Open Research Online

On the management and performance of a class of local area networks

Thesis

oro.open.ac.uk

On the Management and Performance of a Class of Local Area Networks

BY

MOHAMMAD MAHBOOB YASIN
M.Sc.(Physics) 1975
M.Engg.(N. Tech) 1977
M.Sc.(Comp. Sci) 1983

This thesis is presented as a requirement for the Degree of Doctor of Philosophy. The work reported on here has not been presented for any other degree.

Faculty of Mathematics,

The Open University,

Walton Hall,

Milton Keynes

23rd October,1986

HIGHER DEGREES OFFICE

LIBRARY AUTHORISATION

STUDENT: MOHAMMAD MAHBOOB YASIN   SERIAL NO: ....

DEGREE: Ph. D.

TITLE OF THESIS: ON THE MANAGEMENT AND PERFORMANCE OF A CLASS OF LOCAL AREA NETWORKS

I confirm that I am willing that my thesis be made available to readers and may be photocopied, subject to the discretion of the Librarian.

Signed: Yasin   Date: 20-10-88

Abstract

This dissertation is concerned with the management and performance issues of those register-insertion (R-I) ring type local area networks (LANs) which employ the *message removal by destination node* protocol. A typical example of such a network is the Distributed Loop Computer Network (DLCN), developed by Liu et al for fundamental research in the field of distributed computing. After considering the management issues of low cost R-I LANs, the research deals with performance analysis of the dynamically reconfigurable register-insertion (DRR) network.

In the first part of this research, a set of management functions is identified which are desirable and can be provided economically by a low cost LAN. A unique feature of the DLCN is that messages are removed from the network by the destination node, that is, messages do not travel whole of the loop. Therefore, it is not possible for a special control node to monitor the data traffic on the network without providing support functionality in each network access unit (NAU). The minimum functionality which must be provided in each NAU is identified in the thesis. A skeleton network was implemented to verify the feasibility of the proposed scheme. A paper describing the findings of this research was published and is reproduced as appendix A.

In the second part of this research, a new feature of the DLCN network is introduced, namely, that the performance of a network employing removal by destination protocol can be improved by reconfiguring the network in a particular way. A methodology to find the optimal configuration is developed and is shown, by worked examples, to lead to improved performance. The findings of this research are particularly applicable to the dynamically reconfigurable register-insertion (DRR) network. A paper dealing with the optimisation of a hypothetical fully connected DRR network has been accepted for publication. Another paper, which considers the general case of less than fully connected DRR networks, is to be published. Both papers are reproduced as appendices B and C.

Finally, a performance study of the DRR network is undertaken. As there seems to be no published attempt at formal analysis or simulation of a DRR network, a survey of literature dealing with performance study of the basic DLCN is performed. A simulation model of the DRR was then developed and implemented to verify the results arrived at in the previous section. Later, a queueing model of the DRR network, based on the work of Bux and Schlatter [7] is developed and analysed. Both simulation and analysis support the claim that the performance of a DRR network can be improved by adopting the configuration strategy developed in this thesis.

Preface

Three papers have arisen out of the research work reported on in this thesis. These papers form a part of the thesis and are reproduced as appendices A through C. The material presented in the papers has not been repeated in the body of the thesis which contains explanatory material with frequent references to the appendices.

to my wife *Fozia*

Acknowledgements

Contents

List of Figures

List of Symbols and Abbreviations

Symbol: Definition

Γ the average transfer time per message for the whole network.

Γ_i the average transfer time for node i.

Γ_{ij} the average transfer time of messages originating at node i and destined for node j.

Λ the total through traffic on the network.

λ mean arrival rate of messages at a node.

λ_i mean arrival rate of messages at node i.

λ_i^e mean arrival rate of external messages at node i.

λ_i^r mean arrival rate of messages to be relayed at node i.

λp mean arrival rate of messages of priority p.

$\dfrac{1}{\mu}$ mean departure rate of messages.

ρ_i traffic intensity at node i.

ρ_i^e traffic intensity of external messages at node i.

ρ_i^r traffic intensity of relay messages at node i.

τ latency of a node $= \tau_{in} + \tau_{out}$.

τ_{in} delay caused by the input section of a NAU.

τ_{out} fixed delay caused by the output section of a NAU.

τ_i latency of node i.

ATDM   Asynchronous time division multiplexing technique.

C   the channel bandwidth or the transmission speed of a channel.

CSMA   the collision-sense-multiple-access protocol.

DDLCN   the distributed double loop computer network.

DRR	the dynamically reconfigurable register-insertion network.
DLCN	the distributed loop computer network.
d_{ij}	the 0/1 variable determining the relative positioning of nodes on a ring.
E[x]	the expected value of variable x.
E[T]	the expected value of T, the message transmission time.
$E[T^2]$	expected value of square of T or variance of T.
e_{ij}	probability that a message originating at node i passes through node j.
HOL	head-of-line queueing discipline.
I(i,k)	set of nodes encountered when travelling from node i to node k in the direction of data transfer on the ring.
IB	the input buffer of a NAU.
IPC	inter-process communication.
LAN	local area network.
l_{ij}	the distance, in number of arcs, between nodes i and j in the direction of data flow.
MDLCN	modified distributed loop computer network.
N	the number of nodes in a network.
NAU	network access unit.
n	mean number of messages in the system (queue + service).
OB	the output buffer in a NAU.
P	the number of distinct priority classes in a system.
p_k	probability that a system is in state k.
p_{ij}	probability that a message originating at node i is destined for node j.
$[p_{ij}]$	the probability matrix of a network.
RC	the receiver section of a NAU.

R_{ij}	average character rate originating at node i and passing through node j.
R-I	register-insertion ring.
RSR	receiver shift register.
S	set of all nodes, {1,2, ..., N}.
SAP	service access point.
STDM	synchronous time-division multiplexing.
T	transmission time of a message.
V	the volume distribution matrix.
v_{ij}	the volume of traffic originating at node i and destined for node j.
VSR	variable length shift register.
W_p	mean waiting time for messages of priority p.
W_i^e	mean waiting time of external messages at node i.
W_i^r	mean waiting time of relay messages at node i.

1. Introduction

The work reported on here begins by investigating. in broad terms, local area network operating systems. In particular, the research looks at management issues with a view to providing a user-friendly interface to the network. From this investigation comes the idea of a network manager. and the issues of distributed versus centralized management facilities. One function of a network management facility is to gather statistics about network usage. The question then arises as to the purpose for which the statistics are to be gathered. One reason is to inform management (people) about the effectiveness of the network. It is then possible to pose the question: given statistics about a running system can they be used to improve the performance of the system? It was noticed that the performance of a register-insertion ring network is sensitive to the pattern of data traffic among its nodes. so it is natural to ask what the effect of altering physically the relative positions of nodes on the network would be in reducing through traffic. thereby improving network performance.

These investigations lead naturally into other work on reconfigurable networks. proposed mainly to improve the reliability of ring networks. The results of the work reported on here show how the statistics can be used to determine an optimal configuration. which can also be applied to reconfigurable networks.

Having a method for determining an optimal configuration. analytical and simulation techniques were then used to verify the work.

