} f_n^2$$

$\therefore$ as before

$$P_{suc} = \bar{x}\left(1 - \sum_{n=1}^{\infty} f_n^2\right)$$

Thus for the uniform distribution where $f_n = 1/L$

$$P_{suc} = \bar{x}\left(\frac{L - 1}{L}\right) \qquad 4.3$$

The delay experienced by the stations consists of two components: that due to clashes, and that due to the successful transmission when $W_1 \neq W_2$. Let $E(Max(W_1, W_2))$ represent the mean value of the chain of higher W's. When the two stations clash, $W_1 = W_2$, and the delay component due to clashes is given by

$$E(Max(W_1, W_2) \mid W_1 = W_2) = \sum_{n=1}^{\infty} nf_n^2 \qquad 4.4$$

$\therefore$ for the uniform distribution

$$= \sum_{n=1}^{L} \frac{n}{L^2} = \frac{L + 1}{2L} \qquad 4.5$$

The delay component due to successful transmissions is given by

$$E(\text{Max}(W_1, W_2) \mid W_1 \neq W_2) = 2 \sum_{n=2}^{\infty} n f_n (f_1 + f_2 + \ldots + f_{n-1}) \quad 4.6$$

$\therefore$ for uniform distribution

$$= 2 \sum_{n=2}^{L} \frac{n(n-1)}{L^2} = \frac{2(L+1)(L-1)}{3L} \quad 4.7$$

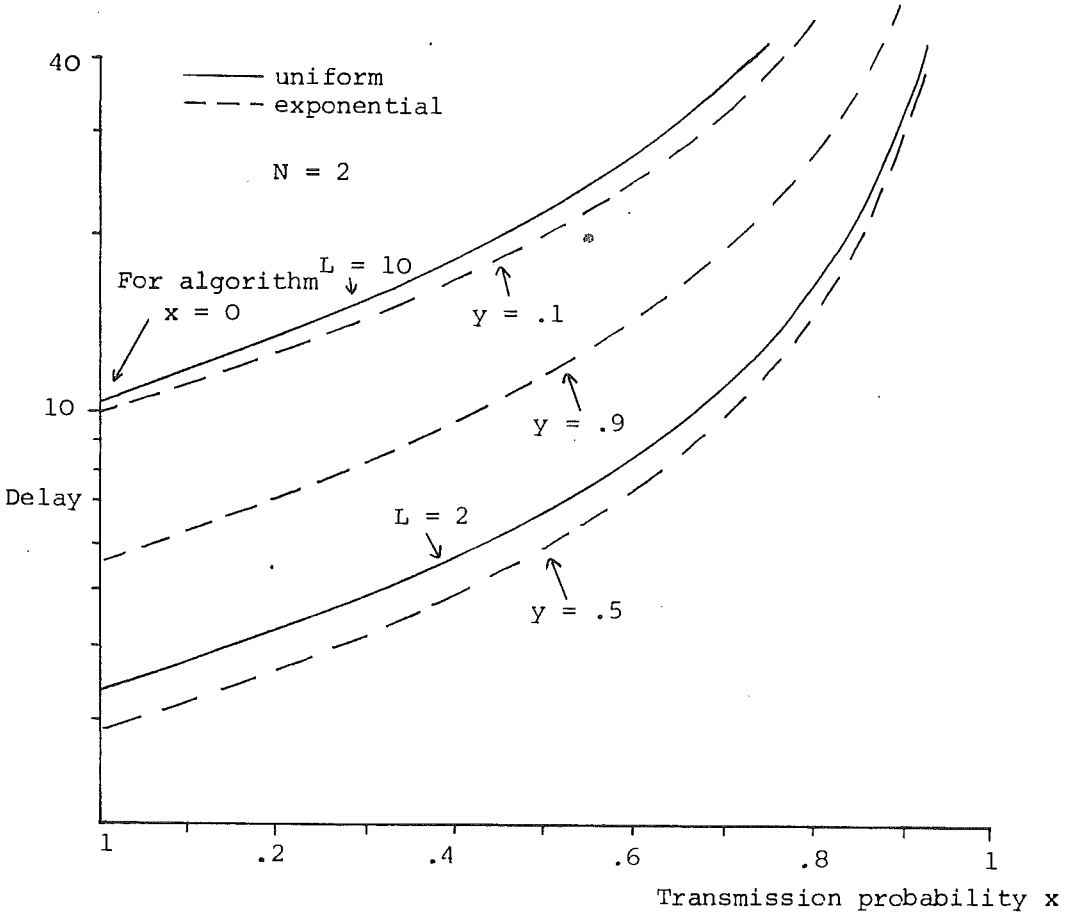
Thus, combining equations 4.3, 4.5 and 4.7, the mean delay until success for the uniform distribution with parameter L is given by

$$D_{suc} = \frac{L}{\bar{x}(L-1)} \left(\frac{L+1}{2L} + \frac{2(L+1)(L-1)}{3L} \right) \quad 4.8$$

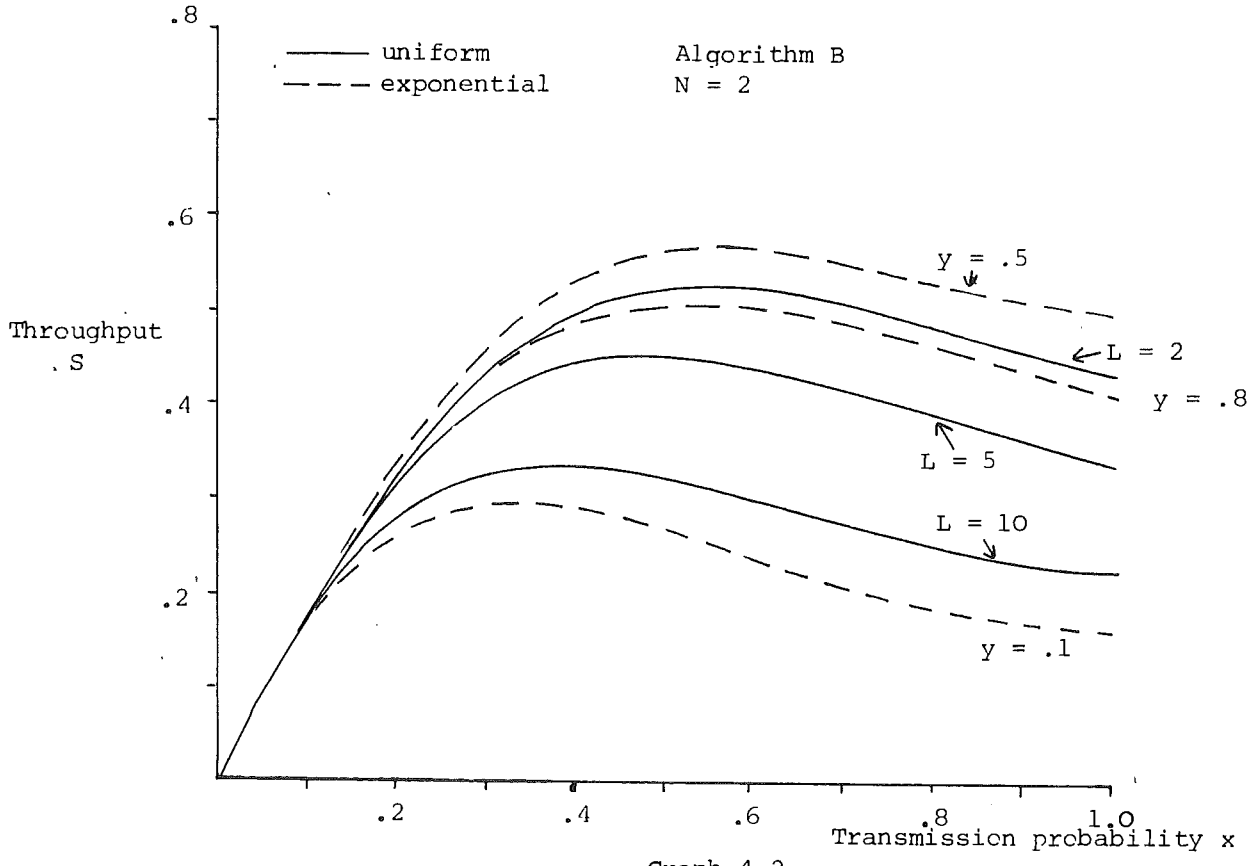
As expected for large L, the delay component of D_{suc} tends to $\frac{2L}{3}$.

Equations 4.2 and 4.8 are plotted in Graph 4.1. It can be seen that as x increases, the mean time to resolve a clash also increases, and at $x = 1$ is infinity. For the algorithms considered in the previous section the free station cannot interfere with the blocked one which corresponds to $x = 0$. It can also be seen that an optimum value for y (and L) exists because as y increases in value, D_{suc} reaches a minimum and then begins to increase again. However, as the system clash count is decremented for both gaps and data, there is a severe penalty incurred if this count is reduced before the two stations have transmitted successfully. Thus, in practice the retransmission slot would be chosen from the two succeeding slots ($L = 2$).

The throughput and delay of the slotted Aloha system with stabilising algorithm B are now derived. This is calculated by considering the mean time between clashes (Mt) and the number of successful transmissions in that period. Only the number of transmissions in one direction (Ex)



Graph 4.1



Graph 4.2

is considered, since by symmetry this is half the total number of successful transmissions. As the system is ergodic, the throughput is given by

$$S = \frac{2Ex}{Mt} \qquad 4.9$$

Let us first consider the exponential distribution where a blocked station retransmits in the next slot with probability y, and where W_1 and W_2 are random variables defined as before. The delay between clashes is given by

$$\begin{aligned} &E(\text{Max}(W_1, W_2) \mid W_1 = W_2) && \text{for } W_1 = W_2 \\ &E(\text{Max}(W_1, W_2) \mid W_1 \neq W_2) + x^{-2} && \text{for } W_1 \neq W_2 \end{aligned}$$

where x^{-2} is the mean time until a clash in the straight system (no blocked stations). Now $P(W_1 \neq W_2)$ is given by equation 4.1, and for the exponential distribution

$$E(\text{Max}(W_1, W_2) \mid W_1 = W_2) = E(\text{Max}(W_1, W_2) \mid W_1 \neq W_2) = \frac{1}{y}$$

Therefore, the mean time between clashes is given by

$$Mt = \frac{1}{y} + \frac{2y(1 - y)}{x^2(1 - y^{-2})} \qquad 4.10$$

The number of successful transmissions in this period is obtained by conditioning on W_1 and calculating the number of successes for all possible values of W_2.

Let Θ represent the probability that W_1 and W_2 have split in a particular way

$$\Theta = P(W_2 < W_1) = P(W_1 > W_2) = \frac{1}{2}\left(1 - \frac{y^2}{1 - \bar{y}^2}\right) \qquad 4.11$$

The number of $2 \rightarrow 1$ transmissions before W_1 (with probability Θ) is

$$\begin{array}{ll} 1 & \text{if } W_2 < W_1 \\ 0 & \text{if } W_2 > W_1 \end{array}$$

Therefore, the total number of transmissions before W_1 is Θ. The number of $2 \rightarrow 1$ transmissions after W_1 (with probability Θ) is

$$\begin{array}{ll} 1 + R & \text{if } W_2 > W_1 \\ R & \text{if } W_2 < W_1 \end{array}$$

where R is the number of successful transmissions in an unblocked system before the first clash.

$\therefore$ total number of transmissions after W_1 is

$$\Theta R + \Theta(1 + R)$$

Now R is given by

$$R = \sum_{n=1}^{\infty} (\text{Prob. success at } n \,|\, \text{first clash beyond } n)$$

$$= \sum_{n=1}^{\infty} x\,\bar{x}\,(1 - x^2)^{n-1} = \frac{\bar{x}}{x} \qquad 4.12$$

Thus,

$$Ex = \Theta + \Theta\left(1 + \frac{\bar{x}}{x}\right) + \frac{\Theta\bar{x}}{x}$$

$$= 2\Theta\left(1 + \frac{\bar{x}}{x}\right) \qquad 4.13$$

The throughput for the exponential system incorporating stabilising algorithm B is

$$S = \frac{2EX}{Mt} = \frac{4\theta(1 + \frac{\bar{x}}{x})}{\frac{1}{y} + \frac{2y(1 - y)}{x^2 (1 - \bar{y}^2)}} \qquad 4.14$$

By taking a similar approach the throughput for this system with a general distribution can be calcualted. Let θ' be the probability of a split in a particular direction for the general distribution

$$\theta' = P(W_1 < W_2) = P(W_2 > W_1) = \frac{1 - \sum_{n=1}^{\infty} f_n^2}{2}$$

The mean time between clashes can be calculated by considering the contributing components due to clashes and due to W_1 and W_2 splitting. The component due to a clash is given by equation 4.4, and the component due to splitting is given by equation 4.6.

Thus, for the general distribution

$$Mt = 2 \sum_{n=2}^{\infty} nf_n(f_1 + f_2 \ldots + f_{n-1}) + \sum_{n=1}^{\infty} nf_n^2 + \frac{1}{x^2}\left(1 - \sum_{n=1}^{\infty} f_n^2\right) \qquad 4.15$$

where the last component of this equation is $\frac{2\theta'}{x^2}$ due to the additional delay in the straight system before a clash.

In order to calculate the total number of successful transmissions in this period, the system will again be conditioned on $W_1 (W_1 = m)$, and successful transmissions will be calculated for all m. The number of successful $2 \rightarrow 1$ transmissions before W_1 is

$$1 \quad \text{if} \quad W_2 < W_1,\ W_1 = m$$

$$0 \quad \text{if} \quad W_2 > W_1,\ W_1 = m$$

$$\text{where } P(W_2 < W_1 \mid W_1 = m) = \sum_{n=1}^{m-1} f_n$$

thus the total number of transmissions before W_1 is $\sum_{n=1}^{m-1} f_n$

The number of successful transmissions after W_1 is

$$R \quad \text{if} \quad W_2 < W_1,\ W_1 = m$$

$$1 + R \quad \text{if} \quad W_2 > W_1,\ W_1 = m$$

$$\text{where} \quad P(W_2 > W_1 \mid W_1 = m) = \sum_{n=m+1}^{\infty} f_n$$

Thus, the total number of successful transmissions after W_1 is

$$\frac{\bar{x}}{x} \sum_{n=1}^{m-1} f_n + \sum_{n=m+1}^{\infty} f_n \left(1 + \frac{\bar{x}}{x}\right)$$

The total number of successful $2 \rightarrow 1$ transmissions conditioned on m is

$$(Ex \mid W_1 = m) = \sum_{n=1}^{m-1} f_n + \frac{\bar{x}}{x} \sum_{n=1}^{m-1} f_n + \left(1 + \frac{\bar{x}}{x}\right) \sum_{n=m+1}^{\infty} f_n$$

Thus, the total number of successful transmissions is

$$Ex = \sum_{m=1}^{\infty} f_m \, (Ex \mid W_1 = m)$$

$$= \left(1 + \frac{\bar{x}}{x}\right) \left(\sum_{m=1}^{\infty} f_m \, (1 - f_m) \right)$$

the throughput being given by equation 4.9. The uniform distribution with parameter L is a special case of the general distribution, and it can be shown that for this distribution

$$Mt = \frac{(L + 1)(4L - 1)}{6L} + \frac{L - 1}{x^2 L} \qquad 4.17$$

and

$$Ex = (1 + \frac{\bar{x}}{x})\left(\frac{L - 1}{L}\right) \qquad 4.18$$

the throughput being given by

$$S = \frac{12x(L - 1)}{x^2(L + 1)(4L - 1) + 6(L - 1)} \qquad 4.19$$

Similarly it can be shown that where the retransmission slot is chosen from the next two slots, and where the probability of choosing the first one is f_1 and the second one f_2, $f_1 + f_2 = 1$

$$Mt = 4f_2 f_1 + f_1^2 + 2f_2^2 + \frac{1}{x^2}\left(1 - f_1^2 - f_2^2\right) \qquad 4.20$$

and

$$Ex = (1 + \frac{\bar{x}}{x})\left(f_1(1 - f_1) + f_2(1 - f_2)\right) \qquad 4.21$$

It is of interest to derive the throughput as a function of the applied traffic per slot G. This can be obtained by considering Mt. As each clash represents two transmission attempts, the total number of transmissions including successes and failures in a Mt cycle is $2E_x + 2$. Thus, the applied traffic per slot G is

$$G = \frac{2Ex + 2}{Mt} \qquad 4.22$$

This leads directly to the ergodic probability of success for each transmission

$$P_{suc} = \frac{Ex}{Ex + 1}$$

and as the process is memoryless, the number of transmission attempts before success is $\left(\frac{1}{P_{suc}} - 1\right)$. To calculate the delay it will be assumed that for the uniform distribution the difference between retransmission and success delay components is small. Thus, the delay for the exponential case is given by

$$D = \frac{1}{y}\left(\frac{1}{P_{suc}} - 1\right) \qquad 4.23$$

for the general distribution by

$$D = \sum_{n=1}^{\infty} n f_n \left(\frac{1}{P_{suc}} - 1\right) \qquad 4.24$$

for the uniform distribution by

$$D = \frac{L + 1}{2}\left(\frac{1}{P_{suc}} - 1\right) \qquad 4.25$$

and for the distribution f_1, f_2; $f_1 + f_2 = 1$

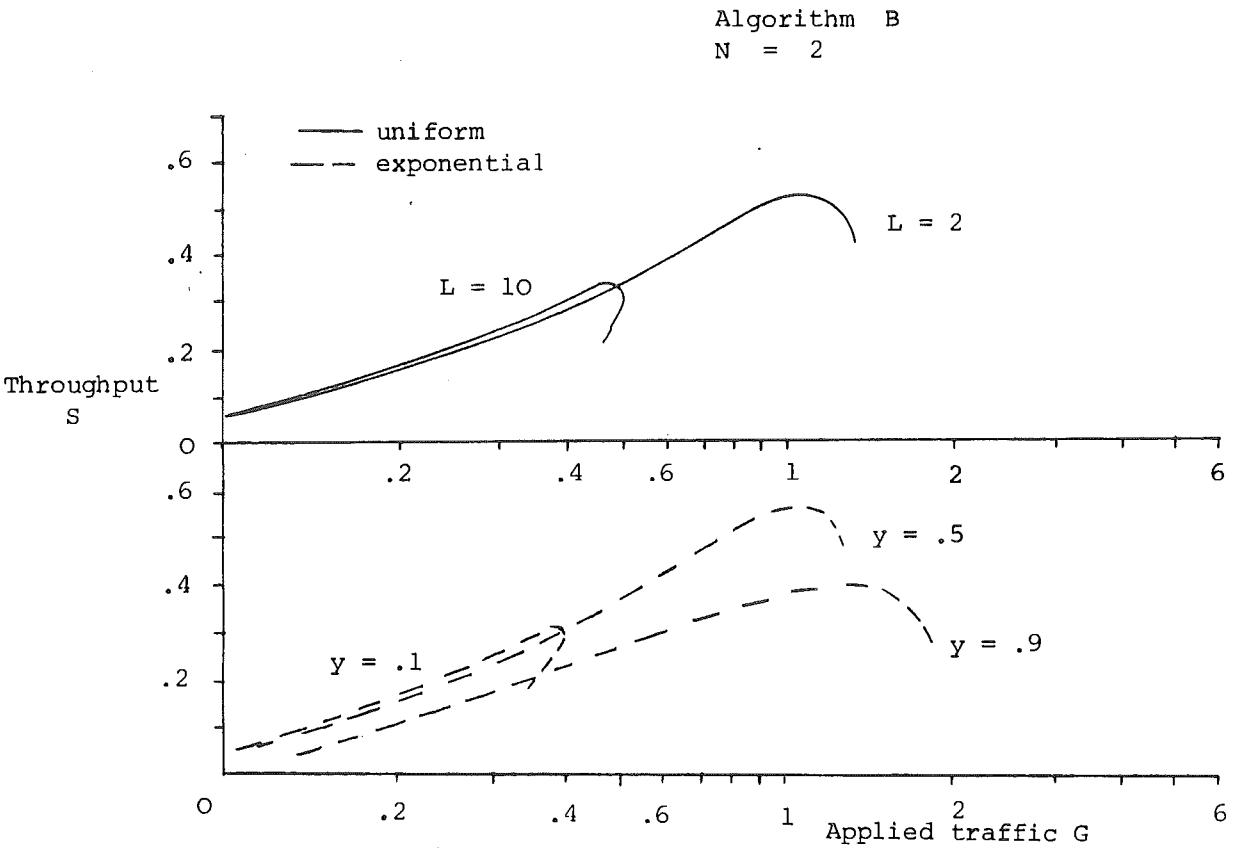
$$D = (f_1 + 2f_2)\left(\frac{1}{P_{suc}} - 1\right) \qquad 4.26$$

4.4.2 Graphs of analytical model

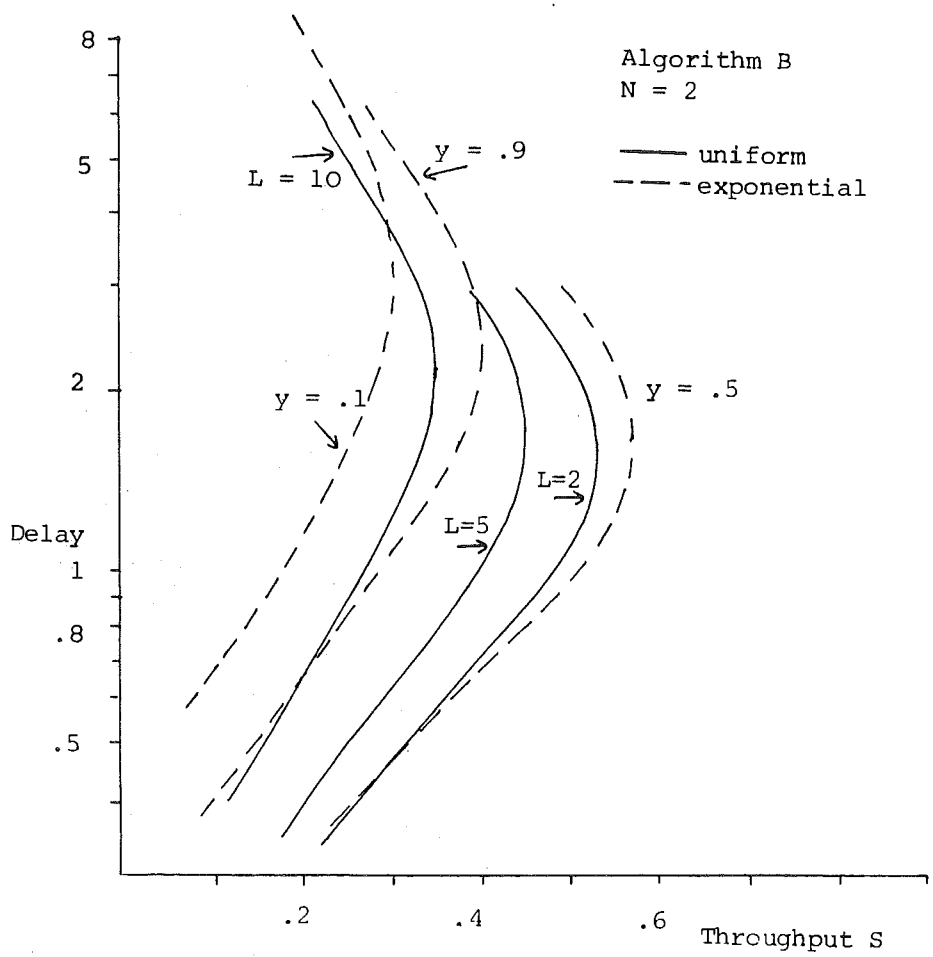
The equations derived in this section are plotted in graphs 4.2, 4.3 and 4.4. Graph 4.2 shows the throughput for the stable Aloha system against the probability of transmission x. It can be seen that as x increases, a peak in throughput is reached; and then as traffic becomes heavy, it falls off. The exponential model shows better throughput than the uniform model, but as has already been mentioned, this cannot be utilised in practice because for N > 2 the penalty for not transmitting before the system clash count is decremented is high. This graph will later be compared to a similar graph for the uncontrolled Aloha channel, and it will be shown that average throughput has been improved. Graph 4.3 shows the throughput against the applied traffic. It can be seen how the maximum value of the applied traffic has an upper bound due to stations not transmitting while blocked, and how this maximum increases with decreasing mean retransmission time. Graph 4.3, like graph 4.2, shows that there is an optimum value for the retransmission parameters at which throughput is at a maximum. Finally, graph 4.4. shows the tradeoff between delay and throughput. Normally, it would be expected that as the mean retransmission delay increases, the maximum value of the throughput would also increase. However, this is not the case, as the free station is blocked after its transmission for all time until the second station has transmitted. Thus, again there is an optimum value for the retransmission parameter which gives the best delay throughput characteristics.

4.4.3 Simulations of the stabilised Aloha channel

In order to test the correct behaviour of the equations derived in



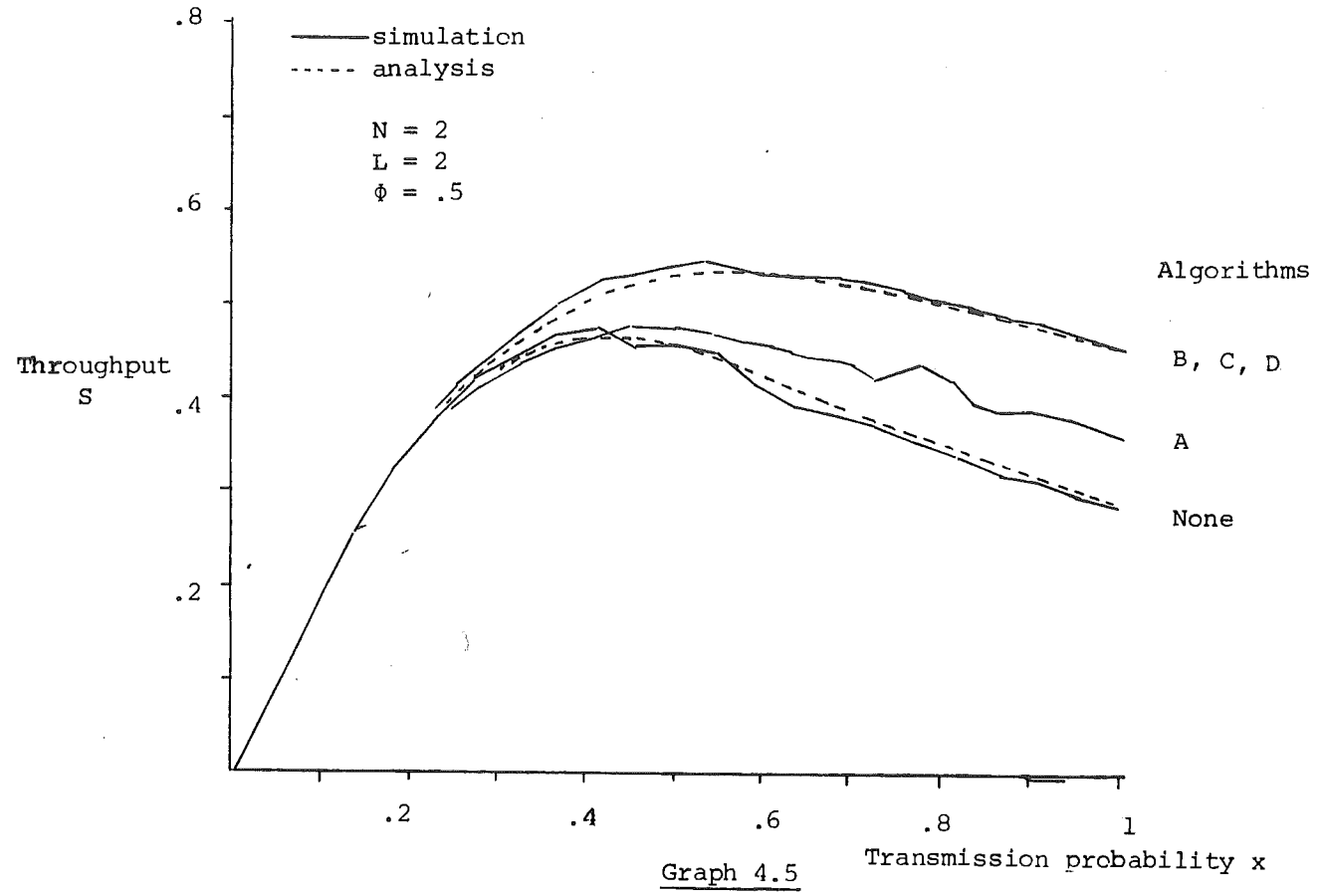
Graph 4.3



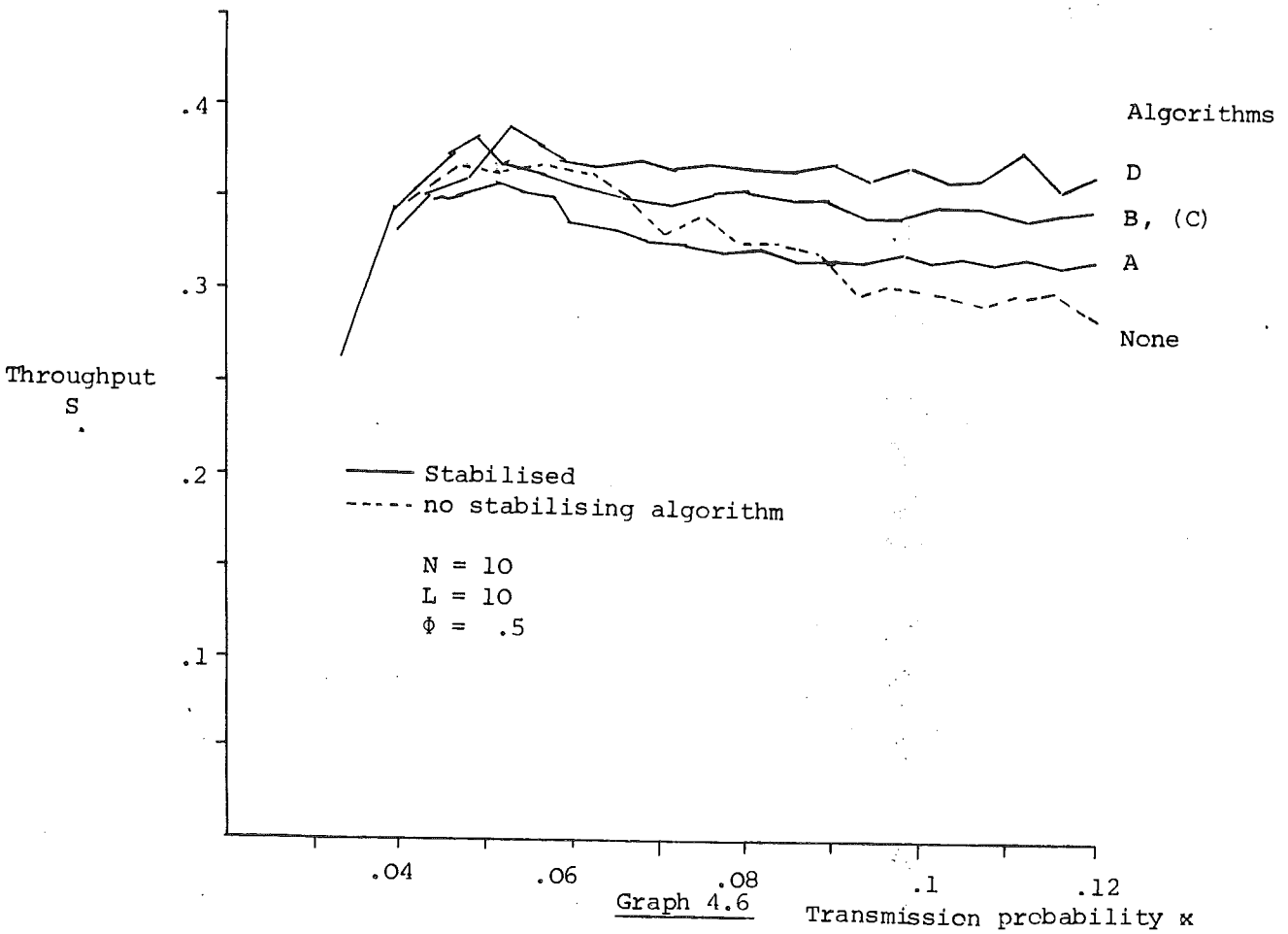
Graph 4.4

the previous section and to see the effect of increasing the number of stations, a simulation program has been written. Two types of assumption about new traffic were made,but it was found the difference between the two models was small. In the first a station transmits with probability x when it is free, and there is no buildup of packets while it is blocked. In the second model if a packet arrives while the station is blocked,it is stored and then transmitted with probability 1 when the station becomes free. Any other packets arriving during this time are lost. The latter is the same assumption as made by Kleinrock [KL75b] when calculating delay in the pure Aloha system with an infinite population of users. The different stabilising algorithms were simulated, as well as the uncontrolled slotted Aloha channel with L = N. In each case a station which has clashed chooses one of the next two slots with probability 0.5 (L = 2). The simulation also calculates the mean delay per transmission.

Graph 4.5 shows the throughputs of the stabilising algorithms as a function of the probability of transmission x. There is little difference between the models with and without packet buildup, and hence only the former will be considered. Note the good agreement between the analytical and the simulation results. Algorithm A shows some improvement in throughput over the uncontrolled system; and, as expected,algorithm B shows further gains. Because there are only two nodes in the system,there is no further change for algorithms C and D. In the graphs algorithms C and D include the effect of algorithm B. Graph 4.6 shows the effect of increasing the number of stations to 10. At low loads all systems behave similarly as the probability of a clash is low. However, as traffic increases,the throughput for the uncontrolled system begins to fall off, whereas algorithms A and B



Graph 4.5



Graph 4.6

behave as before. Algorithm C shows little change from algorithm B and is not shown, whereas algorithm D (Φ = .5) now shows the highest throughput. The maximum value of the system clash count increases with N, so it is to be expected that as N increases, so does the relative advantage of algorithm D. This is shown in graph 4.7 where N = 32 and at high loads algorithm D gives highest throughputs. As Φ is decreased from 1, the throughput at high traffic levels improves, until at some value of Φ it reaches a maximum and any further decrease has only an adverse effect on delay.

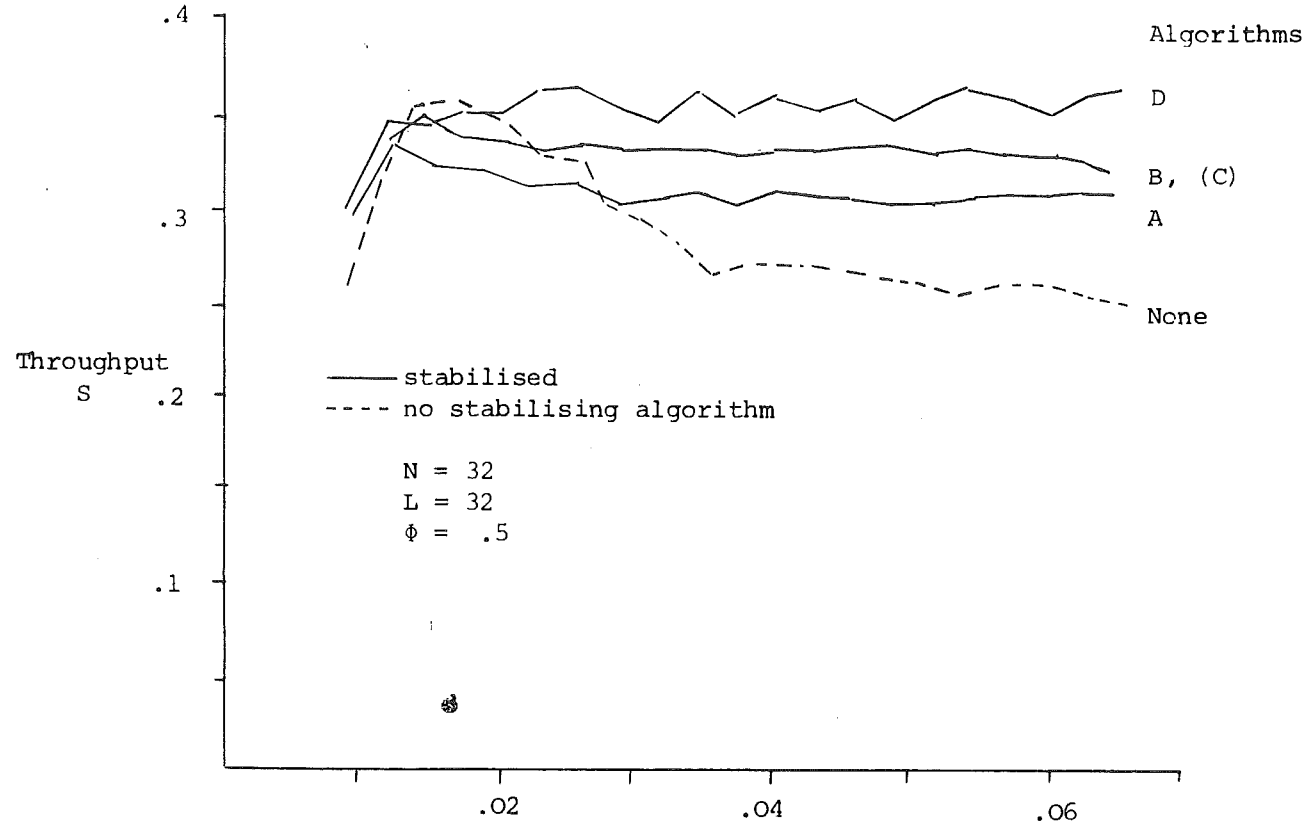
The simulations have shown that not only do the algorithms stabilise the Aloha channel, but that they also improve the performance, especially under high load conditions.

4.5 Performance improvement in the uncontrolled slotted Aloha channel

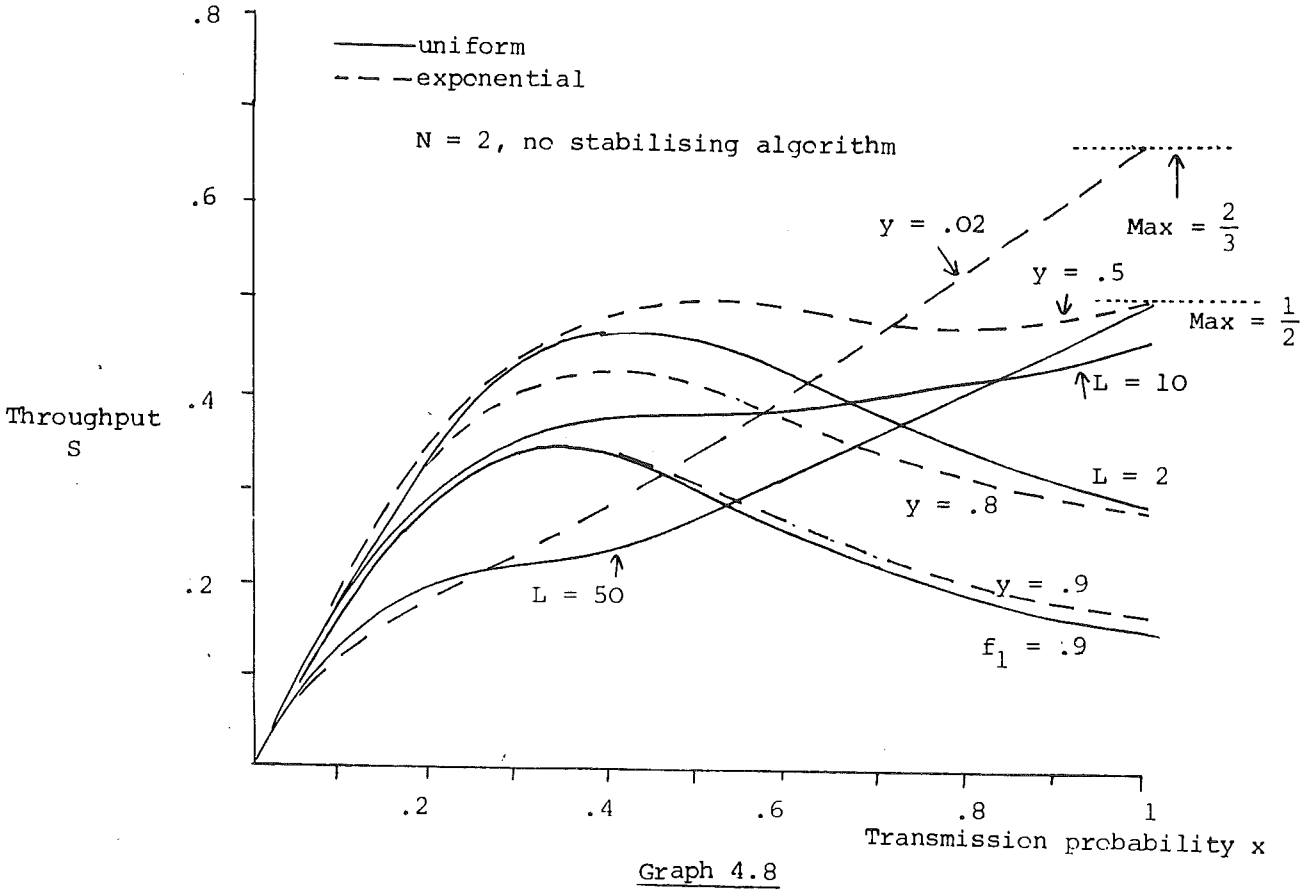
The equations and graphs of the previous section suggest that the exponential distribution improves performance for the two node system. It is thus of interest to develop a model of different retransmission distributions in the uncontrolled Aloha network and to see if any performance improvement extends beyond the case of N = 2. As no stabilising algorithms are being used, any retransmissions distribution can be used without penalty.

4.5.1 Analytical approach

The analytical model for the slotted uncontrolled Aloha network with two nodes is similar to the model described in section 4.4.1. However, when $W_2 \neq W_1$, some additional successful transmissions can take place, and also the transmission at $\text{Max}(W_1, W_2)$ can be interfered with by the free



Graph 4.7



Graph 4.8

station. The exponential retransmission distribution is considered first, and the mean length of a clash-clash cycle (Mt) is calculated. As before, the system is conditioned on W_1, and the different components of Mt are obtained. Since the expected value of the exponential distribution with parameter y is $\frac{1}{y}$, this is the delay component of Mt in all cases, except where the system moves beyond the point $\text{Max}(W_1, W_2)$ before a clash. This can happen if $W_2 < W_1$ and there is no $2 \rightarrow 1$ transmission at W_1, or if $W_2 > W_1$ and there is no $1 \rightarrow 2$ transmission at W_2. Thus, if $W_2 < W_1$ the additional delay component is $\frac{\Theta\bar{x}}{x^2}$, whereas if $W_2 > W_1$, the additional delay component (conditioned on W_1) is

$$\Theta \left(\frac{1}{y} + \frac{\bar{x}}{x^2}\right)$$

where the $\frac{1}{y}$ term is present due to the memoryless property of the exponential distribution. Thus, the mean time between clashes is given by

$$\begin{aligned} Mt &= \frac{1}{y} + \frac{\Theta\bar{x}}{x^2} + \Theta\left(\frac{1}{y} + \frac{\bar{x}}{x^2}\right) \\ &= \frac{1}{y}(1 + \Theta) + \frac{2\Theta\bar{x}}{x^2} \end{aligned} \qquad 4.27$$

where Θ is given by equation 4.11.

The number of successful $2 \rightarrow 1$ transmissions in this period is obtained as before by conditioning on W_1.
of successful transmissions before W_1. If $W_2 > W_1$ there are no successful transmissions before W_1, but if $W_2 < W_1$, a number of successful transmissions can take place after the first one as the station is not blocked by any clash counts. The number of such transmissions is given by

$$F = \sum_{n=1}^{\infty} P(2 \rightarrow 1 \text{ transmission at } n | W_1 > n)$$

$$= x \sum_{n=1}^{\infty} \bar{y}^n = \frac{x\bar{y}}{y}$$

Thus, the total number of transmissions for $W_2 < W_1$ before W_1 is

$$\theta \left(1 + \frac{x\bar{y}}{y}\right)$$

Let us now look at the number of successful transmissions after W_1. If $W_1 < W_2$ and there is no $1 \rightarrow 2$ transmission at W_2, then there is one successful $2 \rightarrow 1$ transmission at W_2, plus R further transmissions before a clash (equation 4.12). If $W_2 < W_1$ and there is no $2 \rightarrow 1$ transmission at W_1, then there are R further transmissions before a clash. Thus, the total number of successful $2 \rightarrow 1$ transmissions is given by

$$Ex = \theta\left(1 + \frac{x\bar{y}}{y}\right) + \theta\bar{x}\left(1 + \frac{\bar{x}}{x}\right) + \frac{\theta\bar{x}\bar{x}}{x} \qquad 4.28$$

$$= \theta\left(1 + \frac{x\bar{y}}{y} + \bar{x} + \frac{2\bar{x}^2}{x}\right)$$

Having obtained Mt and Ex the throughput and delay may be obtained as before (equations 4.9, 4.22).