Chapter 2 introduces the field of local area networks with particular emphasis on register-insertion ring networks.

In chapter 3, network management issues are introduced, and a survey of the literature is given on this subject. Also reviewed is one of our papers [54, Appendix A] which describes our findings on this subject.

Chapter 4 reviews our first paper on the optimization problem [57, Appendix B], where the simplified case of a fully-connected physical network is considered. The important notions of a logical ring and optimality of configuration, and a relative measure to compare alternative configurations are introduced. The problem of determining the optimal configuration for a given situation is formulated in a way which is suitable for integer programming techniques. It is shown by examples, and confirmed by simulation, that the suggested approach results in lower transfer delays across the network.

Chapter 5 gives a review of our second paper on the optimization issue [56, Appendix C] which extends the integer programming formulation to the more general case of less-than-fully connected networks. Worked examples show that the solutions obtained by integer programming techniques result in appropriately constrained logical ring networks.

The next step was to determine the effectiveness of the optimality by analytical techniques. Here a survey of the published literature on the performance analysis of the register-insertion ring network is presented in chapter 6.

There appears to be no literature on the performance analysis of the dynamically reconfigurable register-insertion ring (DRR) network. Therefore, in chapter 7, a queueing model is used to analyse the performance of a DRR network. This is based on the Bux and Schlatter model [7] of a register-insertion ring network. Analysis shows that

an intuitive choice of the relative performance measure, introduced in appendix B, corresponds to an important parameter: the total through traffic in the network. It is also shown that by minimizing this parameter the average message transfer time across the network decreases.

Chapters 8 presents the methods of solution and experimental results of more substatial examples.

Concluding remarks appear in chapter 9, where it is concluded that the novel use of the network usage statistics for optimization of logical ring configuration of register-insertion networks may result in significant improvement in network performance. It is also concluded, by referring to the work being done on dynamically reconfigurable networks, that the proposed approach is practicable. Finally, some directions for further research in this field are proposed.

The papers describing the findings of this research study are reproduced as Appendices A to C.

2. Local Area Networks (LANs)

This chapter provides an introduction to the field of local area networks (LANs). A review of various approaches to local networking is presented. Emphasis is given to a particular LAN topology, namely a loop.

Discussion in this chapter is restricted to single loops. an introduction to multi-connected loop networks appears in chapter 7 which deals with the performance analysis of reconfigurable networks.

2.1. Introduction

With the proliferation of low-priced personal computers and increased acceptance of decentralized management. there is an ever-growing demand for networking systems to provide an integrated access to the distributed corporate data-base. The scope of network systems ranges from terminal-oriented networks providing remote terminal access to a selection of computer systems. through computer networks providing computer-to-computer communication at peer level. to the more sophisticated distributed database systems.

In what follows. the term network refers to peer-level inter-connection of computing equipment such as computers. printers. file stores and terminals. The various sites to be networked are inter-connected by some transmission medium such as an optical fibre. a cable. a telephone network. or a sattelite link.

In recent years, much research on computer networks has focussed on a particular subfield called local area networks. A local area network (LAN) interconnects computers and terminals scattered over a limited geographical area such as a university campus, a factory building or an office plaza. This contrasts with the wide area networks (WANs) which span countries and even continents as is the case with ARPANET [52] and EURONET [9]. This is a very significant difference which makes LANs to be an entirely different breed from WANs in respect of sociology, economics and technology.

In social terms, a LAN affects the corporate structure rather than just the data processing (dp) department. Introduction of a LAN brings some order to the chaotic situation created by the spread of cheap and often incompatible personal computers. These computers have become cheap enough (and still coming down in price!) to be authorized by section heads without consultation with the central dp department. However, once a particular LAN is chosen, compatibility with the LAN serves as a significant requirement for convergence.

In economic terms, a LAN interconnects relatively inexpensive computing equipment and, therefore, the network interfaces have also to be cheap. Because of the small area spanned by a LAN it is economically feasible to lay down high quality cable and even provide redundant links.

Technologically, as well, LANs have very different characteristics from those of wide area networks. LANs utilize optical fibre or high quality cable to inter-connect equipment. Therefore, they provide high data transmission rates together with very low error rates. Also, since the overall length of the transmission medium is limited to a few kilometers, the end-to-end propagation delay is small as is the network transfer

delay. Thus, LANs can support sufficiently high data rates at acceptable transfer delays to be used as a backing store or as a replacement of a direct link for interactive computing.

Terminology

The terminology used throughout this thesis is introduced in the following paragraphs with reference to figure 2.1 for explanation.

In a local area network, the computing equipment, called the *host*, is connected to the cable, the *physical medium*, by means of a *network access unit (NAU)*. An NAU consists of a hardware interface, which connects directly to the physical medium, and appropriate software to affect the actual data transfer. The combination of an NAU and its attached host is called a *node* or a *station*. The complete system consisting of the physical medium, NAUs and hosts is called a computer communications network, whereas the subsystem consisting of the physical medium and NAUs is referred to as the communications subnetwork. Apart from chapter 3, where the issues of computer communications network management are considered, the rest of this study is concerned with the performance enhancement of the underlying communications subnetwork. Therefore, the term network generally refers to the communications subnetwork.

The pattern in which the NAUs of a network are connected by the physical medium is referred to as the *physical topology* of the network. A segment of the physical medium connecting any two NAUs is called a link. Thus, the physical topology of a network is the graph consisting of a number of NAUs and a number of links. Many physical
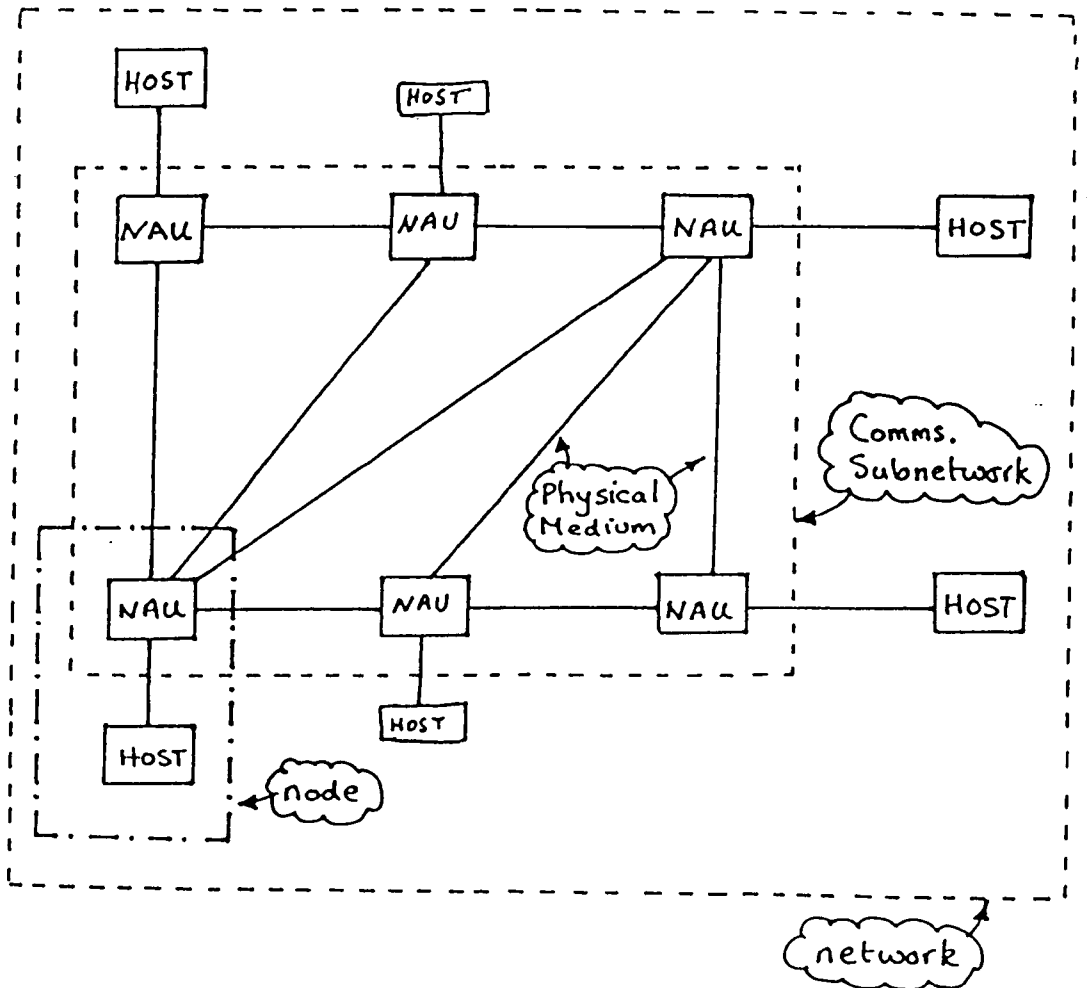
Fig. 2.1. A typical local area network (LAN)