The values of Ex and the Mt for the uncontrolled slotted Aloha system with a general retransmission distribution are now derived. The value of Mt is similar to that for the stabilised system (equation 4.15), with the addition of one component due to the free station now not being

blocked for transmitting. Thus Mt is given by

$$Mt = 2 \sum_{n=2}^{\infty} nf_n (f_1 + f_2 \ldots + f_{n-1}) + \sum_{n=1}^{\infty} nf_n^2 + \frac{\bar{x}}{x^2}\left(1 - \sum_{n=1}^{\infty} f_n^2\right)$$

4.29

To calculate the number of successful 2 → 1 transmissions in this period, again condition on W_1 ($W_1 = m$) and sum for all m. Thus, the number of transmissions before m (with probability 1) is given by the sum of probabilities that $W_2 < W_1$ plus all other transmissions which take place with probability x before $W_1 = W_2$. Thus, the number of transmissions before m is

$$(Ex^b | W_2 < W_1) = \sum_{n=1}^{m-1} f_n + \sum_{n=1}^{m-2} f_n (m - n - 1)x \qquad 4.30$$

The number of transmissions after m is given by the probability that $W_2 < W_1$ multiplied by the probability that no 2 → 1 transmission took place ($\bar{x}$) at $W_2 = W_1$ followed by the straight system R

$$(Ex^a | W_2 < W_1) = \sum_{n=1}^{m-1} f_n \frac{\bar{x}\bar{x}}{x} \qquad 4.31$$

The number of transmissions after m when $W_2 > W_1$ is given by

$$(Ex^a | W_2 > W_1) = \bar{x}\left(1 + \frac{\bar{x}}{x}\right)\left(1 - \sum_{n=1}^{m} f_n\right) \qquad 4.32$$

Thus, the total number of successful transmissions is obtained by summing equations 4.30, 4.31, 4.32 over m.

$$Ex = \sum_{m=1}^{\infty} f_m \left\{ \sum_{n=1}^{m-1} f_n + \sum_{n=1}^{m-2} f_n(m - n - 1)x + (1 - f_m) \frac{\bar{x}^2}{x} + \bar{x}\left(1 - \sum_{n=1}^{m} f_n\right) \right\}$$

4.33

It can now be shown that for the uniform distribution with parameter L

$$Ex = \frac{L - 1}{Lx}\left(\frac{x^2(L + 1)}{6} - x + 1 \right) \qquad 4.34$$

$$Mt = \frac{(L + 1)(4L - 1)}{6L} + \left(\frac{L - 1}{L} \right) \frac{\bar{x}}{x^2} \qquad 4.35$$

and for the distribution f_1, f_2; $f_1 + f_2 = 1$

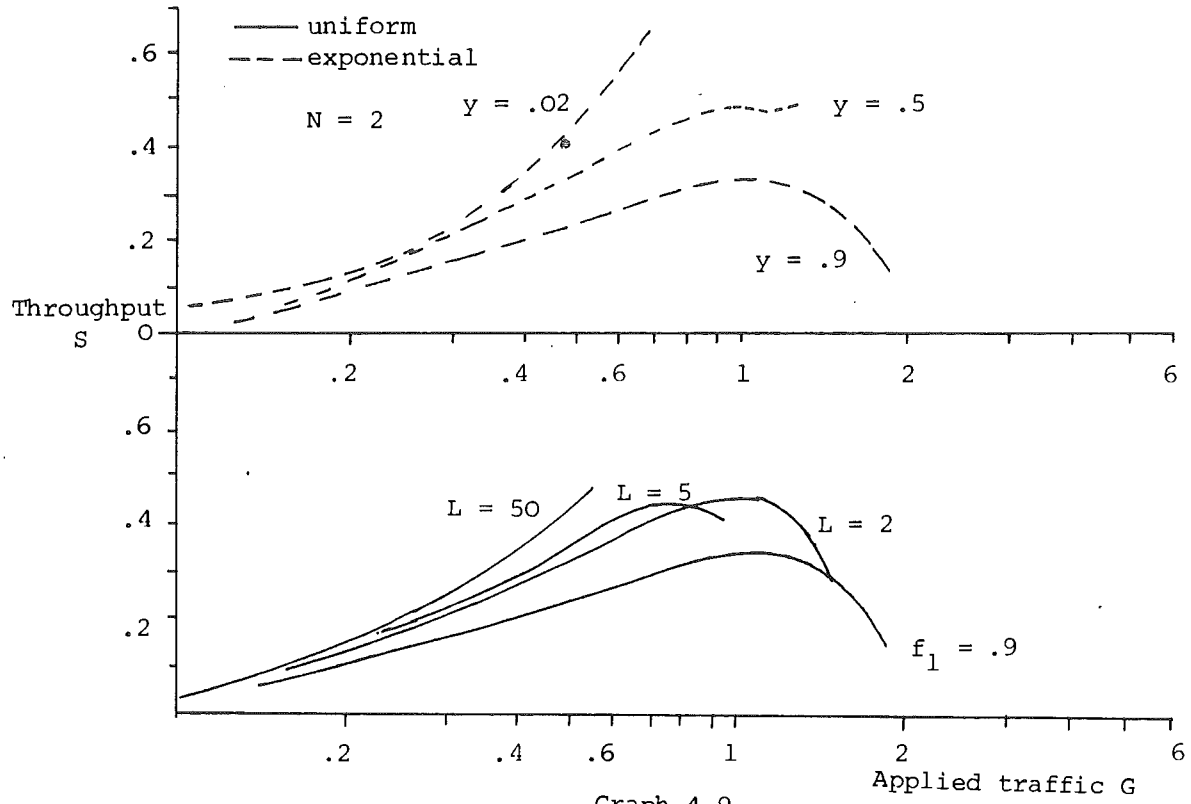
$$Ex = f_1 f_2 + \frac{\bar{x}^2}{x}\left(f_1(1 - f_1) + f_2 (1 - f_2) \right) + 2\bar{x} \left(1 - f_1 - \frac{f_2}{2}\right)$$

4.36

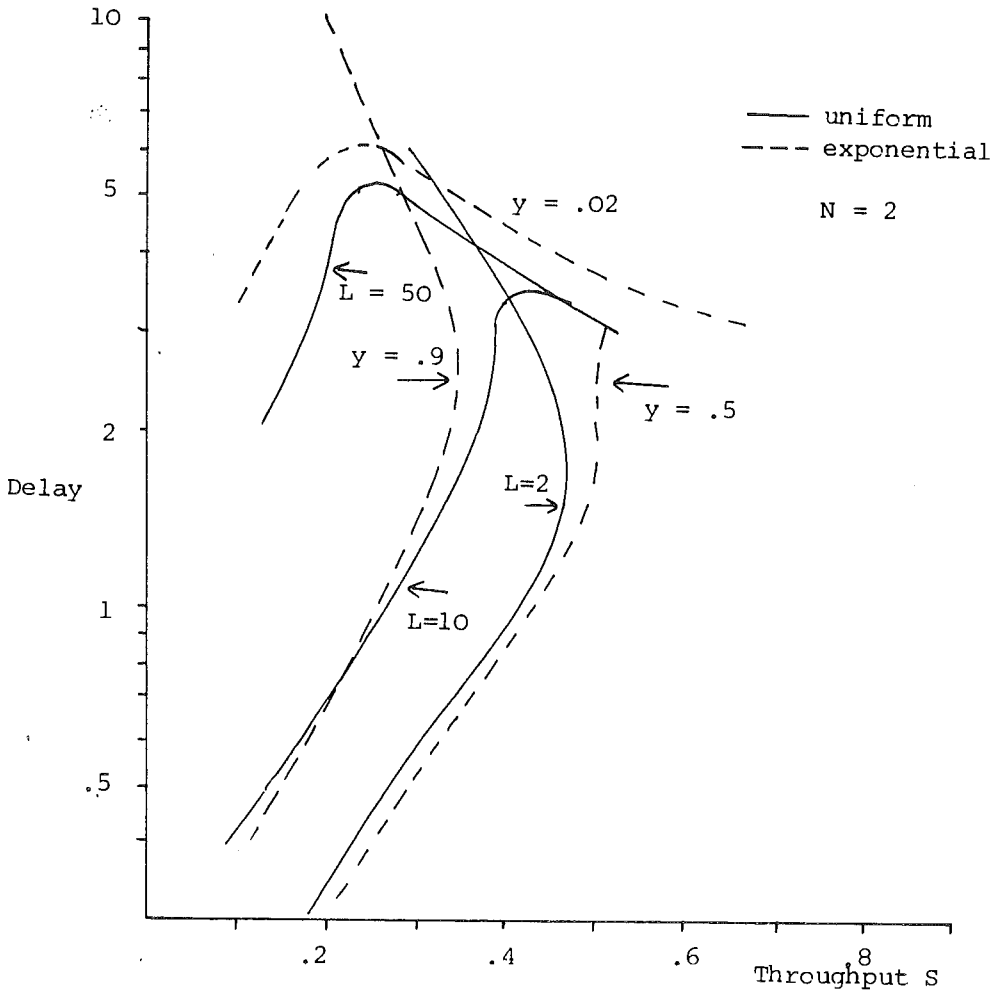
$$Mt = 4f_2 f_1 + f_1^2 + 2f_2^2 + \frac{\bar{x}}{x^2} (1 - f_1^2 - f_2^2) \qquad 4.37$$

4.5.2 Graphs and analytical model

The formulae obtained for the uncontrolled Aloha channel have been verified by simulation and are plotted in graphs 4.8, 4.9 and 4.10. Graph 4.8 shows the throughput against the probability of transmission x. It is noticeable how the central peak is now lower compared to that for the controlled Aloha channel (Graph 4.2), and how at high loads the



Graph 4.9



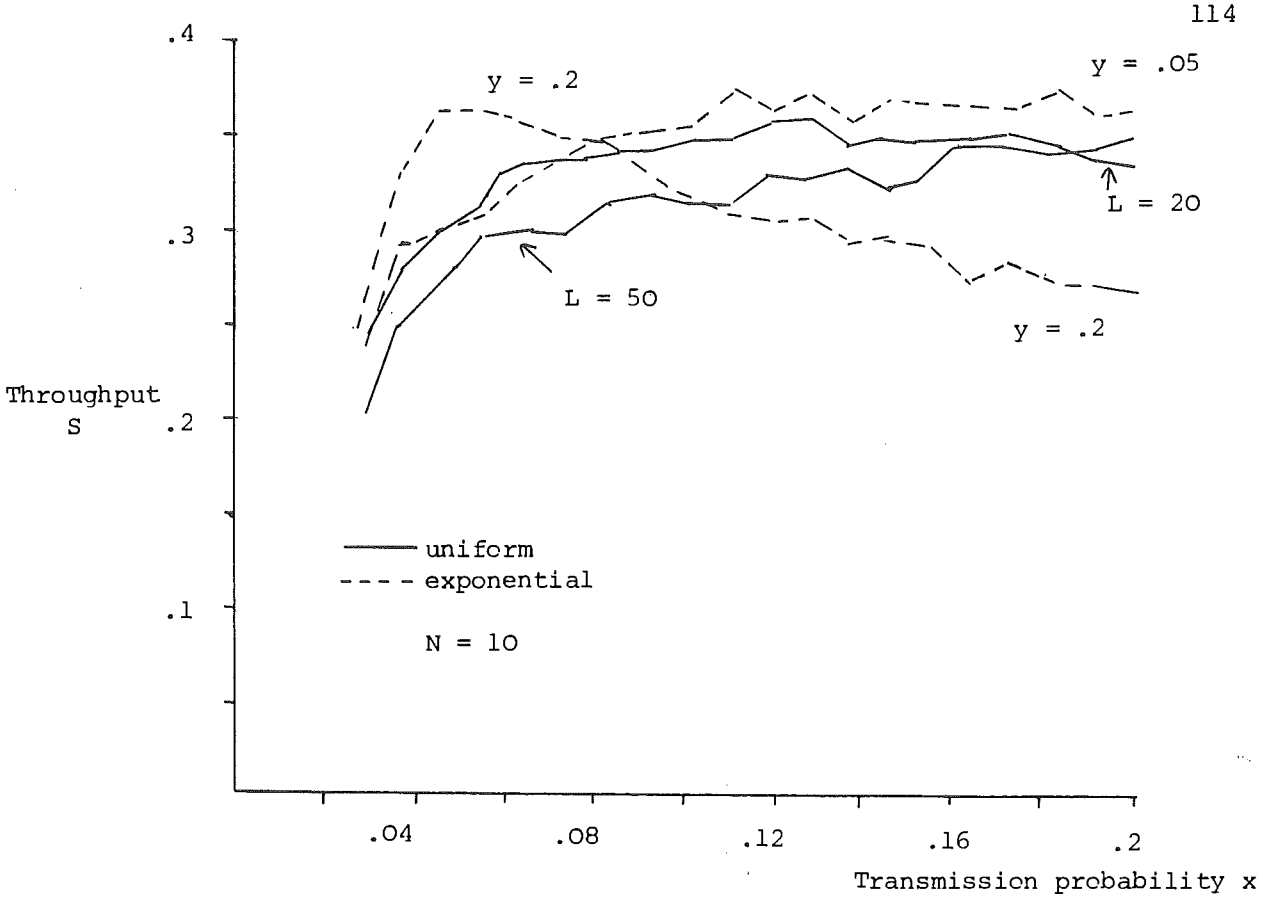
Graph 4.10

throughput increases. The increase in high load throughput is especially noticeable for large L, which indicates that this is due to a large number of transmissions taking place when there is a large difference between W_1 and W_2. Graph 4.9 shows throughput against applied traffic, and again the exponential model shows good performance characteristics. Finally, Graph 4.10 shows the delay as a function of throughput and has some unusual features. As expected, the delay increases with throughput; however, there comes a sharp turning point where the delay begins to decrease with throughput. This can be considered to be hogging; and as the number of transmissions between W_1 and W_2 increases, so the average delay decreases. With increasing N this effect tends to disappear for a given value of L, although it will occur at any N providing L is sufficiently large.

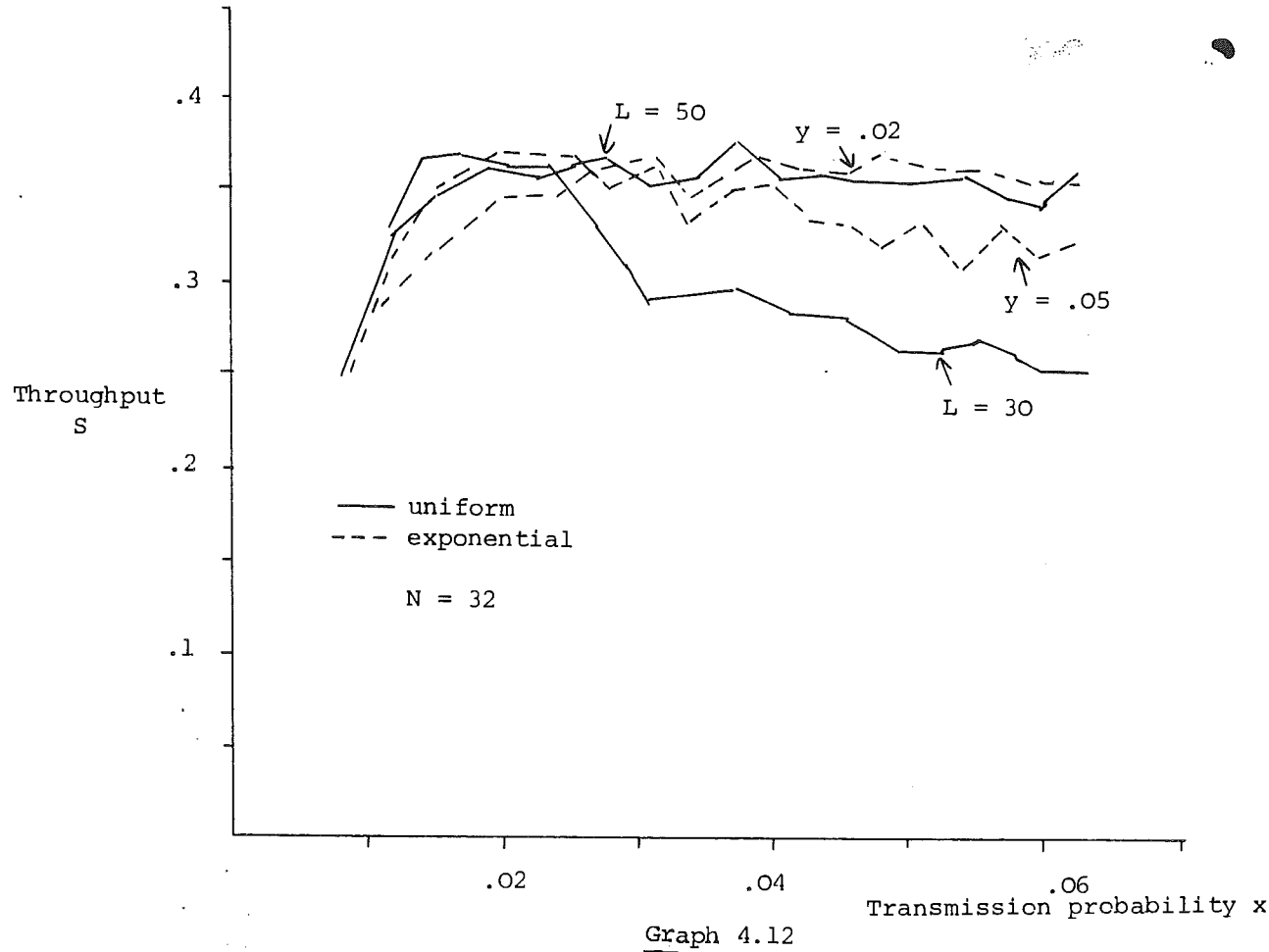
4.5.3 Simulations of the exponential and uniform distributions for the slotted Aloha channel

A number of simulations were carried out to compare the maximum throughput obtained using a uniform retransmission distribution and an exponential retransmission distribution. The assumptions were the same as those described in section 4.4.3.

The analytical and simulation results were compared (graph 4.5), and agreement was good. Graph 4.11 shows the throughputs for the two systems with optimum values for retransmission parameters, and N = 10. Although the exponential distribution still shows the highest throughput, the difference between the two systems is now smaller. This suggests that as N increases, the gain in using the exponential distribution decreases. This is borne out by graph 4.12, which shows that for a system with 32 nodes the difference between the two systems is small.



Graph 4.11



Graph 4.12

It has been shown that for a system with a small number of nodes an exponential retransmission distribution gives higher throughputs than the uniform distribution, but that as N increases, the difference between the two retransmission schemes becomes small.

4.6 Carrier Sense Multiple Access Networks

Networks where the channel is sensed before transmission are now considered. Once the channel has been acquired, the transmission is not interfered with, and thus throughput can be considerably improved by making packets large and variable in size. This, however, has the undesirable effect that for very large packets hogging can reappear. A new CSMA network is described below, followed by a description of a technique for modelling CSMA performance when the number of nodes and the retransmission parameter L are finite.

4.6.1 The ring contention network

The ring contention network has been proposed independently at the University of California Irvine, M.I.T., and Cambridge. The particular variant described here has better performance characteristics than the other schemes due to the very small delay through each node.

One of the most difficult problems encountered when designing a contention network is to detect the presence of another transmission during the vulnerable period. As the length of the cable between the furthest points in the network increases, the difference in strength between the local transmission and the distant colliding transmission can become large, for example in the Ethernet the distant signal is 20 DB weaker. This can be overcome by using an undirectional ring in a contention mode as the transmission medium. Consider such a ring under

very light loads. When a station wishes to transmit, it senses the ring for the presence of another transmission, and if the ring is free it transmits its own packet. After one ring delay, the packet will begin to be received back, and the transmission can proceed uninterrupted. If it is arranged that in the quiescent state there are zeros on the line, and the length of the ring is known, then a collision can be detected if a one is received at the transmitter before one ring delay has elapsed (SOP = 1). This is a digital operation and can thus be performed easily.

If when the station wishes to transmit the ring is busy, the transmission is postponed until the end of the current packet. There are a number of ways of indicating the start and end of a packet- for example, by use of a unique token and bit stuffing, but in the present system it will be assumed all stations are synchronised with packets by counting ring delays. The length and address information are specified at the head of the packet. A station which is deferring due to another transmission proceeds to concatenate its transmission at the end of the current packet train. This is done by arranging that the bit following the SOP bit is the full/empty bit. When a station wishes to transmit, it overwrites this bit at the same time testing for empty. If it is found empty, the transmission proceeds; otherwise the algorithm is repeated after the next SOP bit. By using such a scheme, the delay through each node is reduced to one gate propagation delay.

It can thus be seen that under light loads the delay in transmitting a packet is very small since these are no non-alignment delays. As traffic increases, the stations tend to become synchronized to the packet train and fewer collisions occur. Thus, an order is placed on the stations wishing to transmit (which is <u>not</u> pre-determined) and performance is

improved, there being no collisions when the system is operating at full capacity. Furthermore as for high loads the probability of a clash reduces to zero ,the system is stable. The scheme can be extended, so on a clash each station continues transmitting with probability 0.5.

Advantages of the contention ring over a normal ring are that there is now no need to provide a mechanism for setting up the slot structure on turn on (or a lost token restoring mechanism) and that delays at low loads are small. The advantage of using the counting scheme, rather than a pair of unique packet delimiters, is that delay through each node is reduced from one bit time to a gate propagation time. This means that for large systems there is no penalty in using the contention ring over a bus system since line delays and control establishment times (in terms of the vulnerable period) are the same. It can further be argued that because the real time to establish control in the normal contention system is twice the propagation delay, whereas for the contention ring it is a single propagation delay, the latter is better.

4.6.2 Analytical modelling of the CSMA network

In this section a technique for modelling CSMA networks similar to the Ethernet is given. This technique is analytical in nature but requires large amounts of computer resources and thus can be impractical, except where the number of nodes in the system is small.

The CSMA network behaves in the same way as the Aloha network during the vulnerable period : that is, transmission may be successful, clash, or there may be a gap. Let $F_k (k = 0, 1 \ldots N)$ represent the probability that k free stations transmit. Thus, in terms of previously defined parameters

$$F_k = {}^{N}C_k \; x^k \; (1 - x)^{N-k} \qquad 4.38$$

In a similar way, let $Q_k (k = 0, 1 \ldots N)$ be the probability that k stations clash and are rescheduled, and $H_k (k = 0, 1 \ldots N)$ be the probability that k previously blocked stations retransmit. Thus

$$\begin{aligned} Q_0 &= F_0 H_0 \\ Q_1 &= 0 \\ Q_k &= \sum_{A=0}^{A=k} f_A H_{k-A} \end{aligned} \qquad 4.39$$

Finally, let D_1 represent the probability that a station is successful in acquiring the slot.

$$D_1 = F_0 H_1 + F_1 H_0 \qquad 4.40$$

When a station clashes, it retransmits in one of the following L slots with equal probability (uniform distribution). In order to calculate the probability H_k that k blocked stations retransmit in a slot, consider the probability of finding z balls in a specified box when k such balls have been distributed in L boxes. This probability is given by Feller [FEL50]

$$P_z = {}^{k}C_z \; \frac{1}{L^z} \left(1 - \frac{1}{L}\right)^{k-z}$$

Thus ,

$$H_z = \sum_{\substack{k=z \\ k\neq 0}}^{k=N} Q_k \, {}^{k}C_z \, \frac{1}{L^z}\left(1 - \frac{1}{L}\right)^{k-z} \qquad 4.41$$

$$H_0 = Q_0 + \sum_{k=z}^{k=N} Q_0\left(1 - \frac{1}{L}\right)^{k}$$

By substituting equation 4.39 in equations 4.41, N non-linear simultaneous equations are derived which can be solved as

$$\sum_{k=0}^{k=N} H_k = 1 \qquad 4.42$$

In the above model packets are of the same size as slots (slotted Aloha). This can be extended by introducing a parameter W which represents the number of slots the transmission is continued on beyond the first slot. The probability of a successful transmission is now given by

$$D_1 = (1 - D_1)^W \, (F_1H_0 + F_0H_1) \qquad 4.43$$

as there can only be one successful transmission for the duration of a packet, all other transmissions being rescheduled. The probability of one station being rescheduled exists and is given by

$$Q_1 = (F_1H_0 + F_0H_1) \quad WD_1(1 - D_1)^{W-1} \qquad 4.44$$

In a system with an infinite number of nodes, new arrivals F_k are not conditioned on the blocked arrivals H_k. However, for a finite N

the number of new arrivals is dependent on the number of blocked stations so that

$$\sum kH_k + \sum kF_k \leq N \qquad 4.45$$

Equation 4.45 defines the channel load and can be incorporated in equations 4.38 to 4.42 by assuming that the channel backlog and the combined input rates are linearly and inversly dependent. This is a linear feedback model and is similar to the one used by Kleinrock [KL75b].

The solution of the non-linear simultaneous equations 4.38 to 4.42 is a prohibitively large computational task if attempted by hand. However, by using a computer algebra system the process can be mechanised. Programs have been written in the CAMAL algebra language to derive the simultaneous equations which are then solved by standard library routines. Since the H_k's define probabilities, their values are numerically similar, and thus the non-linear simultaneous equations are well conditioned, and convergence to a solution is rapid. However, computer algebra systems are inefficient, and on a machine with no virtual store such as the IBM 370/165 at Cambridge, it is soon found that maximum storage (400K) is exceeded.

The above method calculates the probability of acquiring the channel in a CSMA network in terms of the number of stations N, the retransmission parameter L, and the probability of a free station transmitting x. It was found that for small N the results agreed with previous work, but that for N > 4 the expressions grew beyond store size.

4.7 A comparison of broadcast and ring networks

Having modelled different types of ring and broadcast networks, it is of interest to see if there are any major differences between the schemes.

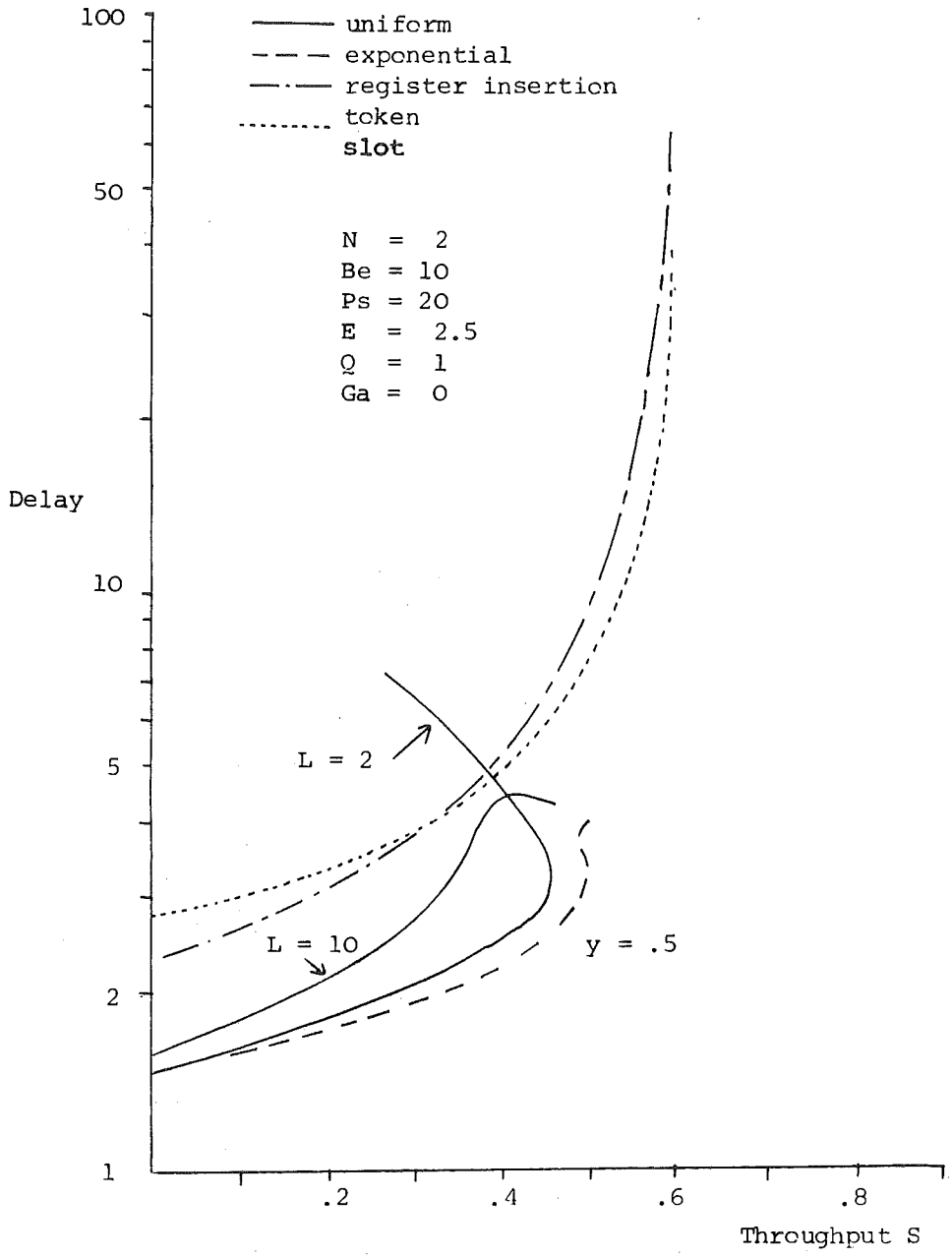
In terms of physical characteristics both systems use a single wire to implement the network. This wire tends to be longer for rings as stubs, and discontinuities require only a single branch for the broadcast scheme (but such extensions will add to the reverberation and attennation problems). If the communicating devices are located in physically isolated locations it is advantageous to use radio as the transmission medium. It can be argued that it is easier to add nodes to the broadcast system because they are passive and do not have to regenerate the signals; in any case, it is likely that the propagation delay will be smaller for broadcast than for ring. On the other hand, the hardware of the broadcast system has to provide a collision detect mechanism, perhaps a stabilising mechanism, and sometimes a two way repeater to prevent the signals from becoming too faint.

In terms of efficiency, it is the propagation delay which is the critical factor in all systems. This has a lesser effect on the empty slot ring than on the others, but even there it is of importance. The difference between the efficiency characteristics shows up when the effect of N is considered. For the broadcast scheme, as N increases, the maximum attainable throughput tends to decrease. On the other hand, for the ring systems, and especially for the slot system, the maximum efficiency tends to increase with N. However, for reasonable values of all system parameters, the maximum throughputs are closer than would initially be envisaged and are governed by the propagation delay. Efficiency of all systems is also influenced by packet size and improves

for long packets.

The delay throughput characteristics of the slotted Aloha and register insertion systems are shown for N = 2 in Graph 4.13. Although this is a special case with a small number of nodes, the performance characteristics for larger N will not be significantly different. The assumptions for the Aloha system are the same as those for the insertion system, although only one way transmissions are considered, and thus no low level acknowledgement is returned to the source. It can be seen that for low loads the Aloha system shows lower delays as the transmitted packet is not received back at the source. As the throughput increases, however, the Aloha system shows a faster rate of increase of delay than the ring and becomes saturated at a lower value of throughput. The register insertion system, on the other hand, reaches a higher maximum throughput but at a larger delay. The other ring system exhibit similar behaviour to the insertion system. For larger N it is to be expected that the performance curves at low loads will move closer together, but at high loads the ring system will continue to be better. Two further advantages of ring systems are that they are stable and that the effect of transients is well defined.

In terms of reliability and errors the systems are again similar, a break in the wire precipitating a complete breakdown. It is easier to detach a broken node in the broadcast system as it is passive and does not regenerate signals. However, for ring systems these components can be made very reliable and unlikely to fail. There is an advantage in broadcast systems in terms of errors. Since the network is in any case error prone due to collisions, the software and hardware are designed with this in mind. If an error occurs then this


100
50
10
5
2
1
Delay
uniform
exponential
register insertion
token
slot
N = 2
Be = 10
Ps = 20
E = 2.5
Q = 1
Ga = 0
L = 2
L = 10
y = .5
.2
.4
.6
.8
Throughput S

Graph 4.13

has little effect (e.g. Ethernet), and the transmission can be restarted at the appropriate level. In the ring systems an error in the data field poses few problems, but an error in a control field has to be restored by hardware, and this procedure is normally not elegant. The error rate on the Cambridge ring is comparable with internal computer error rates, and thus the mechanism for restoring the frame structure is called into operation very infrequently.

The conclusion to be drawn when comparing broadcast and ring systems is that their performance characteristics are similar, especially under low loads. Thus preference will be governed by other factors, such as the physical environment, the characteristics of the communicating devices, and the grain at which transmission is required.

CHAPTER 5

THE DESIGN OF A LOCAL NETWORK LSI CHIP

5.1 Introduction

This chapter deals with the design of a general purpose, multi-network LSI chip. The chip can operate in a number of ways defined by a microprogram.

Local networks operate in a way which does not depend upon the characteristics of any particular device, nor on the data being transmitted. They are capable of handling a general purpose data link control architecture (such as IBM's SDLC) and are able to operate over a large range of propagation delays. This means that they are designed at the bit rather than the character level, that they are capable of operating at high speeds, and that they do not require sophisticated interfaces to the communicating devices. On the other hand, one of the most important decisions to be made is to what extent control functions are performed by the network itself, and to what extent they are passed on to the attached host. Thus some hardware level protocols are defined, which in turn are influenced by the traffic characteristics and by the speed of the network. In the past this led to the design of special purpose logic which was then difficult to change. Universal Synchronous Asynchronous Receivers Transmitters (USARTs) have been designed and manufactured in LSI, but as they do not implement a specific network architecture additional logic is required. This logic can become complex, especially if the basic protocols are to be extended to include more flexible addressing and control structures and acknowledgement of packets. By using a microprogrammed chip it is easy to tailor

the architecture of the network to the application and to similar foreseeable applications, so that some developments are readily implementable.

5.2 The environment of a local network chip

The nodes in a local network can be interconnected in a number of ways. The completely connected network offers the lowest delays but is also expensive. A star network depends upon a fast central switch for routing but has fewer connections. The simplest is the ring network, which has no routing problems but the largest theoretical delays. Other networks which do not perform any routing are the contention network (e.g. Aloha) and the carrier sense multiple access network (e.g. Ethernet).

Once the structure of the network has been chosen, its mode of operation has to be defined. One of the most important areas is that of addressing. There are two basic addressing schemes; position(time) addressing and code addressing. In the former each port has a number of permanently assigned slots and cannot use others. In the latter each packet is prefixed by an address which defines the user of the slot at that time. Code addressing has the advantage of being flexible but does take up extra bandwidth. With fixed assignment of slots the characteristics of operation of the network are better known.

The communication process between two nodes will generally consist of the transmission of a number of packets. If this is known in advance, then a call can be setup. This can be done in three ways; by using a 'start of call' 'end of call' protocol, by specifying a 'start of call' and a count number, or by a continuation marker in each packet. In every case the destination is made deaf to all but the originator of

the call for some period of time. It is now possible to shorten the address fields of packets since it is only necessary to distinguish between the maximum number of setup calls, and the surplus bits can be used for data. This can be extended to include conferencing where the packets are read by the destination and re-addressed somewhere else, or by forwarding where they are only re-addressed. If process to process addressing is used (section 2.2.2), the location of a particular process does not have to be known in advance. However, such 'processes' are only likely to move between nodes if they correspond to simple entities such as file names.

In a simple network a source node is only allowed to transmit to one destination at a time. Such a point-to-point mode of operation can be extended to allow multiplexing of packets to different destinations. This affects the way acknowledgements are handled and the way that the control bits are used. If the source does not know the speed characteristics of the destination, then the network should not allow it to transmit in such a way that congestion develops. One way of doing this would be for the destination to signal when it is ready to receive again. This is achieved neatly in the register insertion system by temporarily taking the packet out of the ring at the destination; however, both communicating partners must have their registers inserted. When a large amount of data is being transmitted between a pair of nodes, the acknowledgement traffic can be reduced by acknowledging on a per block rather than per packet basis. Even more complicated systems can be devised where more than one packet can be in flight to a particular destination, in which case the acknowledgements have to contain sequence numbers, and care has to be taken with respect to the build up of undelivered traffic.

For contention type networks, errors are assumed to occur

frequently, and powerful error detection facilities are provided, whereas ring type networks can be made relatively error free. Thus, for the contention network it is essential to be able to restart a packet at the source, while this need not be done for the ring. If the transmitted packet is retained, then the operation of the network can be made autonomous with respect to the host, and most hardware errors can be hidden from the outside world.

In some applications the services of a particular node might be required extensively. If the calls are setup, then that node might be blocked from some sources for considerable periods of time. Under these circumstances it is advisable to implement an algorithm where transmission requests to a busy destination node are queued rather than ignored and later arbitrated on a random basis.

Further choices include those between duplex and half-duplex operation and between synchronous and asynchronous transmission. The costs of changing software in the host machines have to be considered: if these are high, then a transparent protocol might be used.

5.3 Hardware and interface design

The hardware design goals for a local network are that it should be reliable, cheap, have few wires, and consist of distributed identical units. It should not be dependent upon any modulation system or clocking technique. For these reasons it is desirable to treat the repeater (or regenerative transmission) section of the node and the logical section (which deals with the operation of the network) separately. The interfaces between these have to be defined, as well as the interface to the local host. The logical section shall be referred to as the

station logic chip (SLC). The structure of such a system is shown in Fig. 2.2.

The network interfaces to the repeater consist of the input-from-medium and output-to-medium wires which carry the signal between the nodes of the communications system. The interface to the SLC consists of data out, data in, one control line for 'external data in' as well as clock and power lines. One of the differences between contention type systems and ring type systems is that the former need to detect when a collision has occurred. The way this is done is dependent on the transmission scheme being used. Since the SLC is capable of operating in a contention mode,the repeater has to provide this 'collision detected' signal,and this is shown as one other control line on the interface (Fig. 5.1). This interface has to operate in a fail safe manner so that if the SLC malfunctions,it can be disconnected and the repeater will continue to operate normally.

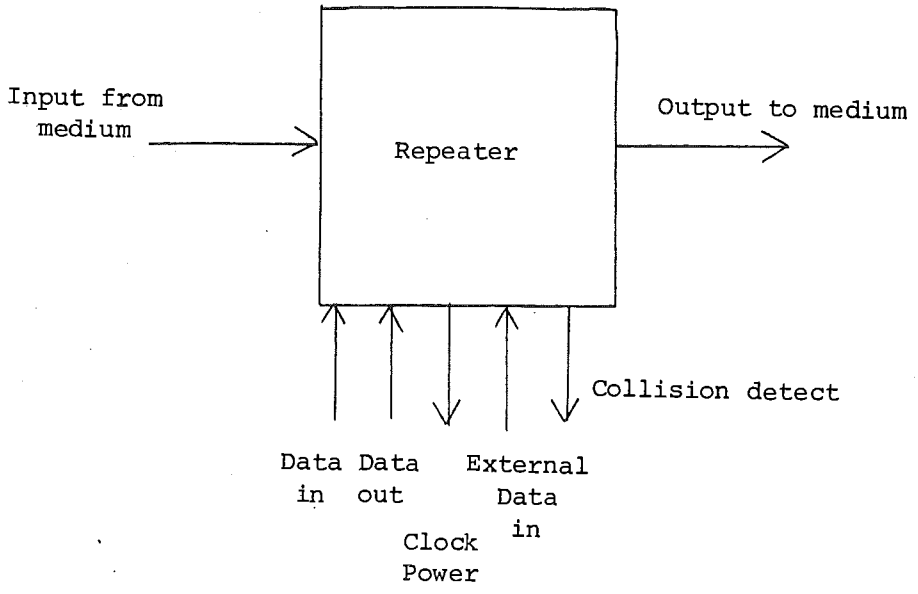


Fig. 5.1 Repeater Interfaces

The interface between the SLC and the access box will generally link asynchronous devices. This means that under some circumstances enough time has to be allowed for flip flops to settle before being sampled. Another important constraint is the pin count,which will make it necessary to multiplex data along a bus. Thus, on the receive side, the interface will consist of the data/address bus and the control lines for packet received, source address enable, data enable and disenable, status enable and packet complete; with send packet, source read, data read, next byte, status read,and receive resume being sent from the host (Fig. 5.2). The status and error signals can be output on a separate bus or on the data/address bus. On the transmit side these signals are similar but are originated by the source. There are a number of further signals which are used for loading the microprogram store at start up time, and these are shown in Fig. 5.3. Thus,the SLC interface is controlled by external gating signals which reduces the number of bus wires. If a number of sixteen bit machines are being interconnected eight bit buses might prove inconvenient, but since a separate access box will be built for each type of host,the added inconvenience should not be high. Although an error signal is provided, it will not be used in cases where the manufacturer does not provide checking on the I/O bus.

It is unlikely that enough space will be available on the SLC to provide full size buffers for the largest packets and an extensive address table. Further interfaces are thus defined for extending the packet buffer to a FIFO buffer chip and the address table to an associative address chip (NTEC). The structure of such a system is shown in Fig. 5.4. The SLC/FIFO interface is shown in Fig. 5.5. The associative address chip architecture is shown in Fig. 5.6 and its interface to the

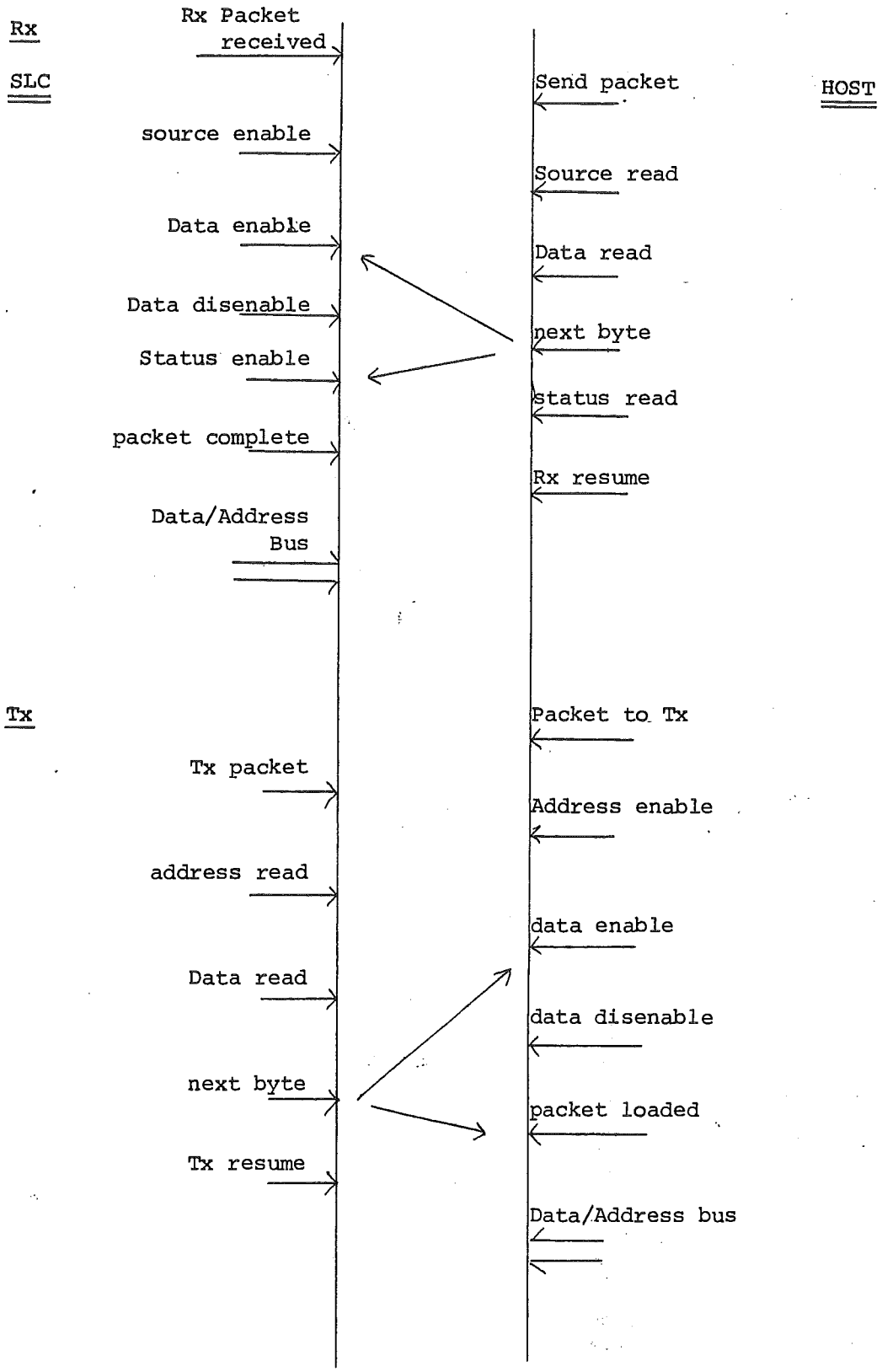


Fig. 5.2 SLC/Host Interface

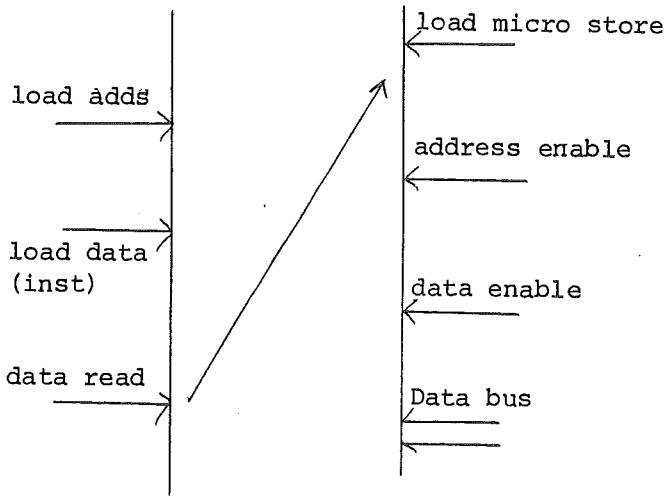


Fig.5.3 Bootstrap Interface

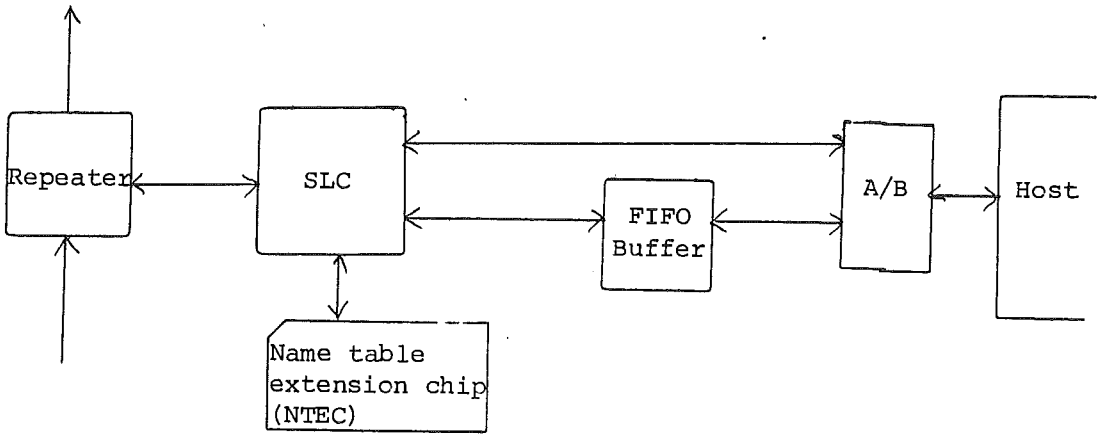


Fig. 5.4 System Structure with Extension Chips

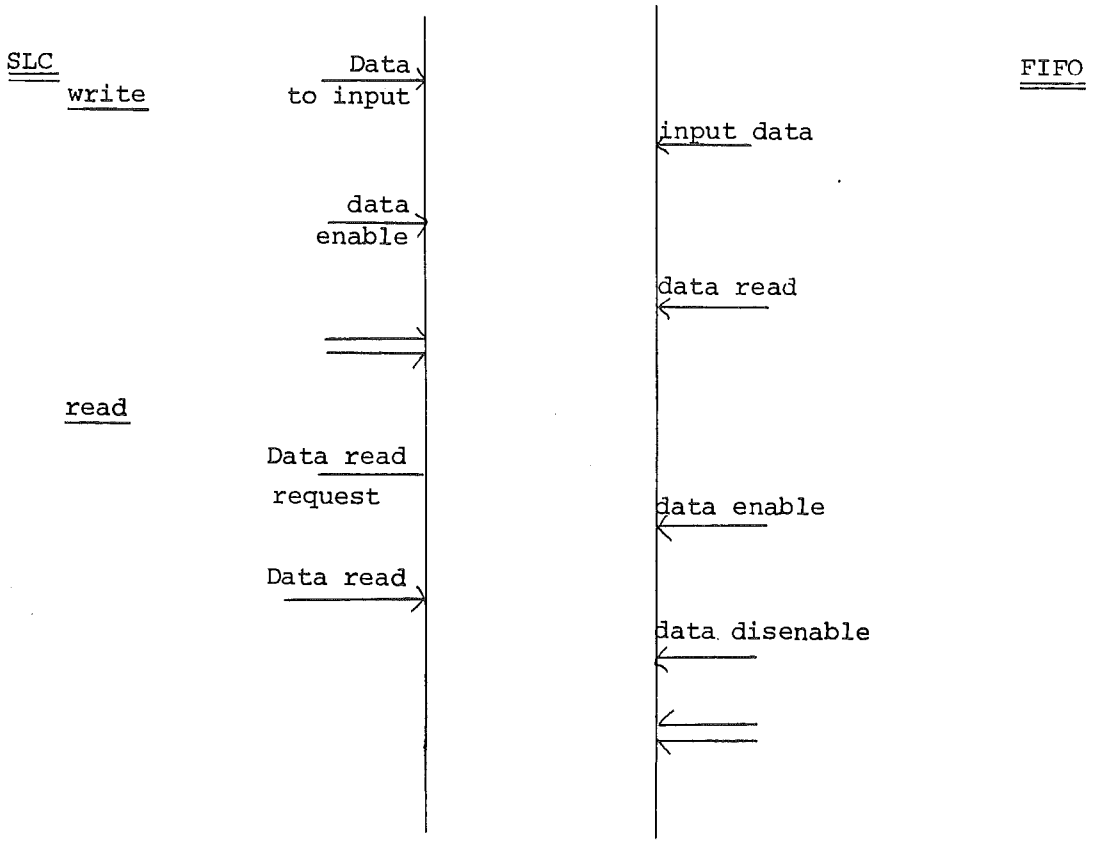


Fig.5.5 SLC/FIFO Interface

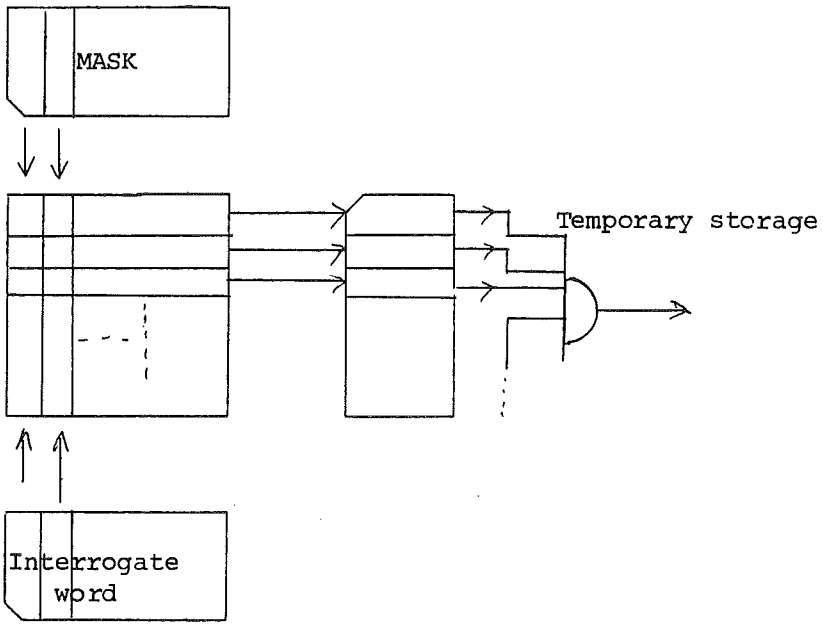


Fig.5.6 Name Extension Chip Architecture

SLC in Fig. 5.7. The interface between the host and the SLC can now be simplified: the address and status bits can be written into the FIFO in the same way as data.