topologies have been proposed for laying down local area networks such as star, bus,
ring or mesh (see figure 2.2).

At any instant of normal operation, data transfer over a link between two nodes can
be performed in one specific direction only. In the case of certain mesh networks, only
a selection of available links may be used to carry data, the rest being redundant links
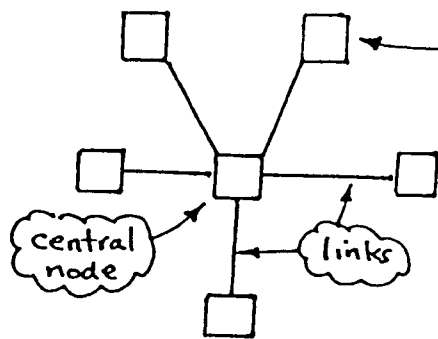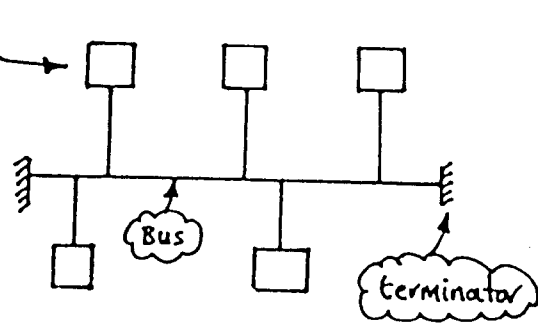
(a) <u>Star Network</u>   (b) <u>Bus Network</u>
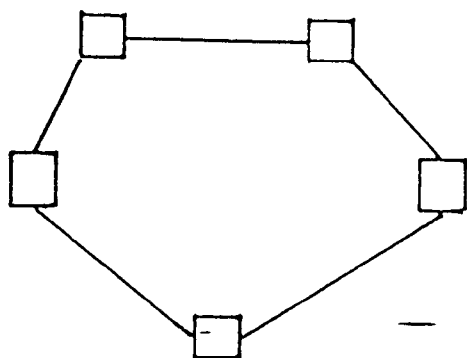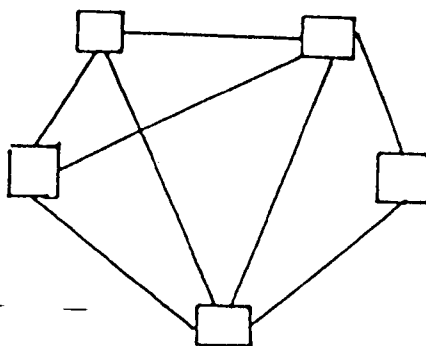
(c) <u>Ring Network</u>   (d) <u>Mesh Network</u>

Fig. 2.2. Different LAN topologies

- to fall back on in case of failure of a previously selected link. A link that is currently selected to carry data is termed a *data path*. The directed graph consisting of NAUs and data paths is called the *logical topology* of a network. Figure 2.3 shows a six node network whose physical topology, consisting of 11 links, shown in the figure by

dashed lines, is a mesh, while the set of the six currently selected data paths, drawn as solid arrows in figure 2.3, forms a logical ring.
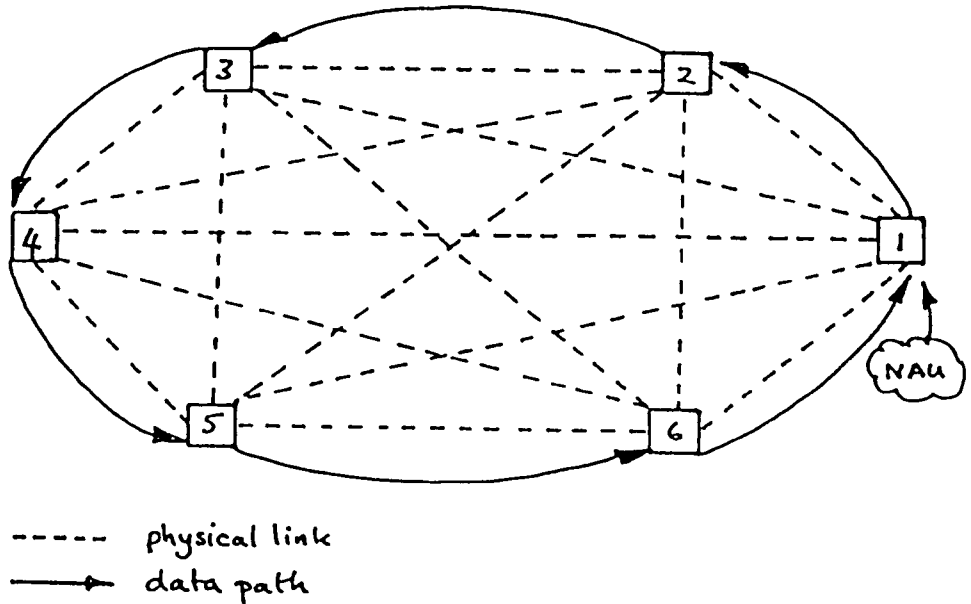
Fig. 2.3. Physical and logical topologies of a network

The difference between the physical and logical topologies of a network occurs only whilst considering reconfigurable networks. The capability to reconfigure a network may be *static*, as in the case of IBM's System Network Architecture (SNA) [16] where a logical topology can be selected during network initialization only, or it may be *dynamic*, as in the case of the dynamically reconfigurable network proposed by Lee in 1979 [26] where reconfiguration may occur during normal network operation without affecting on-going communication.