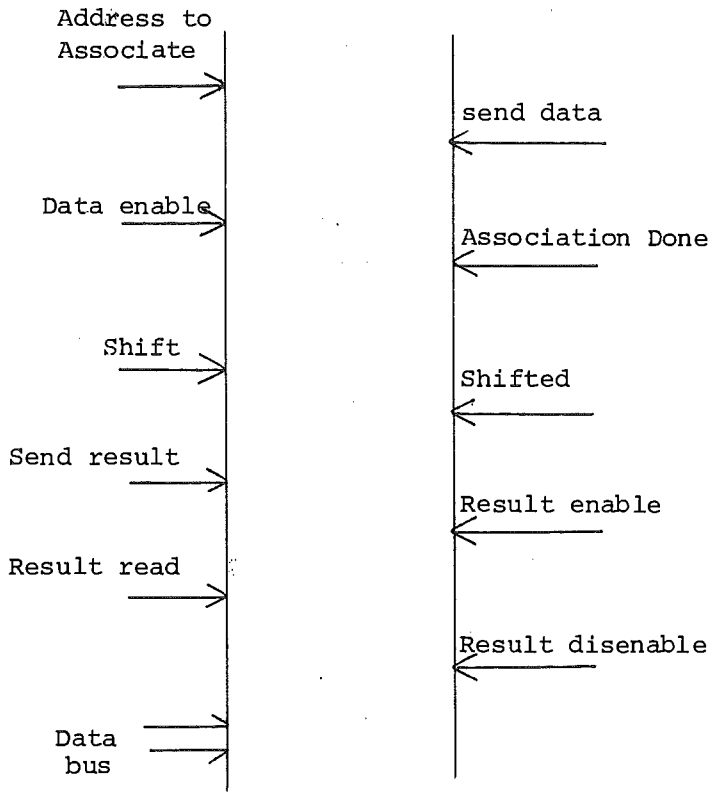


Fig. 5.7 SLC/NTEC Interface

5.4 Modulation and clocking

The modulation technique should allow the signals to travel a maximum distance between repeaters, not be dependent upon any particular clocking system, and have signals available to make clock regeneration easy. It should have maximum timing tolerance between digits and not

require complex logic to implement. In a simple system the clock can be transmitted separately from the raw data. This minimises the bandwidth required but introduces a DC component, and if it does not allow the use of transformers. Phase modulation can be used to give AC balance and allows the use of transformers for common mode rejection and isolation, but requires twice the frequency. Transformers can further be used to provide a separate path to a node and for distributing power to the essential logic of the repeater. This is necessary because no assumption can be made as to the availability of power from any node, and the network has to operate in a fail safe mode. Other modulation systems can be developed: for example, a four wire system can be used if the wires available are of the type used for duplex operation of teletypes. Such a system is similar to phase modulation with the signals along each pair being one bit time out of phase with each other.

An attractive way of achieving isolation between components in the system is by use of optical couplers. In this case, each repeater and SLC possesses its own individual earthed power supply; however, this is an expensive solution, especially at high speeds.

Clocking for a simple system can be provided by employing a central master clock. Such a clock is easily adjustable and can be made reliable. As it is passed from node to node it has to be reshaped. This may result in a system in which the data is not DC balanced and so does not permit the use of transformers. A better solution is to use a phase locked loop (PLL) at each stage. This assumes that there are pulses on the ring at all times and that a suitable signal can be decoded from the input for controlling the PLL. The PLL has a certain pull in range, and providing the incoming signal is within that range, the PLL will lock on. Also, it has a fraction of a bit of

'elasticity' once it has locked on, thus improving margins. As a number of PLL are connected end-to-end,the amount of jitter increases and when connected back on itself,the signal might stray out of range. This is unlikely to happen for a small ring. There is a trade off between oscillator pull-in width and jitter. Furthermore, the performance of a PLL can be improved by increasing the PLL bandwidth. However, this has the effect of increasing the effect of additive noise,and so an optimum value has to be chosen for the open-loop gain.

Another way of managing the clock is to provide each repeater with a separate monostable of uniform time constant. As the system is started up,a pulse train of the correct frequency is introduced, which has the effect of triggering the monostables one after the other at that frequency. When the pulses have travelled once round the ring, the pulse train input can be removed, and the clock pulses will continue to circulate. This system keeps the clock one shot period constant (and allows each adjustment of the mark-space ratio for use in simple PM systems), but will probably introduce more jitter than the PLL and may be unworkable for a large number of repeaters without an elastic buffer.

5.5 Networks implemented in the station logic chip (SLC)

In this section,the general requirements of a SLC are examined and a number of systems are chosen for implementation.

A general purpose architecture should allow the system designer to choose any particular packet format he requires. This means packets must have a variable number of fields each of variable length. The function of a particular field in a packet should be allowed to change, but these will fall into the following general categories:

1. Data - the data can take any format, and under some circumstances a description of the code it is written in is transmitted.
2. Address - there will be a number of address fields with some addresses being reserved for special purposes. A mask field can be incorporated to allow communication between sets of stations (from both transmitting and receiving point of view).
3. Control - the control fields of a packet have special meanings according to the system being implemented. They will be used to control the hardware retransmission mechanism and the acknowledgement mechanism. Under some circumstances they will define control packets (e.g. setup).
4. Error - this is optional and can be a CRC, parity, or similar code. If transmissions of whole blocks are taking place, then only a single error check may be performed per block.

Under all circumstances packets should be processed within their time slot and otherwise ignored as this will ensure their retransmission.

A major question to be answered when designing a ring is how the addressing mechanism will be structured. Of the two basic ways of doing this, position addressing stabilises delay and prevents hogging; but due to its inflexibility it will not be considered further. The savings in bandwidth due to shorter address fields can be partially achieved by a setup mechanism anyway. If addressing by code is used, then the transmitting device has to know the address of every entity in the system. However, if each destination is allowed to possess

several addresses ('names') and a hardware association is done for all of these at each node, then this frees the source from knowing where particular names reside. This requires extra hardware but does have the added advantage that special purpose addresses can be implemented easily(e.g. broadcast). Where a station possesses only one name ,the code addressing and name addressing schemes are equivalent.

A number of systems will now be considered ,all of which are suitable for implementation on a chip. These systems are chosen to be simple and reliable so that implementation is easy. Each system has the following characteristics:

1. It is decentralised
2. It is self regulating ,and its operation is well defined under heavy load conditions
3. It is able to handle both sophisticated devices such as computers and 'stupid' devices such as teletypes
4. It has common features with the others which make it suitable for implementation on a single chip
5. It has minimum logical delay through each node
6. It can be implemented independently of any transmission medium

The first system is the fixed length slot system ,which is similar to the Cambridge ring and offers a relatively fine grain of transmission. It is assumed the ring has an integral number of circulating slots ,each of which can hold a packet with the following format:

<--

(F) (dest) (srce) (data) (data..) (error) (control)

F - full/empty

If the ring does not have a logical length equal to an integral

number of slots ,then a shift register is inserted to make up the difference. If the length of the ring is not an integral number of packets ,then the gap digits have to be treated specially which makes the hardware unduly complex (especially if packet length is greater than ring length). It will also be assumed that each node knows the logical length of the ring. When a station wishes to transmit, it tests for an empty slot by overwriting the full/empty bit as in the Cambridge ring (section 2.2.3.1). When a slot has been filled ,the packet makes its way to the destination where the control bits are set and then returns to the source. The destination can set the control bits to signify accepted, busy ,or unselected. The source knows when its packet is coming back as it knows the length of the ring. This allows it to delete (make zero) the packet as it passes. By doing this and not using the same slot again immediately, hogging is avoided as a round-robin scheme operates.

If, under error conditions the station moves out of synchronisation , it will make its decision whether or not to transmit according to a bit following the full/empty bit. After a number of packet(s) it will find this bit zero and will attempt a transmission. As it is transmitting ,it will notice that it is not only overwriting zeros ,but also ones. This indicates that it is out of synchronisation ,and it will abort its transmission and attempt to resynchronise. (Such a process is similar to collision detect for CSMA network). A station will normally monitor its synchronisation state whether transmitting or not as an empty non zero packet indicates synchronisation error. To get back in synchronisation the station looks for a gap of zeros equal in length to the slot length, followed by no synchronisation error for one ring delay, at which point it can reset its packet counter. If after some period of time it does not find the zeros ,it calls a time

out, and outputs the zeros on the ring itself. Thus the system is stable since a station can detect when it is out of synchronisation and will not corrupt more than one packet. On turn on the stations tend to latch onto the first group of zeros. If these are not available, a time out will be generated, and the system will settle down in a stable state. Alternatively, the station can resynchronise by assuming the next one is the full/empty bit. If no synchronisation error is detected for one packet length after this, the assumption was correct and transmission can proceed. If the ring is empty for one ring delay, then the station can transmit and self synchronise.

If one of the bits in the destination address field of the packet is corrupted, then the packet may not be recognised by any destination but will be removed by the source. If the full/empty bit becomes full, then the packet will not be removed by a source and will circulate for ever. There are a number of techniques for removing lost packets in a distributed system, and these are discussed in Section 5.6.

A final requirement of the slot system is that it should provide a facility for setting up calls. This can be achieved by using the source select register and associating with it a counter which defines the number of packets the call is setup for. This requires a special control packet to be sent which initialises these two registers and which requires an extra bit in the control field. Another way of defining the setup protocol would be to allow for two control packets, one to indicate start of setup and the other for end of setup.

It can be seen that in the fixed length slot system all functions are decentralised and there is no central monitor station. Very simple devices can be handled by using the setup protocol with the setup counter being used as a time-out device. There are no gap digits and

therefore, no discontinuities in the performance characteristics (with synchronisation on the full/empty bit, this is true for any ring size).

The second system is the token system as described in Section 2.2.1. The token can be implemented in two ways. It can either be a unique bit pattern or a single 'one'. If no empty token arrives within a specified period of time, a time-out is called and new token is generated. The packet format is as follows:

<--

(T) (F) (dest) (srce) (length) (data..) (error) (control)

T - token

F - full/empty

When a token is recognised by a station, a procedure similar to that for the slot system is followed with the full/empty bit being overwritten and examined at the same time. The length of a packet is governed by the width of the data field, and thus file transfers take place easily.

The error conditions for the two token schemes are considered below. If the token becomes corrupted in either scheme in such a way that it no longer exists, a time-out is eventually called, and a new token is created. The time-out parameters are set to different values to prevent the same two stations timing out and generating tokens repeatedly. An error can occur in such a way that two tokens are present in the system (which is equivalent to a station being out of synchronisation). In the unique token scheme this can be detected by storing the CRC check of the transmitted packet and comparing it to the received one. As CRC checks are unlikely to be identical because only one source/destination transmission at a time is allowed) a fault in the received CRC check indicates there are two tokens or that the CRC check has been corrupted. Since there is no way of distinguishing between these two

cases,the assumption is made that the station is out of synchronisation and both packets and tokens on the ring are deleted. A time-out then takes palce, and a new token is created. As the time outs are set to different values the ring recovers. In the single bit token scheme a station can detect whether it is out of synchronisation by examining the bits it is overwriting for one ring delay after starting transmission. If the ring is cleared to zero between transmissions and the station overwrites ones, then it is out of synchronisation. This synchronisation procedure is carried out whether the station is transmitting or not. Once a station detects that it is out of synchronisation, it looks for a pattern of empty token, followed by the appropriate number of zeros (one ring delay), and the token again. The station resynchronises when this pattern has been detected and there is no further synchronisation error for some period of time. Because the bit pattern the station synchronises on is not unique, resynchronisation might occur in the middle of a packet, in which case the algorithm is repeated. Under some circumstances the station may be out of synchronisation but carry on transmitting since the packet it is overwriting contains the appropriate bit pattern at that point. As time goes on, the probability of the bit pattern occurring at the right place decreases, and eventually a station will detect that it is out of synchronisation. In this scheme the longer the ring delay, the higher the probability of detecting non-synchronisation on first transmission. In terms of hardware the unique token system requires a bit stuffer/destuffer and CRC storage. The single bit token system requires the ring to be cleared to zero between transmissions and a capability for recognising a specific bit pattern at the input. As these features are already provided for the slot system the single bit token scheme will be chosen.

If an error occurs in one of the other fields of the packet, the consequences are not serious unless it is the length field. In this case there is a possibility that the receiving station will be forced out of synchronisation,and a procedure as described above will be followed. The control field bits have the same meanings as in the slot system. With the token system the length of the ring will generally be less than the packet size, so that the register for counting the packet length at the source has to be duplicated. On the other hand,long packets can be transmitted easily, and hogging does not occur (except for very long packets) since the token distributes bandwidth in a round robin fashion. Under light loads the token system is slightly inferior to the slot system as the token has to travel the whole way round the ring before it can be reused. The station select, acknowledge,and setup protocols can be incorporated in much the same way as for the slot system.

The third system is the carrier sense multiple access system (CSMA) and the packet format is as follows:

<--

(SOP) (dest) (srce) (seq.no.) (length) (data..) (error)

SOP - start of packet

When a station wishes to transmit,it looks to see whether another transmission is taking place (by examining its counter to see if it is counting through another packet). If another transmission is taking place,it initialises its counter to a random value and repeats the process this random time later. If it finds it is not in the middle of a transmission,it begins to transmit its packet. If at any time during the transmission period the 'collision detected' signal is sent by the repeater,the transmission is aborted and attempted a random time later. Let us again consider the error conditions. As the system is

no longer a ring type system it does not matter if the address bits are corrupted as the packet will disappear anyway. If the length field is corrupted, then the packet will be incorrectly delivered, and the bits remaining in the system will produce some collisions, however, the chance of such an occurrance decreases as time goes on, and so the system recovers. CSMA poses few hardware problems in the SLC, but its throughput is lower than that of the ring, and there are no low level acknowledgements. The throughput can be improved by loading the random number counter with a number from a distribution computed from the load statistics. This is also required when the number of nodes is sufficiently large because CSMA system can become unstable. On turn on and in the rest state the repeaters generate zeros.

There is one other system which warrants consideration for inclusion in the SLC: and this is the register insertion system. Such a system can be made to operate in either variable or fixed packet length modes. When it is used in the variable length mode, the scheme for removing packets from the ring becomes complicated, especially under error conditions. For this reason this mode will not be considered. In the fixed length mode a packet is transmitted by placing it in a shift register and inserting it into the ring at the appropriate moment in time. The packet then moves round the ring until it reaches a destination. Once the data has been read, the packet could theoretically be removed by any station with an inserted register; however, to allow low level acknowledgements and to keep the packet removal strategy clean, the packet will be removed from the ring at the source. The packet format is as follows:

<--

(SOP) (dest) (srce) (data..) (error) (control)

Under error conditions the SOP bit can become corrupted and a

station might synchronise to another bit. This will corrupt the passing packet which will then be delivered incorrectly,or circulate round the ring without being recognised. Thus,two stations will have their registers inserted and will be blocked from transmitting. In order to restore the system,the two lost packets have to be cleared to zero (section 5.6). A blocked station will in due course time-out and start searching for a string of zeros of packet length. As zeros tend to cluster, such a string will in due course arrive, and the blocked register can be removed from the ring. A consequence of this design is that if fixed length packets are used by each station, but vary in length from station to station then the minimum size packet must be larger than the data length of the largest packet (otherwise, resynchronisation in the middle of another packet is possible). In the register insertion scheme,the acknowledge and setup mechanisms operate in the same way as in the slot system, and at turn on the latent digits in each node are forced to zero. If large packet lengths are required,then the shift register is placed outside the SLC.

In all the ring systems that have been described the control bits are situated at the end of the packet. This is so that the delay at each node can be as small as possible. The error check for these control bits can be in the last section of the error control field. As well as being used for acknowledgement purposes and for lost packet monitoring, the control bits can be used to improve the performance characteristics of the system. If a packet returns to the source marked busy and the host is notified of this fact immediately it might immediately attempt the transmission again which will tend to clog the system with busies. If the host is not notified of the busy until some time later then the busy traffic will tend to decrease. If the extra

delay is made dependent on ring loading, an optimum solution results. A suitable metric of ring loading is the time to acquire the next slot or token. This can be the delay before outputting the busy first time. If the transmission is attempted immediately again and another busy is received, then the extra delay can be increased to several slot acquisition times. This process could be made completely transparent to the source by automatically attempting the retransmission, bringing the ring and CSMA systems closer together. The disadvantage is that the station is blocked from transmitting any other packet while the destination is busy. The ring and CSMA systems can be further brought together by requiring that the CSMA network provides hardware acknowledgements in the same way as a ring.

5.6 Removal of lost packets

There are a number of techniques for removing lost packets in a distributed network. These techniques will be considered for both the slot and the register insertion systems.

A crude but effective way of monitoring lost packets in the slot system is by each node storing a table of the source addresses which pass the node. This table is of the same length as the number of slots in the system. The algorithm is based on the fact that a particular slot cannot be used by a source twice in a row but has to be passed on downstream. Thus, if the table is updated every time a slot passes the station and the oldest entry is deleted, and if the new entry already exists the packet is either being used illegally or is lost. In either case it is deleted next time it passes the station. This technique requires additional hardware which need not be excessive as the number of packets in the ring even for a large system is small.

Another method for removing lost packets in the slot system is by idle stations selecting the source address of the next full packet. If this slot returns marked empty or used by another source, the algorithm is repeated; otherwise, the packet is deleted at the next revolution.

The third way of removing lost packets is by assigning the address zero to the station where packet removal or deletion takes place. The logic for each station is the same, but the way it is executed is dependent on the station number. A special control bit is allocated in the packet and follows the address bits. When a packet is transmitted, this bit is set to one. The principle of operation is that the zero station is defined as the transition point. Every time a packet passes the transition point it is marked with a zero. If the packet arrives at the transition point already marked zero, then it is removed or deleted at the next revolution. This is achieved by overwriting the control bit by the OR of the bits of the station address AND the control bit (i.e. (OR(addr.bits))AND(control bit)). Thus, the control bit will change to zero at the transition point. Then if (OR(addr.bits)NOR(control bit)) is true, the packet should be removed. If the packet is addressed to the station at the transition point, then the control bit will be set to zero at the station following the transmitter, but as zero is a valid address and will be recognised anyway, this does not matter. The control bit is placed after the address bits so that the counter for deletion at the next revolution is not initialised in this case. This technique has the disadvantage that it is not decentralised as the station at the transition point cannot be removed. If this is done, another station has to be initialised to address zero.

The fourth method for removing lost packets was proposed by

Reames [RE75a] and depends upon arranging the stations on the ring in such a way that all with MSB=1 (most significant bit) are grouped on one side, and all with MSB=0 are grouped on the other. Two control bits are allocated in the packet for lost packet removal. Thus, two transition points are defined and if a packet passes through these thrice it is lost. Every time a packet passes a transition point, the appropriate bit of the control field is set; and when both are set and the packet passes a transition point, it is lost and will be removed or deleted at the next opportunity. If the packet is to be deleted then the control bits are reset to zero to ensure no other station also attempts to delete the lost packet. This method has the advantage that even if only one register is allocated in each station for removal of lost packets, then more than one packet can be in the process of being removed at any one time since this can be done by any station in the system. However, due to the additional two bit control field the delay through each node increases.

5.7 SLC Architecture

Having proposed a number of systems it is now possible to examine them more closely with a view of implementation on a single chip. A decision which effects the design of the transmit and receive shift registers is whether the station operates in a duplex or half duplex mode. The CSMA system can only operate in the half duplex mode. The register insertion system can operate in either mode, the duplex system being preferable as a station is not blocked from receiving data from another node while waiting for an acknowledgement. Similarly the slot system can operate in either mode, duplex being preferable since it allows a station to receive while waiting for an empty slot. The token

system can operate in the half duplex mode if the length of the ring is greater than maximum packet length. Because most of the time this is not the case, the token system will only operate if the station can transmit and receive data at the same time. Thus, to implement all systems the SLC will have to provide both transmit and receive registers. The length of these registers is determined by packet size. For the token and CSMA systems, packets can be large and full size shift registers are likely to take up too much space on the chip.

Each system requires a packet size counter, and as this takes up little chip space it will be long enough for the largest packets. If large data transfers are taking place, this will enable the system to operate efficiently, even if a long header is used for the transmission protocol (e.g. TCP). Another counter is required in each system to count through different fields of a packet, although this information could be derived directly from the packet size counter.

For the slot and token systems a further counter will be required for counting ring size, and in the token system another counter will count through the packet as it returns to the source when ring length is less than packet size. All systems require a time-out counter. In the CSMA case this time out counter is used to notify the host when rescheduling of a transmission has failed a number of times.

For each system the addressing will be done associatively. A number of address registers will be provided directly on the chip (for self and special addresses), with provision for extension to another chip.

The retransmit/busy logic can be based on one further register. This register is loaded with a random number in the contention case, or with a number which defines the number of ring delays the system waits

before passing the busy to the source in the other schemes.

The setup mechanism is provided using the setup address register and a setup counter. If chip space permits, this can be extended by using another register as a mask.

All systems provide an error check facility. This consists of the error check logic and two shift registers for CRC generation and checking. The logic can be programmed to provide one of a number of error checks.

Finally, four more logic areas are required to complete the SLC hardware. The first is used to define the packet field lengths and the microprogram, the second for writing control signals to interfaces, the third for lost packet detection, and the fourth for directly writing a one to the transmission medium.