Apart from the chapters referring to reconfigurable networks, the term network topology will be used to refer to the physical as well as the logical topologies of the network, both being identical.

Unless a network is fully connected, i.e. there exists a bi-directional data path between each and every pair of nodes, certain data paths will have to be shared among several pairs of communicating nodes. The onus of obtaining access to a shared link is put on the transmitting node, and the methodology adopted to determine which of the contending nodes wishing to transmit at any time will be actually allowed to transmit over a certain data path is known as the *access protocol*.

2.2. LAN classification

Local area networks (LANs) may be characterized by a variety of parameters. For example, (1) whether the physical medium is a twisted-pair cable, a co-axial cable or an optical fibre, (2) whether the data signal is transmitted after modulation over a carrier high frequency signal (*broad-band*) as in cable television networks, or it is being-transmitted un-modulated (*base-band*) as in Ethernet, (3) whether it is a high speed network using data transmission rates of 50 - 100 MHz, a medium speed network utilizing transmission rates of 1 - 10 MHz, or a slow speed network with transmission rates in kilo-Hertz. Another classification, which has emerged recently, determines the cost per connection and the initial cost of the network, as low cost networks such as Cheapernet carve out a market-niche separate from its expensive version, the Ethernet.

An important classification of network architecture is based on the network topology. Three topologies have been studied in relation to the local area networks, a) star, b) bus, and c) ring. A further sub-classification within each topology is based upon the

access protocol employed. The following paragraphs discuss each of these topologies in some detail.

A star network (figue 2.2a) has a control node at the hub and all other nodes are connected to the spokes emanating from the hub. All messages are routed via the control node which first receives each message in full and then relays it to the desired destination. This mode of operation is called the *store and forward* mode. In a star network, the control node is a shared resource, access to which must be co-ordinated to avoid chaos. The access protocol can be either polled, interrupt-based or synchronous. In the first case, the control node polls each node in turn to see if it has any data to transmit. In the second case, a node wishing to transmit sends an interrupt to the control node which serves the interrupt at its own convenience by listening to the transmitter. In the case of synchronous access, each node has a time-slot reserved for its exclusive use.

The star network suffers from several limitations which do not make it a popular choice for LANs. Firstly, the control node is a critical resource whose failure disables the whole network. Second, the control node becomes a bottle-neck and may lead to —unacceptably high delays under moderate loads. Finally, providing bi-directional data paths from the control node to all other nodes can be an expensive exercise even for a medium sized network.

In a *bus* type of LAN (figure 2.2b), the physical medium, invariably a length of co-axial cable, is terminated at both ends and has passive taps for connection of nodes. Messages are broadcast to all nodes and each node must be able to detect messages destined for it. Here, the physical medium, also known as the *bus*, is the shared resource and access to the bus must be regulated. There are two types of protocols for gaining access to the bus, deterministic and random.

Among deterministic protocols are the synchronous access method and the demand sharing access method. The former allocates a fixed sized time-slot to each node in turn, and is known as *synchronous time-division multiplexing (STDM)*. Clearly, if a node does not have anything to transmit its time-slot is wasted. Therefore, the STDM protocol is wasteful of channel bandwidth. A more efficient alternative for deterministic access is the demand based *token bus* mechanism employed by Datapoint in its commercial LAN product, ARCNET. Here, a special pattern, the *token*, passes[52] from node to node. A node in receipt of the token can pass the token to the next node if it has nothing to transmit, or else can transmit a variable length message followed by the token.

More attention has been focussed on the random access protocols for access to the bus type LANs. Purely random access, as in the ALOHA network, is very wasteful of channel bandwidth, and is of very limited interest for LAN applications. A refinement of the ALOHA scheme is the *carrier sense multiple access (CSMA)* technique where a node wishing to transmit first listens to the bus and backs off for a random period of time if the bus is busy, thereby avoiding wasteful collisions. However, as a signal takes some time to travel from one end of the bus to the other, it is possible for a node to sense the bus to be free a short time after another node has started to transmit, resulting in a collision. A third scheme requires the nodes to keep on listening while transmitting, thus detecting collisions at an early stage. The nodes which detect a collision back off for a random period of time and try again. This is known as *carrier sense multiple access with collision detection (CSMA/CD)* protocol. The commercially successful Ethernet series of LANs is based on the CSMA/CD-protocol. Performance studies of Ethernet have shown that this protocol suffers from severe deterioration of network throughput at medium to high loads.

Ethernet has been a commercial success mainly because of the strong backing by computer manufacturers – it was developed by the powerful trio of Digital Equipment Corporation, Intel and Xerox. Such was its influence on the research work on other types of networks, particularly the rival ring type LANs, that Saltzer et al [50] published a paper in 1981 in defence of their choice of a ring LAN at MIT. They compared ring and bus topologies on a variety of operational and subtle technical grounds and concluded that there is substantial technical interest in continuing to develop the ring technology. More recently, the announcement of a ring type LAN from the big blue IBM for its range of personal computers is expected to tip the balance in favour of ring technology with several commercial products expected to appear shortly.

There are two major survey articles dealing with ring type local area networks. First, by Penny and Baghdadi [40, 41], was published in two parts in 1979 and the second, by Liu and Rouse [32], was published in 1984. Part 1 of the former discusses twelve different loop architectures in detail with particular emphasis on the topology and access mechanisms, while the second part discusses technical features such as timing, synchronisation and reliability. A significant contribution of the survey by Penny and Baghdadi, is an extensive review of performance studies of ring networks published until 1978. The authors also compare loop networks with alternative networking approaches and conclude that loops have advantages of simple routing compared with mesh networks and the capability of being extended over large areas compared with bus type networks. The latter survey, by Liu and Rouse, discusses the salient features and technical novelties of ten different loop networking approaches which appear to be variations of three basic loop architectures, namely the slotted ring, the token ring and the register-insertion ring. A performance evaluation of the basic ring networks and the CSMA/CD bus architecture is performed which shows that ring networks have lower average message transmission delays compared to the CSMA/CD bus. Another important factor mentioned in favour of ring networks is that the response time of ring networks is predictable for varying traffic loads on the network. Liu argues that

this property makes ring networks suitable for heavy traffic conditions and real-time applications.

A *ring* network (figure 2.2c) is a sequence of point-to-point data paths closed on itself. Messages travel across the loop from node to node, each node being capable of detecting messages destined for itself. Three ring access protocols have received considerable research interest. a) token ring, b) slotted ring and c) register-insertion ring.

A ring network with a circulating one bit token controlling access to the ring was proposed by Farmer and Newhall in 1969 [13]. Its operation is similar to that of the token bus discussed above. There is one exception, however. In a ring network, messages keep on circulating until they are removed. Therefore, a mechanism is required to remove the messages from the ring after they have been received at the destination node. In a token ring network messages are removed by the originating node. Clearly, when a message has passed its destination, it ceases to be of any use. Therefore, a message travelling from the destination node back to the source node wastes network resources. For random data traffic, a message will, on average, travel half way around the loop before reaching its destination. Thus, half of the loop capacity is wasted. However, since only one node can transmit at any time, removal by the destination node can not recover the wasted capacity. A modified approach was presented by Mirabella and Salvo in 1983 [38] as the *multi-packet token passing ring (MPTPR)* , which is a hybrid between a token ring and a register-insertion ring employing removal by destination.

The *slotted ring*, first proposed by Pierce in 1974 [42], requires that an integral number of fixed length time-slots circulate the ring. A one bit flag at the start of a slot indicates whether the slot is empty or full. A node wishing to transmit waits for an empty

slot to arrive. It then sets the flag to indicate full status and writes its own message in the data portion of the slot. Downstream nodes find the slot to be full and relay it onwards. The destination node copies the message into its buffer store whilst relaying it, still as a full slot, to the next node. Eventually the slot arrives at the originaing node which marks it empty by resetting the *full/empty* flag. It is to be noted that the only delay required at each node is a one bit time equivalent in order to test and, if required, set the flag bit. It is not necessary to further delay the circulating slot at the destination node as the destination copies the message contained therein *on the fly*, i.e. at the same time as relaying it. Also, the originator does not need to perform any address checking before emptying its slot, since there are a fixed number of slots on the ring and the originating node can recognise its slot by counting the number of passing slots. Clearly, messages longer than the data portion of a slot have to be divided into smaller packets, thereby necessitating additional processing to dis-assemble a message at the source node and assemble it again at the destination node. Also, messages, and portions thereof, smaller than the data size of a slot are wasteful of channel throughput. Finally, as each slot is an independent identity on the communications subnet, there is additional overhead of addressing information per slot which is reduced to a per message overhead in case of token and register-insertion rings.

An example of successful implementation of the slotted ring protocol is the Cambridge Ring LAN [62] which operates at 10 MHz. Here, one slot of 40-bits length, called a *packet*, circulates the ring and carries sixteen bits of data and other information relevant to the transfer. The last couple of bits are used to carry low-level acknowledgement signals back to the originator, thus providing a reliable packet delivery service or a *datagram* service.

A *register-insertion ring* (figure 2.4) is more flexible than the token ring or the slotted ring as it provides for the transmission of multiple variable length messages on the

ring at the same time. This is accomplished by providing a shift register in each net-work access unit. A node wishing to transmit can create an empty slot by switching its shift register into the ring, thereby gaining almost immediate access to the ring.
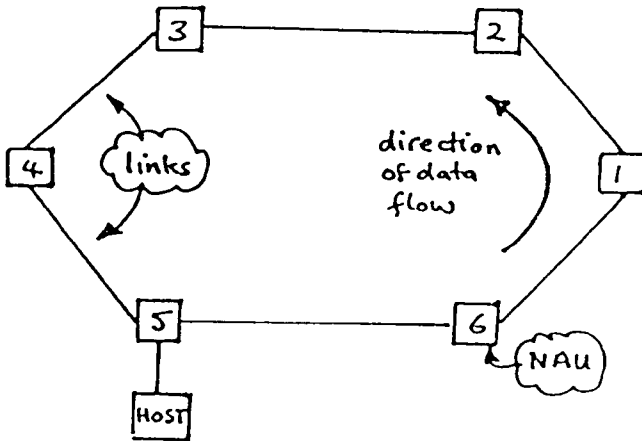
Fig. 2.4. A register-insertion ring network

In its original form as a *static* insertion ring, proposed by Hafner et al in 1974 [18], each NAU incorporates a fixed length shift register (figure 2.5). When a node has a message to transmit, it switches its receiver shift register into the ring by directing ring input to the shift register, and transmits its own message from the transmit shift register. After completing transmission of its locally generated message, it then starts transmitting from the receiver shift register (RSR). When a message arrives at its des-tination NAU, its contents are copied in the local buffer at the NAU, while the message is still being relayed onwards. When a complete message arrives back at the source NAU, the source removes it from the ring by shunting the RSR out of the ring (see figure 2.5).
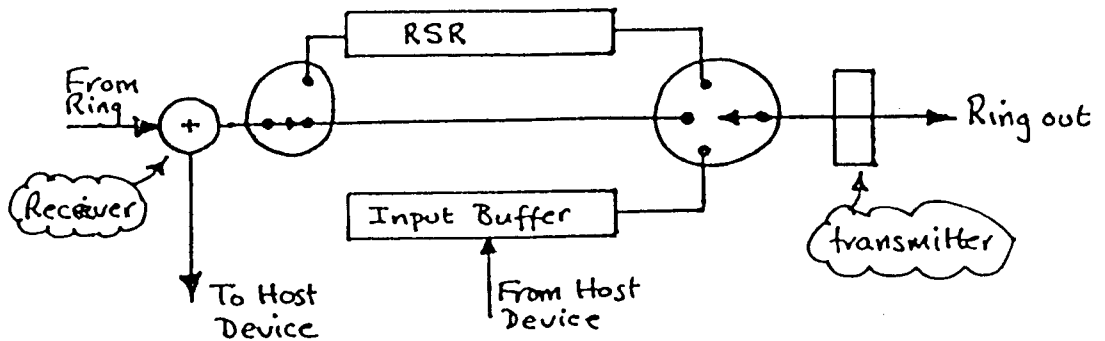
Fig. 2.5. A *static* register-insertion ring NAU

Another version of the register insertion ring. the *distributed loop computer network* *(DLCN)*, was proposed by Liu et al in 1976. The DLCN incorporates a variable length shift register at each NAU (figure 2.6). In contrast with the static insertion. DLCN is a *dynamic* insertion ring because an NAU in a DLCN can adjust the length of the inserted register to the amount necessary to accommodate incoming traffic during the transmission of a locally generated message. Another factor which distinguishes the dynamic insertion ring from the static version is that messages are removed from the ring by the destination node and do not need to travel back to the source node. In a random traffic distribution among the nodes on a DLCN, the average distance a message travels across the ring is half the size of the loop. Therefore, in the random case, the throughput of a DLCN ring can be twice its transmission rate. However, messages have to be delayed at each NAU for a sufficient time to receive and check the destination address field. Even so, the overall DLCN network performance has been shown to be superior to the other protocols [31]. The mode of operation of a DLCN is known as the *check and forward* mode to distinguish it from the *store and forward* mode of operation of a star network discussed in a previous section.
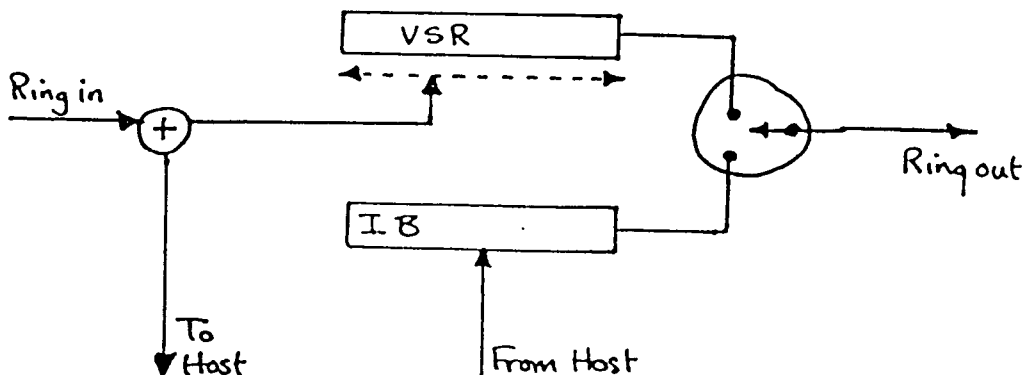
Fig. 2.6. A *dynamic* register-insertion ring NAU

The Computing Discipline of the Mathematics Faculty at the Open University. Milton Keynes. U.K.. where the research work reported on here was carried out. has a variant of the DLCN network. This network is called the *Multilink* network. It is manufactured by the Nine Tiles Computer Systems of Cambridge. England and marketed by Hawker Siddeley Dynamics Ltd. of U.K.