It can thus be seen that in terms of hardware the proposed systems are very similar and implementing them on one chip poses few problems. However, it is unlikely that without an extension chip, the variable packet length networks will operate in an autonomous fashion from the host. If such an extension is not available, then the host will be double buffered in the SLC and will supply data at a rate defined by the width of the double buffer. The size of the buffer will be governed by chip space and pin limitations.

The individual component of the SLC having been described, its architecture will be considered in more detail (Fig. 5.8). The microprogramming approach is adopted because the system is of sufficient complexity to make it economically feasible. If only one communication structure were to be implemented then a discrete system would be cheaper. The microprogramming technique has the disadvantage of being slower, however, as the system operates under well defined

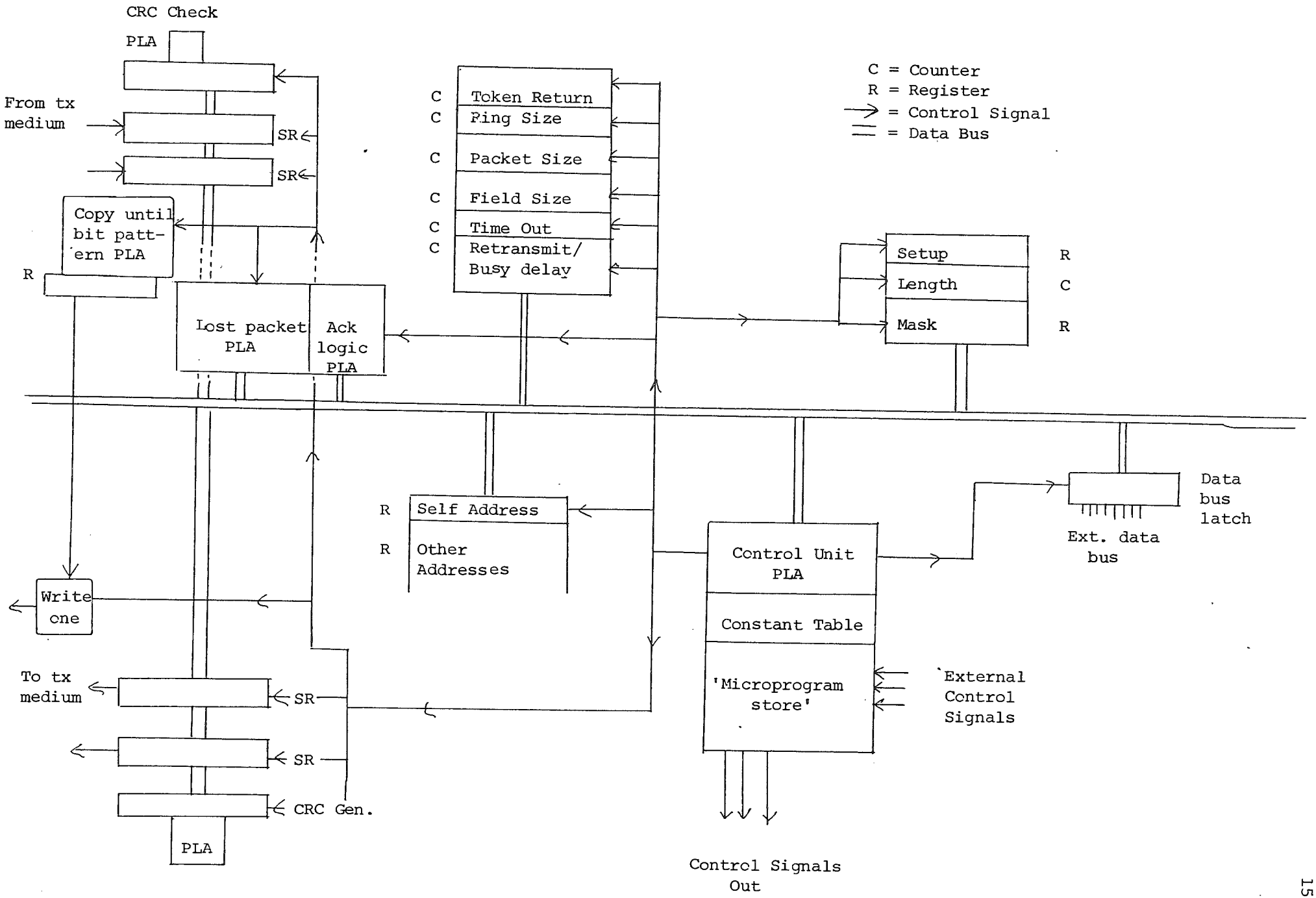


Fig. 5.8 SLC Architecture

conditions,overlapping can be used to increase speed.

The SLC is based around a bi-directional data bus. All system components communicate with each other via this bus which is tri-state to allow the or'ing of other devices for DMA to extension chips. The operation of the system components is coordinated by the timing/control unit. Logic functions are implemented using ROM's and PLA's. The size of the data bus and the registers/counters will be governed by chip space and pin considerations.

The size of the control store is defined by the number of control states in the system. In order to minimise this,the microprogram operations are defined at a miniword level rather than solely at the basic gate level. This means autonomous hardware can be used to perform logic controlled subfunctions. By taking control information out of microinstructions and placing it in discrete logic (and using PLA'a), microinstruction size and control store size are reduced at the expense of more complex hardware functions. If the length of time to execute an instruction is known, the control unit delays by counting;otherwise, an explicit return control path is provided.

Having defined the structure of the SLC it is now possible to look at the operations which will be executed by the various micro-instructions and hardware units. These fall into two categories, I/O and control. The I/O operations are:

- read N bits from medium
- write N bits to medium

which are used for communication between shift registers and the transmission medium and

- output register/counter
- input register/counter

which are used for communication with the registers/counters along the data bus. The read/write operations can be executed simultaneously to allow duplex operation of the SLC. Furthermore, once the read or write operations have been initialised, the input/output instruction can be commenced. If extension chips are being used, the control unit generates additional control information before the input/output instructions are executed.

The control operations include commands which invoke the various special purpose hardware units:

- invoke error check logic (involves 'write one' or 'write byte')
- invoke lost packet logic (involves 'write one')
- invoke acknowledge logic (involves 'write one')

and

- invoke setup logic

An operation is provided to utilise the hardware input unit which detects a specified bit pattern at the input from the medium

- wait for bit pattern

This bit pattern can be stored in a register (in which case it can be used for address recognition), or if it is invariant, it can be hard wired. This command is used to synchronise the SLC to the start of a packet. Operations are provided for delaying by a number of clock periods

- wait until counter N

and for asynchronous counting

- increment counter

Finally, an operation is provided for invoking the address recognition logic

- compare registers (addresses)

As an address extension chip is an option this will be done on a byte by byte basis.

This completes the list of operations. By using them in suitable combinations,any of the previously described network architectures can be implemented. The most critical part of the design is to ensure that enough time is allowed to execute the program at each bit time. As packet field lengths will generally be several bits long,this should be possible; nevertheless,the timing has to be considered carefully. Under all circumstances the fastest decision is made on the full/empty bit,and this is achieved by using the overwriting algorithm described previously.

5.8 Issues of complexity

It is important to consider if the chip is too complex from a microprogramming point of view and thus operates at an unacceptably low speed. It is attractive to design a very flexible microprogram instruction set which leaves the designer a free hand with the system. This unfortunately leads to minimum field lengths which are too high at maximum line speeds. If the transmission speed is reduced to less than the maximum logic speed,then several microprogram instructions can be executed in each transmission time, and programming can be more general. However, the systems being implemented on the SLC have to be capable of handling high speed input traffic,and they do this in the most predictable way when line utilisation is low. It is thus desirable that the SLC is capable of operating at maximum speed, and therefore that the microinstruction format is wide. Further requirements of the microinstruction structure are that it should allow efficient program generation, system development,and debugging.

Some additional features can be provided in the system to help debugging and for special modes of operation. The most important of these is the broadcast facility, which is implemented by reserving a special address for this purpose (e.g. all zeros). This can be done by making it another (perhaps hard) entry in the name table. A facility which helps to hardware debug the system is the echo facility. Another address is reserved for this purpose. When a station receives a packet of this type it transmits an echo packet to the original source. For the register insertion system, this can be done directly by reversing the address fields and retransmitting the packet. The station select register used for setting up calls can also define sets of selected stations by using an associated mask. If the station select register is set to a value which does not correspond to any node, the station will be disconnected from the network. The only way it can then be reconnected is by means of a broadcast packet or by its host.

Because the chip possesses a bus architecture, there is no reason why in a sophisticated system the control store cannot be loaded by special packets from the network. This would lead to interesting possibilities since the structure of the network could be changed dynamically to suit load conditions.

The SLC should also be examined to see if it is too complex for present technology. LSI demands regularity. As the chip consists of areas of regular logic, it is to be expected that these groups can be packed densly. However, there are a large number of data and control lines which take up additional space but should improve yield as faults under transmission lines do not render the chip inoperative. A gate count is shown in Table 5.9 where the gate-to-component ratio is the same as for uncommitted logic arrays (ULAs). The total is within current technology for a number of logic families.

Type	Number	Size (bits)	Gates/ Bit	Total (gates)
I/O Shift Registers	4	16	10	640
Control Registers	5	8	10	400
Status Registers	2	8	10	160
I/O Latch	1	8	4	32
CRC Shift Registers	2	16	10	320
Counters	7	8	10	560
Match Register	1	16	10	160
Self Address Register	1	8	10	80
Setup/Mask Registers	1	8	10	80
I/O Pattern Comparator	1	16	3	48
Control Unit PLA	1			≡ 500
Acknowledge Logic PLA	1			≡ 50
Error Check PLA	2			≡ 100
				3130 gates

Table 5.9 Gate Count for SLC

Table 5.10 shows the pin count with minimum multiplexing of control signals. There is a strong economic advantage in using a standard size package the largest of which is 40 pins. It would thus be advantageous to multiplex some of the control signals (for example, the microprogram bootstrap controls).

Interface	No. of pins	
Repeater/SLC	5	
SLC/Access box (operational)	9 + 12	(Rx, Tx multiplex)
SLC/Access box (bootstrap)	6	
SLC/FIFO	4	(Rx, Tx multiplex)
SLC/NTEC	10	
	46 pins	

Table 5.10 I/O Pin Count for SLC

The choice of logic technology is governed by power dissipation, packing density, and speed. With maximum package power dissipation of 500 mWatts and a 3000 gate chip, only I^2L can be considered from the bipolar technologies (although this gate count is very conservative). MOS has high packing densities and low power dissipation, but due to the interconnection of many logic units with a single bus, the time constant is likely to be high. Also, MOS is slower than bipolar, although devices with 4000 gates, in part working at 7MHz, have been produced.

In order to estimate the size of the chip, the capacitance of the control data bus has to be calculated. For NMOS this is 20 pfs., and to achieve a satisfactory clock rise time for a system operating at 7MHz, this requires a 10 mAmp driving current. The transistors directly driving the bus have to be large enough to provide this current (aspect ratio 120), but they in turn have a self capacitance which affects the next layer and so on. It takes six to seven such layers before the transistors are of minimum size. A new gate count is made for NMOS (as regular devices such as registers and counters can be implemented more

efficiently than with ULAs), which comes to approximately 1300 gates. It is thus found that for a NMOS-SLC operating at 7MHz, the chip size is 225 $mils^2$. This is a large chip, and thus the yield is poor (8%) and costs per chip high (£40/chip for low volume). However, the SLC can be partitioned into two chips (within one pack), thus improving yield (15%) and decreasing cost (£15). The power dissipation assuming half the gates are on, and 5V power lines are being used is 100 mWatts (dynamic) and 200 mWatts (static).

For SFL (a variant of I^2L) the chip is smaller since the logic more compact and is found to be 150 $mils^2$. As 250 $mils^2$ NMOS chips are currently being made, it is expected that the SLC can be manufactured. If chip space is a problem, then some sections of the SLC can be implemented on separate chips, but this has the disadvantage that the machine will be slowed down due to added timing problems.

5.9 Applications

The SLC implements a number of communication systems which can effectively handle various traffic patterns. The applications for such a network range from real time, where the inquiry traffic is short and the output long, to multiprocessor systems, where traffic can be very high and variable. An algorithm has to be developed to match the speed of transmission and reception. The simplest way of doing this is by associating a speed number with every device and updating this when the delay does not meet the estimate. A correct choice of time-out parameters has to be made. If these are too long, then the system does not recover from error conditions quickly; if they are too short, errors will be detected when they have not taken place. A different interface is required for each device attached to the network.

When a network of this type has been set up, the primary traffic will probably consist of file transfer traffic and teletype traffic. However, there is no reason why such a network could not be used to implement a distributed computer system or a distributed database system. This poses some interesting questions of naming and addressing, which are to some extent alleviated by the broadcast facility of the networks. The SLC can also be used to implement a ring contention network. Such a network consists of a ring used in a broadcast mode and has the advantage that at low loads the delays are comparable to a broadcast system, but at high loads clashes are very infrequent.

Another question to be considered is that of reliability. Unfortunately, all SLC networks are susceptable to a single transmission line failure, because even in the contention case the unterminated reflections from the break will render the system unusable. This can be overcome by laying the cables in such a way that the network can be easily reconfigured or by duplicating the system. The latter is as interesting a possibility as provided the clocks can be handled appropriately, there is no reason why two SLC networks should not be placed side by side to double the theoretical bandwidth.

CHAPTER 6

PROTOCOL ISSUES

6.1 Introduction

In this chapter some of the protocol requirements of local networks are discussed. This is followed by a redesign of the Cambridge ring to provide hardware support for protocol implementation. The new ring is similar to the old one, and requires few hardware changes. Simple protocols for resource sharing, peripheral sharing, and file transfer are presented which allow both intelligent and dumb devices to be independently connected to the ring.

6.2 Local Network Protocols

Protocols should be designed in such a way that extensive revisions are not necessary as the network grows and its uses become more general. However, at the outset only simple protocols can be precisely specified and easily implemented. Initially,the protocols provide a mechanism for resource sharing, for file transport, and for connection of interactive I/O devices. Later,they become more oriented to the more complex problems of distributed computing systems, distributed database systems etc. Furthermore, as local networks develop, a need arises to connect such networks to each other and to more complex global systems. This means that protocols have to be incremental in nature and be capable of being extended to allow communication with global networks.

Local and global networks are interconnected using 'gateways'. If local networks incorporate a standard way of implementing a gateway such extensions are not difficult. Local and global networks are

brought together by common network level protocols. This allows local networks to be expanded by introducing a backbone network, the protocol structure remaining unaltered. Some of the proposed standards, such as the X-25 interface [FO75], do not allow this, and it is unlikely that this is the correct direction to be taken in the future. Unfortunately, such layering of protocols causes inefficiencies, but these tend to be optimised for the local case. This can be done if the interfaces between protocol layers are left clean and invariant of the algorithm employed at each level. As the local network is expanded, the additional protocol functions are inserted by a special unit which extends the subset protocol (e.g. routing, addressing). This unit can be an additional computer or program in the host. Such an approach requires a pure, free formatted address scheme : that is address bits belonging to one level are not used by another.

As local networks become more complex, other issues have to be faced. These include the dynamic naming of services, the use of the broadcast facilities, and the problems of distributed file systems. For example, in a distributed computer system the personality of the distant computer is hidden from the user who only sees a global computing resource. This requires additional features to be provided to enable efficient process-to-process communication to take place [STR78]. It is inefficient for each node to possess a catalogue process which indicates where network resources lie and how they are to be used. If another protocol level is used, then such tasks can be rationalised and distributed throughout the network. Unfortunately, this leads to inefficiencies as process-to-process addressing requires long address fields, and the naming structures in such a distributed system becomes complex. Some protocols (e.g. SDLC) do not provide sufficiently

flexible addressing and control structures and thus have been extended for some applications. An interesting scheme is utilised in the Irvine ring, where network addressing is by name, with names corresponding to processes which might be resident in any of the machines and which can move from one machine to another. An alternative way of process-to-process addressing when the location of the destination process is not known is by use of the broadcast facility.

Flow control of packets may be handled at the protocol level. This should not place a limit on the maximum speed of transmission and has been implemented by use of windowing (e.g. TCP). In this scheme the receiver informs the source how many packets can be transmitted past the previously acknowledged packet. However, if the destination does not provide an explicit acknowledgement, then this is not possible and speed matching algorithms have to be employed. Furthermore, if the destination cannot accept a message, then the message can be queued, or it can be ignored. In the latter case no guarantee is given that a destination will respond to a transmission within a specified period of time, which makes network behaviour less predictable. This can be remedied by queuing which can be done at the source, in the network, or at the destination. There are other schemes, for example, dividing transmission requests into two groups as used in the protocol for the Cambridge ring described below.

6.3 A redesign of the Cambridge Ring

The main feature of the redesign is that control packets are introduced and are used both to arbitrate between sources competing for a single destination and by the user, who can set an extra control field in the packet. The wasted bandwidth is reduced as despite the

extra control packets fewer transmissions are returned marked busy. The other major alteration allows the user to transmit to himself either via the ring or directly via the station unit. This allows faults to be detected more easily: as when persistent errors occur, the user can distinguish between faults that occur in his own station unit and in the rest of the ring. The control signal for the direction that self transmissions take place can either be set by switches at the station or by software in the attached computer. This means that this control signal is provided as another line on the interface.

The new packet structure is shown in Fig. 6.1. It can be seen that packet length is increased by one bit although it could be kept at its previous value by removing the SOP bit. The hardware can now function in a number of modes. When the source and destination addresses correspond to station addresses, the packet is one of two addressed packet types (control and data). When the destination address is set to all ones, the packet is one of two broadcast packet types (control or data). These broadcast and addressed packets are under the control of the user and are acknowledged as before. The control bit is placed at the front of the packet to minimise the timing constraints on the logic.

In order to arbitrate between sources competing for a single destination another packet type, the token packet, is introduced. It is distinguished from other packets by the destination address, which is set to zero. The algorithm for ordering transmission requests is completely decentralised and is based on dividing all such requests to each destination into two pools. While one pool is being served, all other transmission requests are placed in the other pool, which is only serviced once the first pool is empty. No queue is formed within

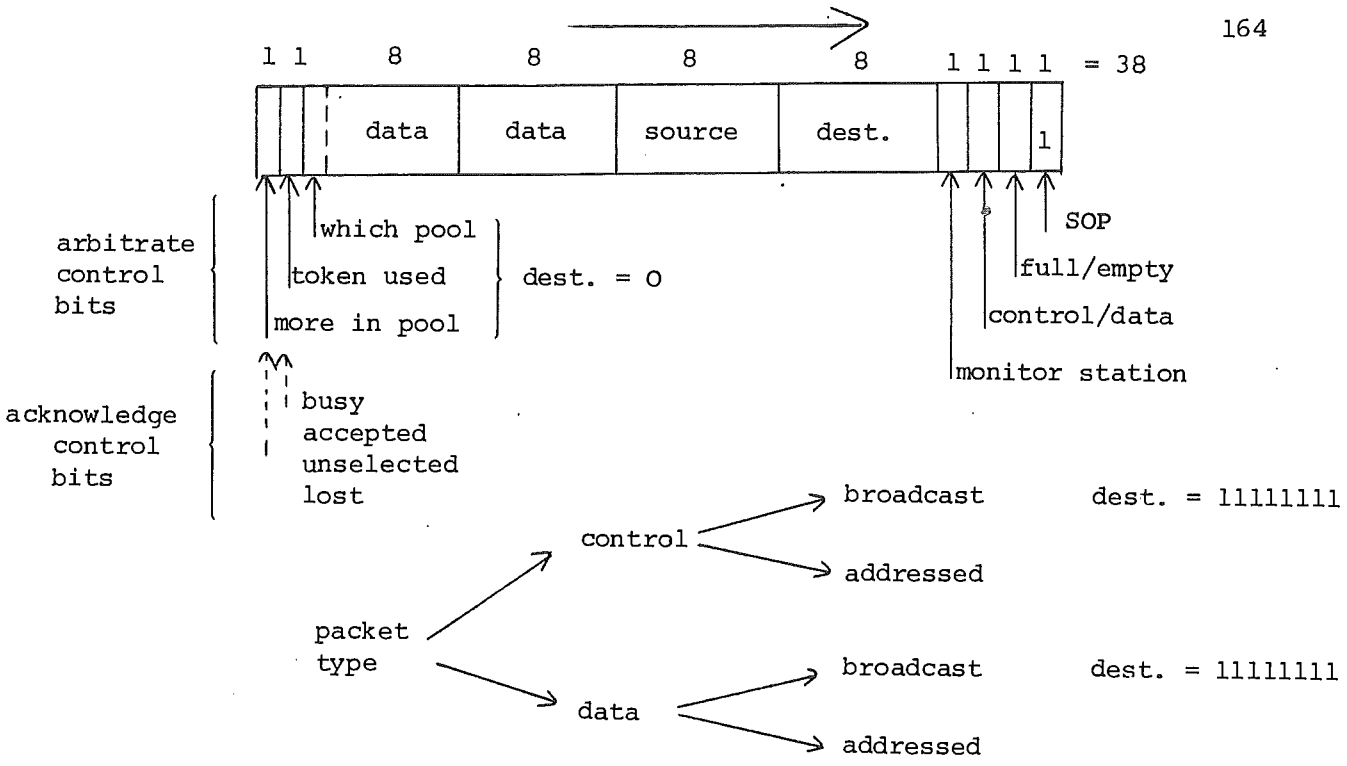


Fig. 6.1 Packet Structure

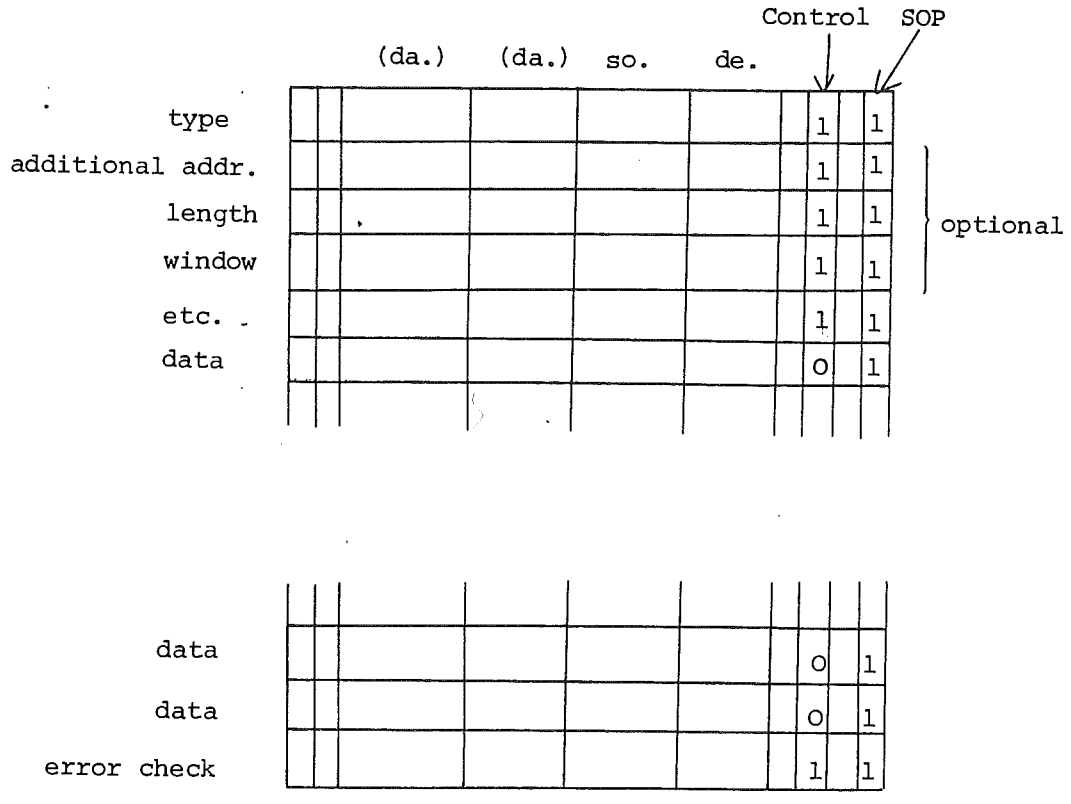


Fig. 6.2 Message Structure

a pool, the emptying process being according to ring order. Three control bits are used which can occupy the last three fields of the packet since neither the data nor the response control bits are in use when the algorithm is in operation.