Multilink is a local area network primarily intended for connecting terminals into the central computing facilities within a building. It has totally distributed network control. and hence it does not require any special control node. Therefore. it does not require huge initial investment. except for laying cables. This makes it particularly suitable for small applications where only a few devices are to be networked. The per connection cost is also small. a stand alone network access unit. called a *station* in Multilink terminology. costs under £400. Cheaper still. and more convenient. is so-called *chip-set* version which fits inside a micro-computer and costs around £100.

Multilink uses the dynamic register-insertion mechanism of the DLCN network. It uses a removal by destination policy, although it avoids the address checking delay by performing an address check *on the fly*. When a node detects its own address as the destination address of a message being relayed, it terminates the on-going transmission by transmitting *null* characters for the duration of the period needed to receive the message. This results in a truncated message on the output of the destination node. This message is smaller than the smallest message allowed on the ring. All such messages are discarded by any downstream node which needs to buffer its input. In the Multlink ring network, therefore, these messages do not consume any useful network resources. The result is that the substantial per node delay incurred in case of a DLCN network is eliminated by using a special message removal technique.

In all other respects, a Multilink network is functionally equivalent to a DLCN network. The results of the research reported on here are applicable to both the Multilink and the DLCN network.

2.3. Discussion

This chapter has presented an overview of various local area networking approaches proposed in the literature. It is suggested that a ring network is technically more interesting than a bus or a star network. Among ring networking technologies, the dynamic register-insertion ring network, employing a removal by destination mechanism (as used in DLCN), has a superior performance than the token ring or the slotted ring mechanisms. The rest of this thesis is concerned solely with the DLCN network. Generalisation to other technologies is explicitly mentioned where appropriate.

3. LAN management

In the previous chapter, an overview of local area networking (LAN) technology was presented. This chapter discusses the issues of LAN management with particular emphasis on the design of a coherent user interface. A paper by Thomas and Yasin [54, Appendix A] presents the findings of our research in this field and is reviewed below. The rest of this thesis deals with the configuration management aspects of LAN management.

3.1. Introduction

Effective management of computer networks is essential for efficient utilization of network resources. In particular, management of local area networks is quite critical in order to avoid chaos and provide user satisfaction. The reason for this emphasis is that the typical user of a local area network uses computers only as a tool. He does not need to understand the operational peculiarities of a computer network. For such a person, all that matters is the use of a certain computing facility, in a reliable and easy manner. Therefore, an important aspect of LAN management is the provision of a simple yet powerful user interface. Various approaches to the research into management aspects of networks may be divided into two classes, namely: a) transparent and b) user-friendly.

The former approach, that is, a transparent network, is applicable to a class of networks called computer-computer networks. Here, the communication on the network is between two computer systems. This may be a communication between two operating system entities, e.g. to make a file transfer, or a communication between two application processes, the *inter-process communication (IPC)*. In a computer-computer

network, a user accesses the network via his local computer (see figure 3.1). The user has to learn the command language of the local operating system and the meaning and significance of signals returned by the local system. The essence of a transparent networking approach is to make the network resources available to a user without, or with minimal, change to his familiar command language. This may be accomplished by: i) intercepting the operating system calls to remote resources, accesssing them via the network and supplying them as local resources, an example is the *Newcastle connection* communications package [4], ii) providing a single coherent network operating system by rewriting the operating system on all networked computers, as in LOCUS [44], iii) concentrating the network communications functions and process scheduling functions into microprocessor front-ends [43], and iv) implementing a guest layer on top of local operating system as in the DESPERANTO system proposed by Mamrak in 1982 [37].
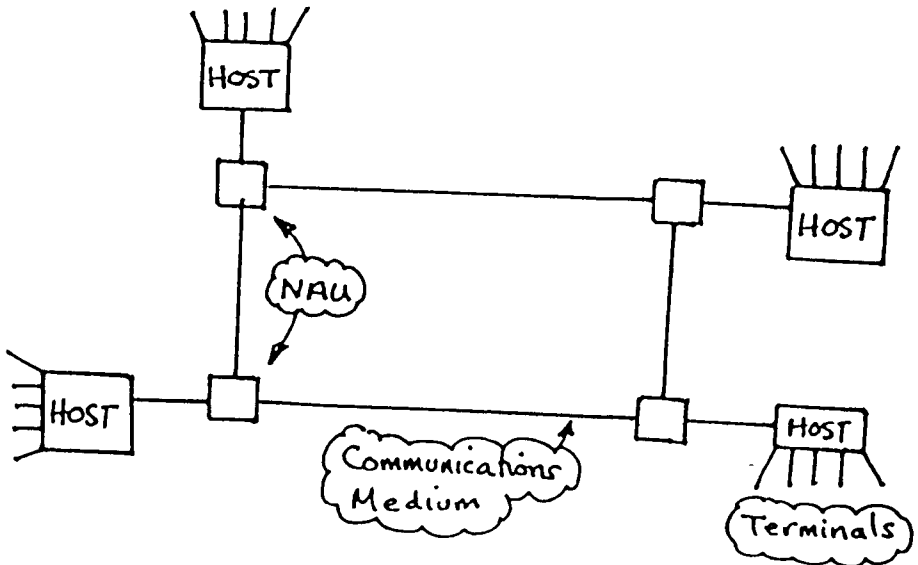
Fig. 3.1. A computer - computer network