The algorithm can operate at the packet or the message level. Messages are implemented either by transmitting the length of the message in the first packet or by using the control/data bit. This bit is set to one at the initial transmission and then zero for all other packets in the message until the last packet, when it is set to one again. Thus, if the initial transmission does not indicate message length (control bit zero), the algorithm operates at the packet level; otherwise it operates at the message level.

The algorithm for the redesigned ring functions at the message level, the control/data bit being used to define message size. If the initial transmission is successful, the station select register of the receiver is filled with the source address and other transmission requests are returned marked unselected. It is these transmission requests that are to be arbitrated, and so their 'which pool' bit is set to indicate to the transmitter the pool in which it has been placed. At the same time, a bit is set in the receiver to indicate that a transmission request has been received. There are two such bits since there are two pools, and it is the other pool from the one currently being serviced that is set. Each receiver possesses a further bit to indicate which pool it is currently emptying. When the last packet of a message has been received (indicated by the control/data bit) and there are no transmission requests in either pool, the receiver goes into the wait state. Otherwise, it indicates to the first transmitter (in ring order) from the new pool to transmit its message. This is done

by the receiver broadcasting a token packet. The first control bit of this packet indicates which pool is being served. The second control bit is the token (initially zero), and the third control bit indicates if there are one or more members in the pool. The token packet makes its way round the ring and is received by all transmitters waiting for a token from the destination. If the token is for the same pool as the waiting transmitter, the token bit is overwritten with a one. At the same time, this bit is read, and if it is found that it was already used, a one is written into the final control field of the packet to indicate that the pool is not empty. When the token packet returns to its source, the control fields are examined, and if the pool is empty (only under error conditions), the other pool is served. Otherwise the station waits for the first packet from the source which has claimed the token. This transmission has to take place rapidly, and when it does, the select register is filled as before. If the token has been claimed by the last member of a pool, the other pool is served next time, and the appropriate pool bit is reset in the receiver.

There are a number of problems associated with this scheme. If a source breaks down while waiting in a pool, then the receiver is not affected. If, however, this breakdown occurs after the token has been claimed, then the receiver will be hung up. This will also happen if the source breaks down before the end of a message and both are remedied by a time out in the receiver. Also, the destination checks the selection register at each packet so as not to mark unselected packets busy. This means that each packet has to be processed within its own slot time. For some traffic speeds the pool system is worse as three transmissions are required to a busy node, whereas normally the minimum is two.

A number of variations are possible in the implementation of the new ring. The control bits can be placed serially at the end of the packet instead of being encoded. This would make them directly readable by using the packet size counter but would also make packets longer and increase delay. The hardware can be extended to transmit to the first available receiver, but this requires extra storage for the waiting destination addresses. The control/data bit can be placed at the end of the packet and used for the arbitration algorithm. This would mean the data field is not used for control, but the acknowledge logic would have to be very fast. The algorithm can be simplified by not checking for the presence of more members in the pool, but one token transmission is then wasted when switching pools. It is noticeable from studying the slot system how variable the performance is due to the gap digits. If the SOP bit is ommitted and the station synchronises on the full/empty bit, the effect of gap digits disappears.

Let us examine the hardware requirements of the new system. On the transmission side each station requires one bit of memory, token and control packet detect mechanisms, and perhaps a destination address store if other transmissions are attempted while waiting for a token. On the receive side three bits of storage are required, along with a length register if the control/data bit is not used to delimit messages. Also, a redesign is necessary to extend the packet size by one bit, to add two signals to the access box interface, and to provide the extra logic for self transmission. Other hardware components required for the new ring are already present, and so the design does not present a major hardware effort.

The new ring retains the advantages of a small packet system, and the availability of a control field gives more flexibility to the

user. No error control hardware has been introduced ,because it has been found that the error rates in the present ring are very low. However, if the counting scheme is used to delimit messages, the control/data bit could be used as a parity bit or to indicate the satisfactory reception of the previous packet. The number of busies has been reduced, although these can still occur once a message is being transmitted.

There are a number of other schemes which could have been implemented to improve the ring performance. The members of a pool can be queued and served in order of arrival as in the DEMOS system [DOW77]. On the other hand, the receiver could mark all unwanted packets unselected and broadcast back when ready to receive again. This reduces the number of busies but does not enforce any service order. The best algorithm in any situation depends on the degree of variability that can be tolerated by the driving software. As the ring service time is likely to fluctuate anyway due to shifts in network load, it is unlikely that a queuing system gives much better characteristics than the algorithms described above. It would be interesting to consider a two tier scheme,where separate pools are used at the packet and at the message levels as the number of packets marked busy would then be reduced to zero (although unselected acknowledgements would continue).

6.4 Protocols for the Cambridge Ring

The protocols described below are designed to be implemented on the redesigned ring. Use is made of the extra control field for delimiting messages ,and thus the scheme is not applicable to the ring as it exists at present. However, an alternative strategy where the start and end of a message are denoted by a unique bit pattern makes

the two ring schemes compatible. This approach is also advisable when error rates suggest that the control bit is likely to be corrupted frequently. Different message types are used to implement the file transfer and dumb user protocols. Multiplexing to a number of dumb users is allowed, although intelligent devices can only receive one message at a time. There is no provision for priority messages, nor for any flow control (e.g. explicit per packet acknowledgement) apart from that already present in the hardware. It is thus assumed that if buffer space is exhausted, the transmission will be completely repeated. However, as the control field of a message can incorporate a window, the protocols can be extended to incorporate flow control.

Initially, a message system (Fig. 6.2) is implemented which allows the meaning of control fields to be easily changed or expanded. The message system allows data to be transmitted efficiently by synchronising the transmitter and receiver and provides error control.

Protocols at the message level are shown in Fig. 6.3. When a message transmission is requested, the control/data bit is set to one. The token mechanism may operate, the source eventually being selected at the destination. A number (defined by convention) of packets containing control information are now transmitted with the control bit set to one. This is followed by first the data with the control bit zero and eventually by a packet containing a 16 bit error check which unselects the destination. It is expected that to avoid prolonged lock-out and to prevent interactive teletype traffic from being unduly delayed, message size will not exceed 4Kbits, and thus a 16 bit error check is adequate. If the error check is incorrect the source is notified by a separate message, and the transmission is repeated. The message scheme does not allow for single packet messages. Such messages can be

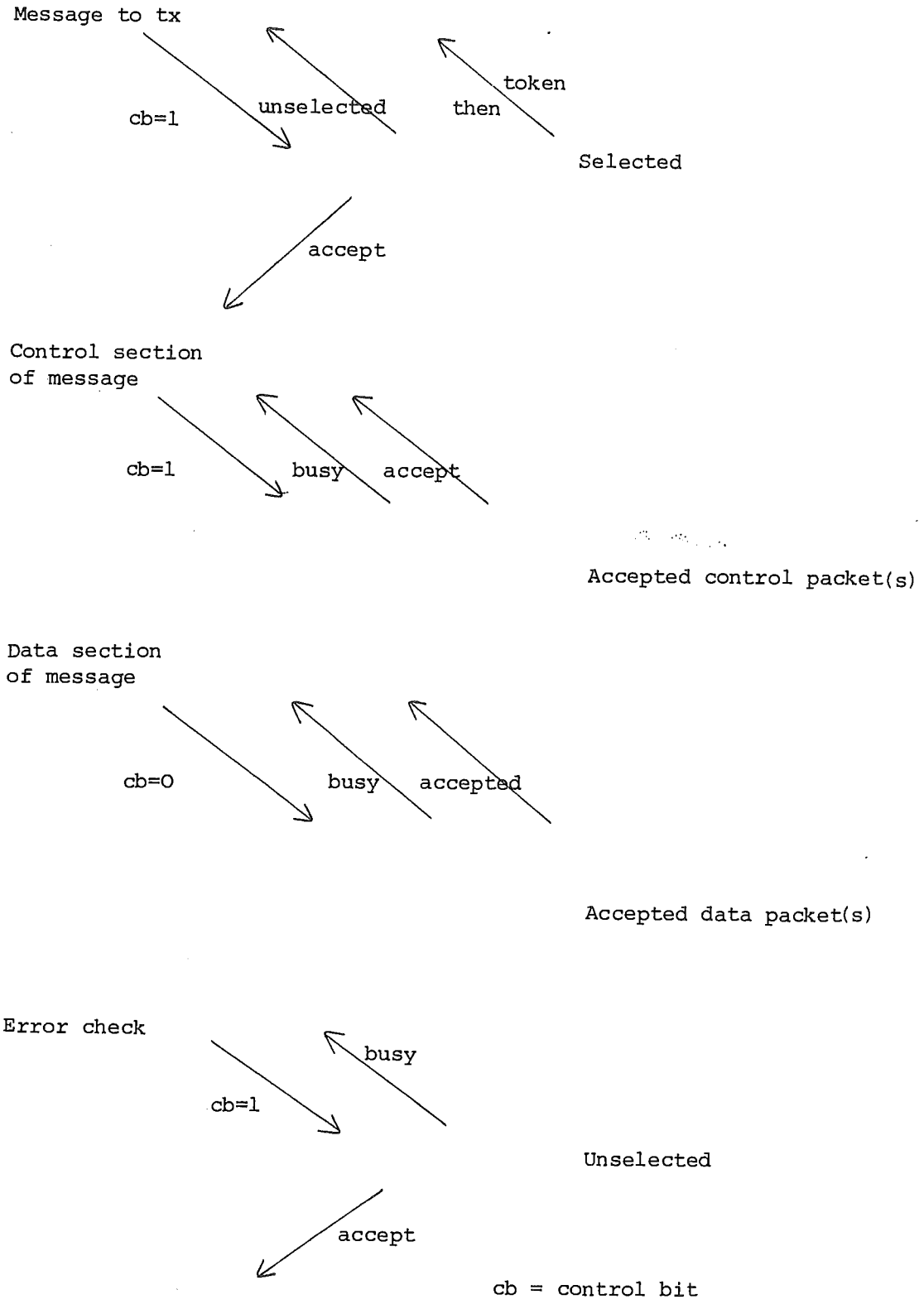


Fig. 6.3 Message/hardware level protocols

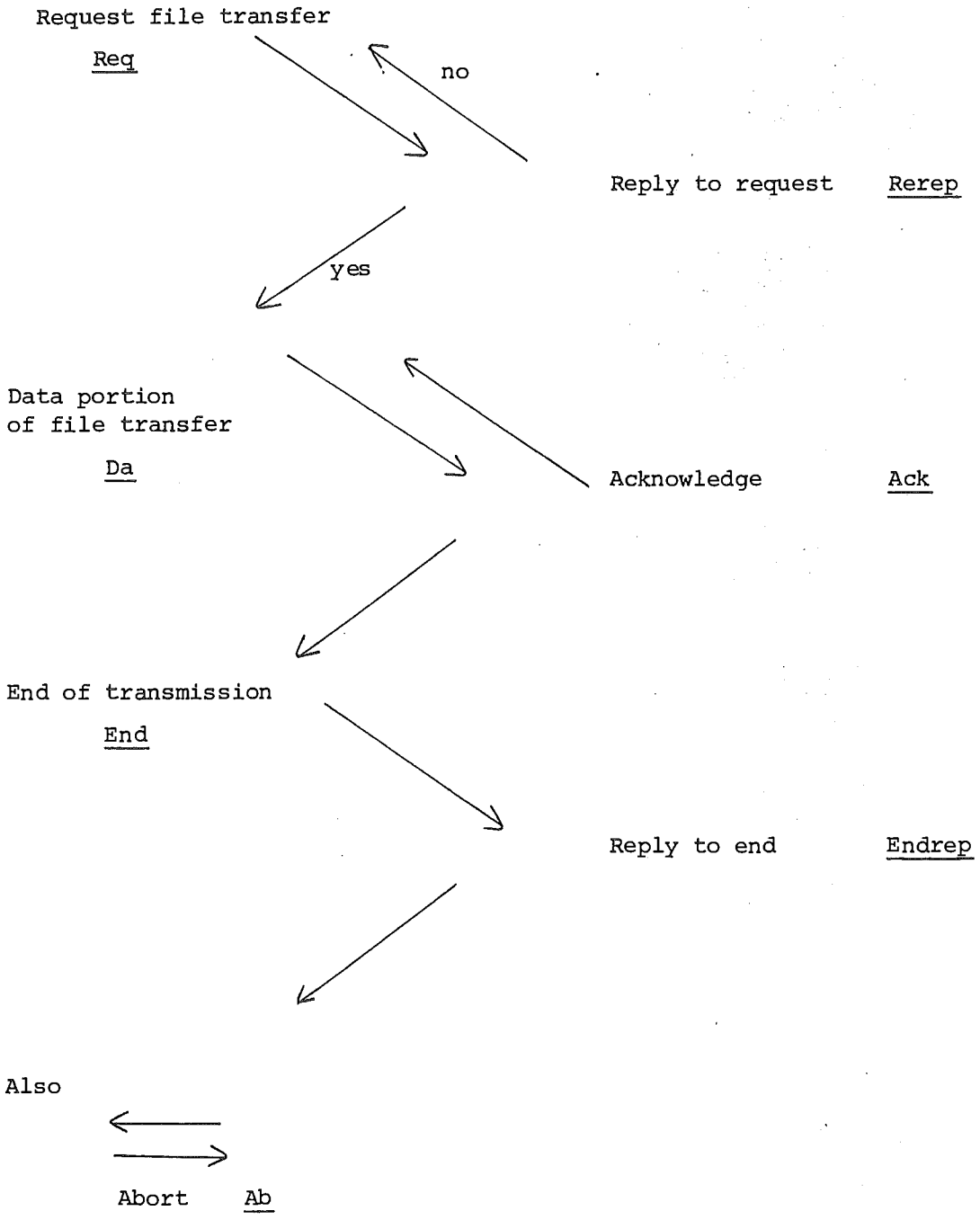
utilised to transmit control information when it can be encoded within one packet. However, not all control messages are short, and as they take only a small portion of the bandwidth, this limitation is not severe.

The message scheme is used for communication with dumb devices. The select register is loaded by the first accepted packet, which is followed by commands for driving the device. This select register is normally released by a control packet, but may be timed out if frequency of transmission is below a predetermined threshold. If this threshold is set to a high value, driving dumb devices poses few software problems as time outs are less critical, although hogging may still occur.

Implemented on top of the message system is the file transfer protocol which is similar to that in Ethernet, the different message types being shown in Fig. 6.4. A file transfer may be aborted by either partner transmitting an abort message. The file transfer protocol can be extended by the use of sequence members and windowing.

Each intelligent host on the ring possess a process which is used for communicating with the ring. This process is likely to govern the speed at which the host transmits and thus should be implemented at a low level, (e.g. in the microprogram). When very high speeds are required, the interface can be designed in channel mode, although for high ring loads the delay round the ring might not warrant this. However, it is suitable where the interface is designed to allow multiplexing at the packet level and where there is a large volume of traffic into the node.

It is interesting to consider the possibility of introducing free-standing service devices to the ring, such as printers, and interactive I/O devices like teletypes. In the former case the service



- if no reply within time out repeat packet.

Fig. 6.4 File transfer level protocols

device will possess some intelligence to allow it to be driven at higher rates than for dumb devices. This entails the introduction of additional protocols and storage and leads to the problem of where output for the device is to be buffered. The problem of free standing interactive devices can be approached by incorporating a microprocessor at each unit. The microprocessor can be used in two modes: character-by-character or line_by_line.Since interactive devices tend to operate in a duplex mode,it is inefficient for the host CPU to generate a separate acknolwedgement for each character. This can be remedied either by using the microprocessor to transmit line-by-line or by the use of a single free-standing concentrator. The concentrator operates in a character-by-character mode with the microprocessor, but in a line-by-line mode with the host CPUs.

The microprocessor is used to setup the connection with the concentrator and also to handle errors, retransmissions, and type ahead. The concentrator acknowledges individual characters, but the line acknowledgement is issued by the host CPU. In order to allow the concentrator to communicate with a number of microprocessors, the protocol incorporates a new select scheme where the station select register is cleared immediately after a packet has been received.

The hardware of the Cambridge ring has thus been redesigned to enable a message system to be implemented easily. Some basic protocols for file transfer and peripheral sharing have also been presented.

Further modifications would be to include a parity bit with each slot and packet or to retain a slot for more than one revolution. In the former case the parity can be updated by each station thereby localising errors and failures to a single link, and in the latter the bandwidth available to a user is increased.

CHAPTER 7

SUMMARY AND CONCLUDING REMARKS

7.1 Summary

Recent years have seen the development of local area computer communication networks. This work has been concerned with the study of a number of such networks, and in particular with the comparison of the performance and hardware characteristics of such networks relative to each other.

Initially, the particular features of local networks were considered and compared to those of global networks. It was found that for reasonably sophisticated systems, the differences were mainly due to physical size. A number of local network systems were then chosen for comparison, and these included the ring network based on the empty slot principle implemented at the Computer Laboratory, University of Cambridge. The other ring systems chosen were the token system, where the user can only transmit when in possession of a permission token, the register insertion system, where a fixed length shift register is inserted in the data path on transmission, and the pre-allocated bandwidth system, where each user exclusively possesses a part of the bandwidth. It was found that for the dynamic allocation systems, the performance characteristics were similar, and that delay and throughput were typically governed by the total delay of the transmission lines. The pre-allocated system performed well when the number of nodes was small and the traffic between them homogeneous.

Two broadcast networks were considered: the first based on the Aloha system at the University of Hawaii and the second on the Ethernet

system at Xerox. An analytical model for a two node Aloha network was developed, and exponential and uniform retransmission distributions were compared. It was found that for small systems, the exponential distribution improved throughput. A new algorithm for stabilising and optimising the use of the Aloha channel was presented, the results being verified by analytical analysis and simulation. A technique for modelling the Ethernet was presented based on a computer algebra system. Ring and broadcast systems were contrasted, and it was found that their performance characteristics were similar. Also, a ring contention network was described with delays similar to a broadcast system at low loads but with ring like behaviour at high loads.

It was observed that not only the performance characteristics but also the hardware requirements of the different networks were similar. A design for a general purpose, LSI, station logic chip was thus considered. The chip is microprogrammed and is based on a simple set of micro-instructions which can be used to build any of the networks.

Finally, protocol issues were considered with special reference to the local network at Cambridge. In order to improve the handling of messages, the Cambridge ring was redesigned so that messages competing for a single destination are served within some maximum period of time.

7.2 Conclusions

As local network techniques become better understood, new areas of application will be developed. These range from simple terminal interconnection schemes to complex resource sharing, process control, and distributed computer systems. An important area is that of interconnection of microprocessor systems. As these are developed, for example in aircraft or ship control, a need arises for a rational interconnection

structure,and this is fulfilled by a local network. Another important area is that of digital voice transmission. As this becomes more common in global networks,the need for a cheap and simple local distribution mechanism becomes greater. This process will be accelerated by the arrival of inexpensive LSI devices.

Of all local network structures the Ethernet (CSMA) approach has recently been most popular. This may not continue as the advantages of rings become better understood. With CSMA the delay before transmission (and no clash) is low. However, rings also perform well at low traffic levels, the mean overhead for the token system being half a ring delay. As traffic level increases the CSMA system has to be carefully adjusted to avoid the build up of undelivered traffic, and this probably requires additional hardware. Line utilization can be improved by increasing packet size (also true for the token system); however,larger buffers have to be provided. Furthermore, low level acknowledgements require separate transmissions additionally reducing the bandwidth. It is also not clear that the CSMA transmission problems are simpler. The tap either matches the impedence of the Ether and can bring it down at failure, or it has a high impedence giving rise to inefficiences and a large driving current capable of inducing failure. The contention ring attempts to aleviate some of these problems, but unless designed carefully, is likely to have more complex hardware than either a ring or bus, with little performance gain.

An attractive feature of the Cambridge ring is its ability to transmit data bytes asynchronously, so that buffers may be dispensed with at the receiver since the reception of a message is suspendable at any time. Thus, a direct memory access controller can be designed using redundant memory cycles and no storage. By using phase-locked loops the

ring hardware can be made synchronous, and there is no need for a preamble, nor for a high frequency local clock. A disadvantage of most ring systems is that the delay through each node is at least one bit. In the Cambridge scheme this can be easily reduced (a factor of six posing no problems), and thus a large number of nodes can be supported. However, there is a point at which other network architectures become attractive, and these are discussed below.

7.3 New local network architectures

With the increased use of optical devices the geographical criteria for local networks will become less apparent. However, there are a number of interconnection structures which will become more widespread as the size of local networks increases. As has been shown in the thesis ring and bus networks possess similar performance characteristics, and the delay is related to the number of nodes N. At the other extreme lies the crossbar switch and the completely connected network where delay between any two nodes is constant, but where the hardware rapidly becomes complicated with increasing N. There are a number of schemes which lie between these two extremes and which are suitable when N is large. The circuit switched approach, where every pair of subscribers can converse simultaneously, utilises Nlog(N) interconnections, has worst case delay of log(N) but requires an external computer to set the switches to achieve the desired connection pattern [BE64]. It is thus suitable for use where calls are not changed frequently and represents the opposite extreme from packet switching. A network based on binary high dimensional cubes has been proposed [SU77], with a number of nodes on each side of the cube and with corner nodes used for switching packets between cubes. As the number of nodes increases, there is a tradeoff between increasing

the number of cubes and increasing the number of nodes per side. If there are two nodes per side, then this is the Boolean-n cube network with homogeneous nodes, Nlog(N) interconnections, no setup time, and delays similar to the switched network. However, as the number of nodes increases, each one becomes more complex in proportion to $\log^2(N)$, whereas for the other networks, there is no increase in complexity. Another disadvantage is that transmissions along each link are bidirectional, unlike ring systems and bus systems where the transmission hardware can be simpler (unidirectional).

A scheme can be devised where the delay is less than that for ring systems and where transmission along each line is (unidirectional. Figure 7.1 shows the routing possibilities for a system where each node can transmit to one of two succeeding nodes. The packet format is similar to that for the Cambridge ring, with the address fields replaced by a routing field (the source being decodable from the route). On transmission the routing information can be decoded from the address portion of the packet and can thus be generated rapidly. Otherwise, a simple route table can be kept at the source which allows alternate routing strategies to be adopted. To minimise delay as the packet passes a node, the routing bits are rotated so that the first bit of the route field indicates the next direction. Because there can be two packets arriving for output along a single wire, a packet buffer is required (also for Boolean cube), but for homogeneous data flow this will not be large. Also, if packets become corrupted they might circulate indefinitely; but this can be avoided by either clearing the route bits after use and taking special action for all zeros, or by rotating and replacing with a modular-2 random number so that the route constantly changes. The

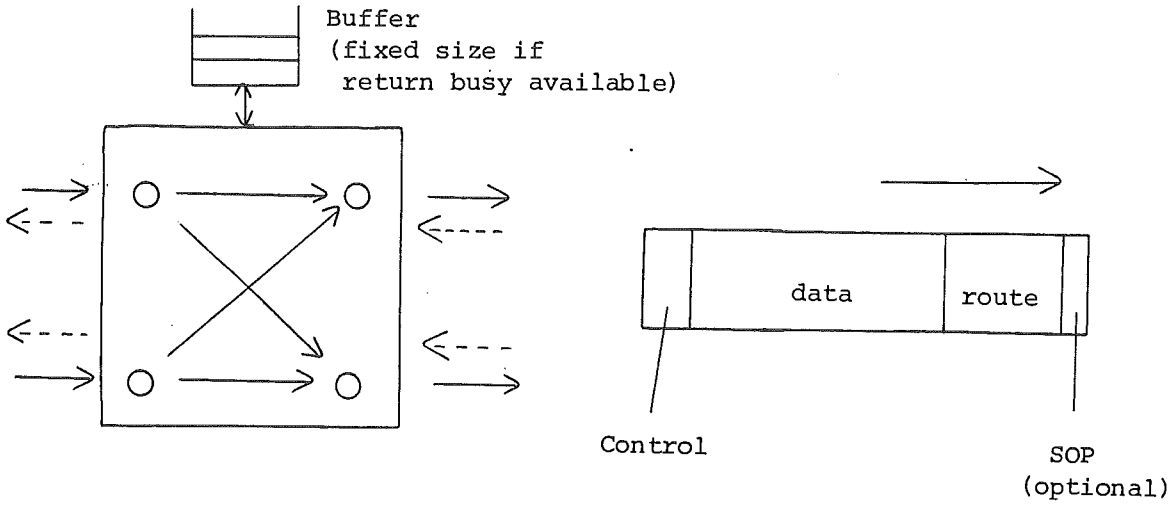


Fig. 7.1 Binary routing network

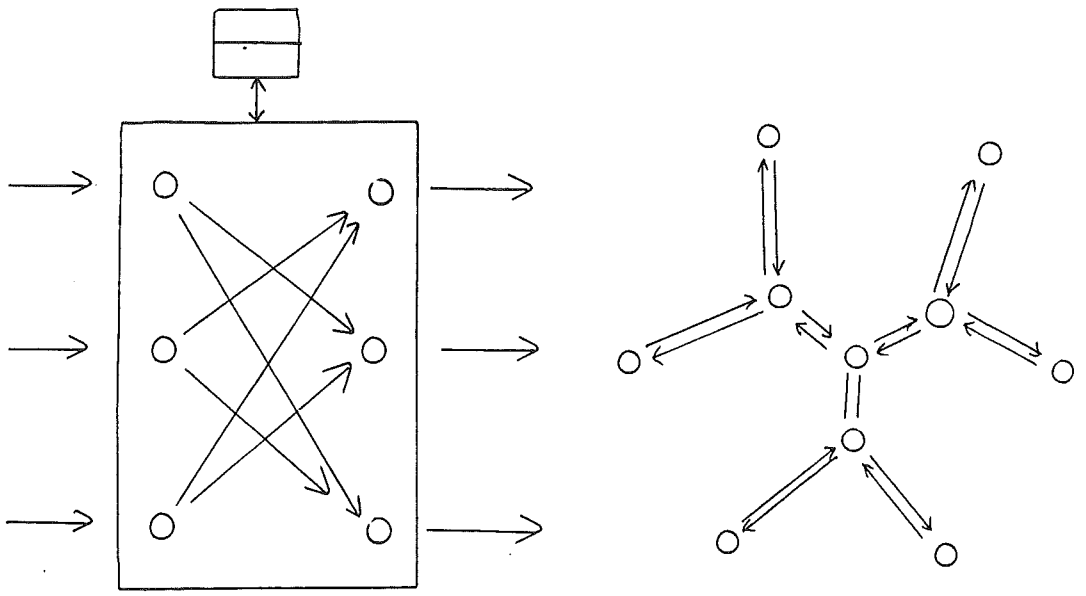


Fig. 7.2 Binary routing network (data and return busy multiplexed)

latter has the further advantage that when a packet is found to be corrupted, its source can always be decoded. The node has two inputs and two outputs used in switching mode; however, it can also be used for attaching host devices to the network by taking one input and one output and connecting them to the device interface. This means that a single unit can be designed to function as a station, which makes an LSI implementation attractive.

In the above scheme there is a finite possibility that data will be lost due to fixed size buffers. This can be remedied by incorporating a return control path for each line and reducing buffer size to two packets. This means that transmissions are bidirectional (or that there is an additional backward path), and congesting traffic is delayed outside the network. Hogging is avoided by alternating the backward busy signal between the two inputs. As the line transmission hardware lies outside the station unit, there is no increase in complexity.

It is possible to further extend the scheme by multiplexing control and data packets. Such a system is shown in Figure 7.2, where there are three inputs and three outputs per node. Each packet is routed left or right as shown, the hardware being similar to the previous scheme. The switching units are compact and again can be used for attaching host devices. Since the network is symmetric, the routing information can be decoded easily.

The networks described above have the property that local transmissions have smaller delays than distant ones. Furthermore, if one of the nodes is disconnected, the network continues to operate in a downgraded mode. New nodes can be connected at any point in the network using homogeneous station units, and delays are of the order of log(N). It is possible to use the station units to construct

rings or buses as they form a subset of the possible network architectures. It is for these reasons that such networks pose interesting problems for the future.

APPENDIX

Summary of Notation

Notation	Definition	Introduced on page
A	Normalised propagation delay	85
Be	Mean electronic delay between two nodes	41
b_{is}	Delay through node i	41
b_{il}	Delay between node i and i+1	41
B_D	Mean ring service delay (bits/packet)	43
B_S	Mean ring service time (secs/packet)	51
D	Denotes deterministic distribution	51
D	Mean transmission delay	52
D_k	Probability of k terminals transmitting succesfully	118
D_{suc}	Mean delay to resolve a clash	92
E	Transmission register data replacement delay (bits)	44
Er	System error rate	79
Ex	Mean number of successful transmissions in one direction	94
Ex_b	Mean value of Ex before slot m	111
Ex_a	Mean value of Ea after slot m	111
e_r	Shift register error rate	78
e_w	Transmission medium error rate	78
F	Number of successful transmissions before higher W	109
F_k	Probability of k transmissions from free nodes	85

Notation	Definition	Introduced on page
f_k	Probability of retransmission in slot k	93
G	Applied traffic	82
G	Denotes general distribution	51
Ga	Number of gap digits in system	44
H_k	Probability of k transmissions from blocked nodes	84
I	Intensity due to all other nodes	54
I_i	Intensity due to node i	54
J	Total traffic due to other nodes	54
K	Size of finite storage	38
L	Uniform retransmission distribution parameter	83
L_n	Number of lost packets	57
L_q	Mean number in queue	52
L_s	Mean number in system	52
M	Denotes exponential distribution	51
Mt	Mean time between clashes	94
m	Conditional value of W_1, W_2 (of retransmission slot)	98
N	Number of nodes in system	41
N_i, N_j	Number of nodes in group i, (group j)	72
$\bar{n}$	Mean number of transmissions in one ring revolution	50
P_l	Proportion of packets lost	57
Ps	Packet size (bits)	42
Ps'	Number of data bits in packet	77
P_{suc}	Probability of success	92
P_n	Probability of n transmission in one ring revolution	53

Notation	Definition	Introduced on page
$p(z)$	Probability of station transmitting in one ring revolution	42
$p(z)_i, p(z)_j$	Probability of station in group i, (group j) transmitting in one ring revolution	75
$p(z)^E_{max}$	Maximum value of p(z) with packet replacement time E	44
Q	Number of slots in system	44
Q_k	Probability of k transmissions clashing	118
R	Mean number of successful transmissions in one direction in free system before a clash	97
S	Normalised throughput (line utilisation)	47
T_l	Total traffic lost or turned away	57
Tr	Ring transmission speed (bits/sec)	42
Tr'	Bandwidth available for data transmission	47
Tr_i, Tr_j	Ring transmission speed for group i, (group j)	75
Tx	Mean data transmission rate from one node (bits/sec)	42
Tx_i, Tx_j	Mean data transmission rate from one node in group i, (group j)	75
t	Time to transmit a packet including control characters	58
W	Packet size (slots)	119
W_1, W_2	Retransmission slot random variables	91
x	Probability of transmission from free node	83
$\bar{x}$	$1 - x$	92
y	Exponential distribution parameter	91
$\bar{y}$	$1 - y$	91
z	Ratio of mean station transmission rate to ring transmission speed	42

Notation	Definition	Introduced on page
$1/\beta$	Mean line busy period	55
$1/\alpha$	Mean line idle period	55
$1/\alpha_i$	Mean line idle period due to station i	55
Φ	Probability of blocked station transmitting when freed	89
Θ	Probability retransmission split for exponential distribution	96
Θ'	Probability retransmissions split for general distribution	98
λ	Mean arrival rate to one node (packets/sec)	51
ρ	Server utilisation	51
μ_1	Mean service rate (random component)	52
μ_2	Service rate (deterministic component)	52
ε	Probability of free station transmitting when packet is available	85

REFERENCES

AB70 Abramson, N. "Another Alternative for Computer Communications", Fall Joint Computer Conference, 1970, pp. 695-702.

AB73 Abramson, N. "Packet Switching With Satellites", National Computer Conference, June 1973, pp. 695-702.

AG77 Agrawala, A.K., R.M. Bryant, and J. Agre, "Analysis of an Ethernet-like Protocol", Technical Report TR-522, University of Maryland, Department of Computer Science, April 1977.

AS75 Ashenhurst, R.L. and R.H. Vonderhoe, "A Hierarchical Network", Datamation, Vol. 21, No. 2, February 1975, pp. 40-44.

BA75 Baskett, F., K.M. Chandy, R.R. Muntz, F.G. Palacios, "Open, Closed and Mixed Networks of Queues with Different Classes of Customers", JACM, Vol. 22, No. 2, Apr. 1975 pp. 248-260.

BE64 Benes, V.E. "Optimal Rearrangeable Multistate Connecting Networks", BSTJ, Vol. 48, No. 7, Jul. 1974, pp. 1641-1656.

BI75 Binder, R., N. Abramson, F. Kuo, A. Okinaka and D. Wax, "Aloha Packet Broadcasting -- A Retrospect", National Computer Conference, May 1975, pp. 215.

BU75 Burchfield, J., R. Tomlinson and M. Beeler, "Functions and Structure of a Packet Radio Station", National Computer Conference, May 1975, pp. 245-251.

CO72 Coker, C.H. "An experimental interconnection of computers through a loop transmission system", Bell Syst. Tech. Jour., Vol. 51, No. 6, July-Aug. 1972, p. 1167.

DA73 Davies, D.W. and D.L.A. Barber, "Communication networks for computers", pub. John Wiley and Sons, 1973.

DE75 Dewis, I.G. "Linking Satellite Processors to a Central Computer Installation", Electronics and Power, 20 Feb. 1975, pp. 171-175.

DOM74 Doman, R.A., and J.R. Kersey, "Synchronous Data Link Control - A perspective", IBM Syst. J. No. 2. 1974, pp. 140-162.

DOW77 Dowson, M. "DEMOS - A Multiprocessor Computer", Comp. Sc. Div., Nat. Phys. Lab., Teddington (to be published).

FARB72 Farber, David J., and Kenneth C. Larson, "The System Architecture of the Distributed Computing System -- The Communications System", Proceedings of the Symposium on Computer-Communications and Teletraffic, New York Polytechnic Press 1972.

FARB73 Farber, David J., et.al. "The Distributed Computing System", Compcon 73, February 1973, pp. 31-34.

FARB75 Farber, David J. "A Ring Network", Datamation, Vol. 21, No. 2., February 1975, pp. 44-46.

FARM69 Farmer, W.D. and E.E. Newhall, "An experimental distributed switching system to handle bursty computer traffic", Proc. ACM Symp. on Data Comm., Pine Mountain, G.A. (October 1969), 1-33.

FAY77 Fayolle G., E. Glenebe, J. Labetoulle, "Stability and Optimal Control of the Packet Switching Boradcast Channel", JACM, Vol. 24, No. 3, July 1977 pp. 375-386.

FEL59 Feller, W. "An Introduction to Probability Theory and its Applications", Vols. 1,2. Pub. Wiley International, 1950.

FERRE Ferreira R.C., D. Vojnovic, "Multi-Mini Computers - A Perspective on the Next Five Years", Infotech State of the Art Report, Future Systems.

FERRA74 Ferranti Ltd., "The Uncommitted logic Array" Gem Mill, Oldham, Lancs.

FIN59 Finch, P.D. "Cyclic Queues With Feedback", Jour. Roy. Stat. Soc., Ser. B., Vol. 21, 1955, pp. 153-157.

FIT75 Fitch, J.P. "Camal Users Manual", University of Cambridge, Computer Laboratory Report, Cambridge 1975.

FO75 Forth, L. et. al. "Null-Protocol Network for Communication Between Digital Devices", Proc. IEE, Vol. 122, No. 8, Aug. 1975 pp. 785-790.

FRAL75a Fralick, S.C., J.C. Farrett, "Technological Considerations for Packet Radio Networks", National Computer Conference May, 1975, pp. 233-244.

FRAL75b Fralick, S.C., D.H. Brandin, F.F. Kuo and C. Harrison, "Digital Terminals for Packet Broadcasting", National Computer Conference, May 1975, pp. 253-261.

FRAN75 Frank, H., I. Gitmand, R. Van Slyke, "Packet Radio System - Network Considerations", National Computer Conference, May 1975, pp. 217-231.

FRAS74a Fraser, A.G. "Spider - A Data Communications Experiment", Comp. Science. Tech. Report No. 23, Bell Labs., N.J.

FRAS74b Fraser, A.G. "Loops for Data Communications", Comp. Science. Tech. Report No. 24, Bell Labs., N.J.

FRAS75a Fraser, A.G. "A Virtual Channel Network", Datamation, Vol. 21, No. 2., February 1975, pp. 51-53.

FRAS75b Fraser, A.G. "Loop transmission systems for data", Comp. Comm. Review, Oct. 1974, Vol. 4, No. 4.X.

FU73 Fuller, S.H., D.K. Sienworek, K.J. Swan, "Computer Modules: An Architecture for large Digital Modules", Comp. Arch. News, Vol. 2, No. 4., Dec. 1975, pp. 231-237.

GO67a Gordon, W.J. "Cyclic Queueing Systems with Restricted Length Queues", Opns. Res., Vol. 15, 1976, pp. 266-277.

GO67b Gordon, W.J., G.F. Newell, "Closed Queueing Systems with Exponential Servers", Opns, Res. 15, 1976, pp. 254-265.

GR71 Graham, R.L., H.O. Pollak, "On the Addressing Problem for Loop Switching", Bell Syst. Tech. Jour., Vol. 50, No. 8, Oct. 1971, pp. 2495-2519.

HAF74a Hafner, E.R. "Digital Communication Loops - A Survey", Proc. 1974 Int. Zurich Seminar, pp. D1.1 - D1.7.

HAF74b Hafner, E.R., Z. Nendal, M. Tschantz, "A Digital Loop Communication System", IEEE. Trans. on Comm., Vol. 22, pp. 877-881, June 1974.

HAL75 Halfin, S. "An Approximate Method for Calculating Delays for a Family of Cyclic Type Queues", BSTJ, Vol. 54, No. 10, Dec. 1975, pp. 1733-1753.

HANS72 Hansler, E., et.al. "Optimising the Reliability in Centralised Computer Networks", IEEE Trans. on Comm., Vol. COM-20, June 1972, pp. 640-644.

HANK77 Hanks, J.P. "Mitrenet-Introduction and Overview", Mitre Corporation Technical Report NTR-3382, Vol. 1-4, Bedford, Mass.

HAY71 Hayes, J.F. and D.N. Sherman, "Traffic analysis of a ring switched data transmission system", Bell Syst. Tech. Jour., Vol. 50, No. 9, Nov. 1971, p. 2947.

HAY72 Hayes, J.F., D.N. Sherman, "A study of Data Multiplexing Techniques and Delay Performance", BSTJ, Vol. 51, No. 9, Nov. 1972, pp. 1983-2001.

HAY73 Hayes, J.F. "Modelling an experimental computer communication network", Proc. Datacomm 73, St. Petersburg, Fla, Nov. 1973.

HAY74 Hayes, J.F. "Performance Models of an Experimental Computer Communication Network", BSTJ, Feb. 1974, pp. 225-251.

HEC77 Heckel, P.G., B.W. Lampson, "A Terminal Oriented Communication System", Comm. of the ACM, Vol. 20, No. 7, July 1977, pp. 486-494.

HEI76 Heitmeyer, C.L., J.H. Kullback, J.E. Shore, "A Survey of Packet Switching Techniques for Broadcast Media", Naval Res. Lab. Rep. No. 8035, Washington D.C.

HOB77 Hobbs, I.C., "Hardware Technology Trends", Infotech State of the Art Report, Future Systems

HOP78 Hopper, A. "Data Ring at Computer Laboratory, University of Cambridge", Report of Workshop on Local Area Computer Networking., Aug. 1977, National Bureau of Standards, Gaithesburg, Md.

HW75 Hwa, H.R., "A Conflict Free Aloha System", Sydney/Aloha Working Paper 3, Comp. Sc. Dept. Univ. of Sydney, Australia, Jan. 1975.

KAH75 Kahn, Robert E. "The Organization of Computer Resources into a Packet Radio Network" National Computer Conference, 1975, pp. 177-186.

KAY72 Kaye, A.R., "Analysis of a Distributed Control Loop for Data Transmission", Proc. Symp. Computer Communication Networks and Teletraffic, New York Polytechnic Press, 1972.

KL72 Kleinrock, L. "Communication nets", pub. Dover Publications Inc., 1972.

KL75a Kleinrock, L., F.A. Tobagi, "Packet Switching in Radio Channels", Parts 1,2, IEEE Trans on Comm., Vol. Com-23, No. 12, Dec. 1975, Part 3 IEEE Trans on Comm., Vol. Com-24, No. 8, Aug. 1976, Part 4 IEEE Trans on Comm., Vol. Com-25, No. 9, Oct. 1977.

KL75b Kleinrock, L. "Queueing Systems", Vol. 1,2 Pub. Wiley Interscience, 1975.

KL75c Kleinrock, L., F. Tobagi, "Random Access Techniques for Data Transmission Over Packet Switched Radio Channels", National Computer Conference, May 1975, pp. 187-202.

KL76 Kleinrock, L., W.E. Naylor, H. Opderbeck, "A Study of Line Overhead in the Arpanet", Comm. of the ACM, Vol. 19, No. 1, Jan. 1976, pp. 3-13.

KO58 Koenigsberg, E., "Cyclic Queues", Oper. Res. Quart., Vol. 9, No. 1, 1958, pp. 22-35.

KR72 Kropfl, W.J. "An experimental data block switching system", Bell Syst. Tech. Jour., Vol. 51, No. 6, July-Aug. 1972, p. 1147.

LID76 Lidinsky, William P., "The Argonne Intra-Laboratory Network", Proceedings, Berkeley Workshop on Distributed Data Management and Computer Networks, Lawrence Berkeley Laboratory, Berkeley, California, May 25-26, 1976, pp. 263-275.

LIU77 Liu, M.T., C.C. Reames, "Message Communication Protocol and Operating System Design for the Distributed Loop Computer Network", Proc. 4th Ann. Symp. on Comp. Arch., March 1977, pp. 193-200.

MA57a Mack, C., T. Murphy, N.L. Webb, "The Efficiency of N Machines Unindirectionally Patrolled by One Operative when Walking Time and Repair Times are Constant", Jour. Roy. Stat. Soc., Ser B., No. 1, pp. 166-172.

MA57b Mack, C. "The Efficiency of N Machines Unindirectionally Patrolled by One Operative When Walking Time is Constant and Repair Times are Variable", Jour. Roy. Stat. Soc., Ser. B., No. 1, pp. 173-178.

METC76 Metcalfe, R.M., D.R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks", Communications of the ACM, Vol. 19, No. 7, July 1976, pp. 395-404.

METZ76 Metzner, J.J. "On Improving Utilisation in Aloha Networks", IEEE Trans. on Comm., Apr. 1976, Vol. 24, pp. 447-448.

MO77 Mockapetris, P.V., M.R. Lyle and D.J. Farber, "On the Design of Local Network Interfaces", NCC, Aug. 1977.

NBS77 National Bureau of Standards, "Report of Workshop on Local Area Computer Networks", Aug. 1977, Gaithesburg, MD.

PE75 Pennotti, M.C. "Congestion Control in Store and Forward Tandem Links", IEEE Trans. on Comm., Vol. Com-23, No. 12, Dec. 1975, pp. 1434-1442.

PI72a Pierce, J.R., "Network for block switching of data", Bell Syst. Tech. Jour., Vol. 51, No. 6., July-Aug. 1972, p. 1133.

PI72b Pierce, J.R. "How far can data loops go?", IEEE Trans. on Comm., Vol. Com-20, No. 3, June 1972, p. 527.

POT71 Potvin, J.M.T., "The star-ring system of loosely coupled digital devices", Univ. Toronto, Comp. Syst. Research Group, Report No. 7, April 1971.

POU75 Pouzin, L. "Basic Elements on a Network Data Link Control Procedure (NDLC)", ACM Comp. Comm. Review, Vol. 5., No. 1, Jan. 1975, pp. 6-23.

RE75a Reames, C.C., M.T. Liu, "A loop network for simultaneous transmission of variable length messages", Proc. of the 3rd Ann. Symp. on Comp. Arch., SIGARCH Houston, Jan. 1975.

RE75b Reames, C.C., M.T. Liu, "Design and Simulation of the Distributed Loop Computer Network", Proc. 3rd Ann. Symp. Comp. Arch., Jan. 1975, p. 215.

RO73 Roberts, L.G., "Dynamic Allocation of Satellite Capacity Through Packet Reservations", National Computer Conference, June 1973, pp. 711-716.

RY76 Rybczynski, A., D. Wessler, R. Despres., J. Wedlake, "A new Communication Protocol for Accessing Data Networks - The International Packet Mode Interface", NCC 1976, pp. 477-482.

SO76 Sonner, R. "Cobus, A Firmware Controlled Data Transmission System", Second Symp. on Micro Arch. 1976, North-Holland Pub. Co.

STE7O Steward, E.H. "A loop Transmission System", IBM Research Triangle Park, North Carolina 1976.

STR78 Stroustrup, B. "On Unifying Module Interfaces", Oper. Syst. Rev., Vol. 12, No. 1, Jan. 1978, pp. 90-98.

SU77 Sullivan, H., T.R. Bashkow, "A Large Scale,Homogeneous, Fully Distributed Parallel Machine", Comp. Arch. News., Vol. 5., No. 7, March 1977, pp. 105-177.

VA74 Vanderhoe, R.H. "Activity on the MISS project", Quart. Rep. No. 43, Inst. for Comp. Research, The University of Chicago, Nov. 1974.

WEB76 Webster, W.J., "Performance of Phase Locked Loops in the Presence of Fading Communication Channels", IEEE Trans on Comm., Vol. Com-24, No. 5., May 1976, pp. 487-499.

WEC76 Wecker, S., "The Design of Decnet - A General Purpose Network Base", Electro/76, Boston Mass., May 1976.

WEL71 Weller, D.R., "A Loop Communication System For I/O to a Small Multiuser Computer", Proc. IEEE International Comp. Soc. Conf., Sept. 1971.

WE72 West, L.P., "Loop Transmission Control Structures", IEEE Trans on Comm., Vol. Com-2O, No. 3, June 1972, pp. 531-539.

WES77 West, A.R., "A Broadcast Packet Switched Computer Communications Network", Design Progress Report, Queen Mary College, London.

WI75 Wilkes, M.V., "Communication Using a Digital Ring", Proceedings of the Pacific Area Computer Communication Network System Symposium, August 1975, pp. 47-56.

WU75 Wu, R.M., Y.B. Cheu, "Analysis of a Loop Transmission System with Round Robin Scheduling of Services", IBM J. Res. Develop, Sept. 1975, pp. 486-493.

YA75 Yajima, S., Y. Kambayashi, K. Iwama, "Optically Linked Laboratory Computer Network - Labolink", Proc. of the Tenth Hawaii Int. Conf. on System Sciences, Jan. 1977, pp. 1-4.

ZA74 Zafiropulo, P. "Performance Evaluation of Reliability Improvement Techniques for Single Loop Communications Systems", IEEE Trans. on Comm., Vol. Com-22, No. 6., June 1974, p. 742.

ZI77 Zilog Inc. "Z80-S10 Product Specification", Cupertino, Calif.