The latter approach - of providing a user-friendly interface to the network - requires that the user be aware of the existence of the network. This approach is particularly suitable to peripheral networks (figure 3.2). The term *peripheral network* as used by MacCalla [35] refers to the inter-connection of computers and peripherals such as terminals, file-stores and printers. Here, the distinctive feature is that a user has to access the network in order to log on to a computer system. Such networks have low cost network access units and are primarily designed to replace the front-end multiplexers. Examples of peripheral networks are Clearway [34], and Multilink [15]. Both of these are ring type networks with low per-node cost and have fully distributed control, therefore, they do not require expensive control nodes. It is the management issues of the low cost peripheral networks with which the present chapter is concerned.
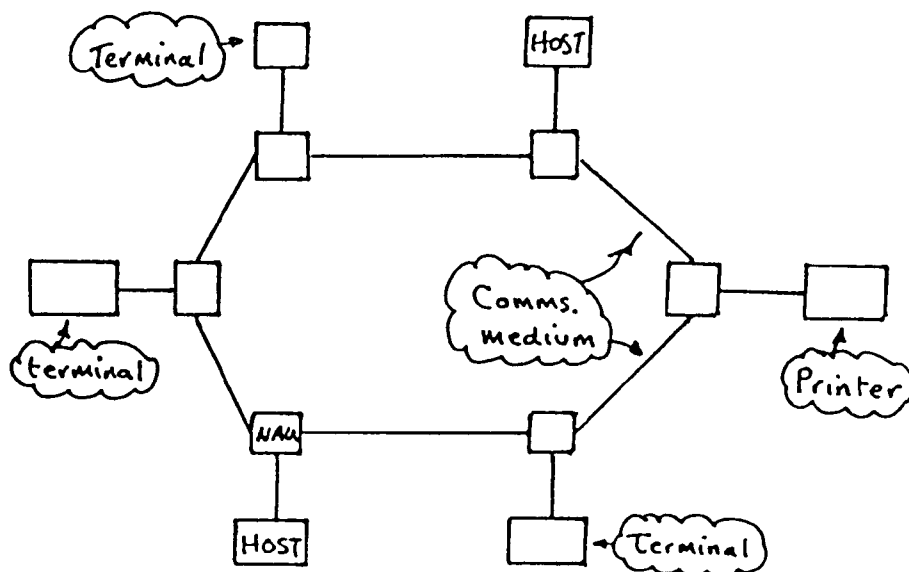
Fig. 3.2. A *peripheral network*

3.2. Literature survey

It is interesting to note that the user interface provided by the peripheral networks mimics that of an entirely different species, that of wide area networks (WANs). The similarity arises because WANs such as ARPANET and EURONET are designed to provide terminal type access to remote computers. This implies that we can draw upon the substantial research into user requirements on WANs presented by Rayner [47], Heap and Hall [21] and Berchanski [2] while considering the design of a user interface for peripheral networks. It is, however, to be noted that the solutions to issues raised by such research are very different in the case of peripheral networks from those proposed for WANs. The reasons being that for peripheral networks: (a) the use of network bandwith is cheap, (b) the network access unit (NAU) must be low cost and (c) modifications to the existing operating systems on the networked computers are undesirable.

In a survey of resource sharing services in computer communications networks published in 1977 [36], Mamrak defines the user access phase to consist of all steps taken after the user first makes contact with the network until the start of the required application program. Similarly, the system exit phase starts from the exit command to the application and continues until the communications channel is disconnected.

For a concatenated network, the connection between the user and the target computer system may be through several intermediaries. Rayner [47], defines a local interface as the first system that the user encounters in any network and suggests the use of a network access system (NAS) to make this the only visible user interface between the user and the target systems. To avoid the user having to give logically redundant information, an NAS could log on to the target system making all intermediate

connections as necessary. An NAS can also maintain logical connections to more than one target system and allow the user to switch between them.

In a detailed study of user logfiles on the EURONET-DIANE wide area network published in 1983 [9]. Cornelius makes several obvious but important observations about the user access mechanism. He notes that the messages received from a network are usually of a technical nature, containing little information for a naive user. He recommends that network messages should indicate only what needs to be done. Also, each message should be identified by date, time and the originating computer system. Finally, Cornelius points to a probable problem area: namely, the flow of error messages across a series of concatenated networks. Since the kinds of errors and the meaning of certain error messages may be peculiar to a particular network, such error messages may lose significance when travelling through other networks. This leads Cornelius to believe that standardization effort is required to avoid such confusion.

In an earlier study of user behaviour in a polytechnic environment, published in 1979 [5]. Bull and Huckle note the bewildering array of user interfaces provided by various operating systems. Their solution, to provide a network-wide editor function and a common set of frequently used commands to be interpreted at each host, requires expensive software development and, therefore, is not suitable for peripheral networks where cost is one of the main considerations.

A detailed breakdown of functional requirements of a LAN management facility is given by Wilbur in a recent paper [61]. published in 1986. Here, the notion of a management entity includes both the human managers and the set of management tools being employed to carry out certain management functions. The management functions are divided into five groups:

i)   operational management.

ii)   configuration management.

iii)   maintenance.

iv)   performance measurement and tuning.

and

v)   user administration.

The operational management functions include setting up and closing down a circuit, management of service access points (SAPs) and mapping of logical names to network addresses. The configuration management consists of the installation and update of network components and keeping a log of all changes in network configuration. Maintenance functions involve fault reporting, fault location and operational testing. Performance measurement refers to the collection of statistics about the usage of network resources and average response times. Tuning is the process of adjusting network parameters, and configuration, according to changing traffic patterns on the network. Finally, user administration functions relate to the identification and authorisation of user access and accounting for the use of network resources.

It is evident, from the above review of a representative selection of literature on network management, that the term management encompasses a wide variety of functions relating to network operation and maintenance. In the paper reviewed below, Thomas and Yasin [54] identify those management functions that can be integrated into the communications subnetwork of a low cost LAN. This paper is reproduced in Appendix A.

3.3. Review of *Enhanced management domain for low cost LANs* by Thomas and Yasin [54, Appendix A]

In appendix A, a selection of management features is presented which could usefully be enhanced in a low cost LAN. This is followed by a discussion on the cost effectiveness of centralized versus distributed implementation in a DLCN network (§ 2.2). The paper concludes by indicating directions for further research.

In the paper under review, the notion of a manager has been restricted to refer to a set of hardware and software devices used to provide the desired management functionality. Next, the concept of user- and peripheral- profiles is introduced, these profiles are kept by the network manager and contain frequently referred to information. A user profile can, for example, contain user identification and authentication codes for auto-log on to host computer systems. The peripheral profile may contain the set of initiation and termination command sequences for accessing a particular peripheral. Attention is particularly drawn to the complexities of the logging off procedures which must be performed in a specified manner. The peculiarity of this aspect of network access has also been mentioned by Olsen et al in a tutorial on data switching [39], but they do not suggest any strategy to handle this situation. The solution, proposed by Thomas and Yasin in appendix A, consists of providing a single log off command directed to the network manager, who will terminate all active sessions on the user's behalf and close the relevant data circuits. This feature is supplemented by a time-out mechanism operated by the network to recover from dead sessions and faulty circuits.

Provision of a file transfer facility is found to be very desirable. However, it is pointed out that, because of the low transmission speed of a peripheral network, indiscriminate use of this facility can seriously degrade the network service. Other

desirable and feasible features identified in this paper include: a) a help facility, b) support of simultaneous multiple sessions, c) electronic mail and d) collection of statistics.

It is then shown in the paper, that, for register-insertion ring networks employing removal by destination such as DLCN, some of the required management functionality needs to be distributed among all network access units (NAUs). The rest of the functionality can be provided as an optional extra on one or more machines acting as co-operating managers. In order to keep the cost of a NAU to a minimum, a bare minimum of functionality is identified which must be provided in each NAU. This functionality includes:

i)   the detection of network attention commands,

ii)   multiplexing of simultaneous sessions,

iii) sending statistical information to the central manager,

iv) operating a time-out mechanism on inactive sessions,

and

v)   acting upon the log-out command.

The central manager provides the following functions:

i)   the network's user interface including various profiles,

ii)   make a file transfer,

iii) manage electronic mail,

and

iv) collect network usage statistics.

An experimental set-up is also described in appendix A, which was used to demonstrate the feasibility of the distributed approach presented above.

Three directions for further research have been indicated. These include: (a) implementation of the proposed facilities, (b) standardization of the the NAU - central manager communications protocol and (c) studying the use of statistics gathered by the network manager to improve the performance of the DLCN network.

3.4. Discussion

Network management is a broad term which has been widely used to refer to the operations carried out by the human managers as well as those by the automated management tools. The literature reviewed in this chapter has been selected to represent those aspects which are relevant to the management af low cost LANs.

The paper by Thomas and Yasin [54, Appendix A], reviewed in 3.2, is a significant contribution in this field in that a) it identifies those management functions that can be automatically provided by a low cost network and b) it discusses implementation specific issues for register-insertion networks employing removal by destination.

As suggested in appendix A, a direction of further research is to investigate ways in which the network usage statistics, gathered by the network manager, can be used to improve the performance of the underlying network. Research in this direction was continued and resulted in a novel technique for improving DLCN performance by dynamic reconfiguration. The findings of this research have been presented in [57] and [56], these papers are reproduced in appendices B and C and are reviewed in the following chapters.

4. Optimization of Ring Configuration

An important function of a network management facility, discussed in the previous chapter, is to gather various statistics about the network which can be used to improve the network performance, and to plan future expansion of the network. This chapter reviews the paper by Thomas and Yasin [57, Appendix B] where it is proposed that, due to a unique characteristic of the Distributed Loop Computer Network (DLCN), the collected statistics can be used to configure a network in an optimum manner and thereby reduce the average message transfer delay which is an important performance measure.

Section 1 of this chapter introduces the reader to the importance of selecting the proper configuration of nodes on a DLCN network. In section 2, a new parameter which defines a network configuration is given. Real world situations where a performance improvement by reconfiguration, as suggested in [57], can be economically affected are described in section 3. Section 4 relates to other published work and sections 5 and 6 summarise the findings of the research reported in the paper. Concluding remarks are drawn in the final section of this chapter.

4.1. Introduction

The distributed loop computer network is a typical example of a register-insertion ring network employing the removal by destination protocol (§ 2.5). The DLCN exhibits a useful characteristic, namely, that messages travel only part of the loop. A message is introduced into the network by a source node and travels along the loop until it reaches its destination, where it is removed from the network by the destination node. In contrast, a removal by source mechanism requires that each message travels the

whole of the loop. The former method, therefore, results in saving of the otherwise wasted network bandwidth.

Consider the six node regitser-insertion (R-I) ring shown in figure 4.1 employing removal by destination mechanism. Let us follow a message originating at node 2 and destined for node 4. When a single message is observed in this way it is referred to as the tagged message. This message will first travel from node 2 to node 3, where a check on destination address field in the message header will indicate that it is to be relayed on to the next downstream node (node 4). Since node 4 is the destination node for the tagged message, its content will be copied into the internal memory of node 4, and the message will be removed from the network (i.e. not relayed on to the next node). Thus, all other nodes on the network (nodes 5,6 and 1) will be un-affected by the transmission of this message.
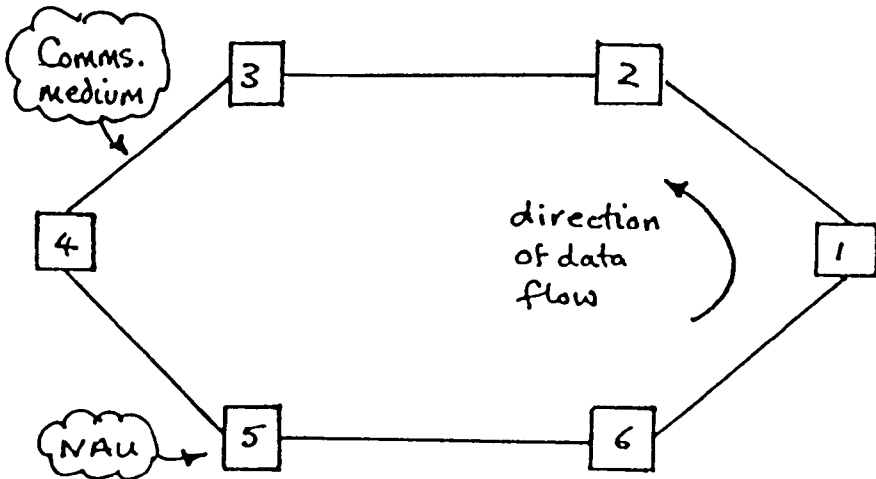
Fig. 4.1. A six node register-insertion ring

In contrast, consider a similar situation arising in a R-I ring employing the removal by source protocol. Here, the tagged message will be relayed onwards by the destination node (node 4) to the next down-stream node (node 5) and travel the rest of the loop (nodes 6 and 1) to arrive back at the source node (node 2), whereupon node 2 will remove it from the network. In this case, the tagged message consumes processing resources at all network nodes, thereby contributing to queueing delays at additional nodes (nodes 5, 6, 1, 2) and causing a higher loading of the network.

The additional load on a DLCN network, due to an extra message being introduced depends, among other parameters, on the length of the data path between the source and destination nodes. A measure of the length of the data path between a source node and a destination node is the number of intermediate nodes encountered when travelling along the loop from the source node to the destination node in the direction of flow of data on the loop. If it is assumed that the delay due to transmission along the physical path joining any two adjacent nodes is small when compared with the magnitude of delay caused by the onward transmission in a node, then the significant measure is the number of intermediate nodes rather than the identity of those nodes. This is a reasonable assumption for LANs where inter-nodal distances are quite small, and all nodes are functionally identical and almost uniformly spread over the length of the network.

By minimising the number of nodes between a source-destination node pair, the effective load on the network, due to the traffic between this node pair, can, therefore, be decreased. In [57, Appendix B] a measure of this effect is introduced as the (imaginary) work done by the network in transporting the data traffic from a source node i to a destination node j, defined as: the product of the volume of traffic from node i to node j, V_{ij}, and the corresponding length of data path, l_{ij}. In the paper of Appendix B it is shown that the product sum

$$\sum_{i=1}^{i=n} \sum_{j=1}^{j=n} l_{ij} * V_{ij} \qquad (4.1)$$

is an appropriate objective function which can be used to compare the performance of different configurations of a network of n nodes for a given traffic pattern among its nodes. If the configuration of a network is variable, as in the case of the dynamically reconfigurable network proposed by Lee in 1979 [25, 55], the above objective function can also be used to find the optimum relative node positioning on the network.

In what follows the term "node" is used to refer to the network access unit (NAU) as well as to the "NAU – attached device" pair.

4.2. A Definition

In order to introduce the network configuration as a variable in the calculation of the objective function (4.1) [57, Appendix B] introduces a new notation d_{ij}, for a pair of nodes i and j, as follows:

(i)   First the loop network is formed into a chain of nodes by breaking the input link at an arbitrary node, called the reference node and labelled as node 1.

(ii)   Then, travelling from node 1, in the direction of flow of data on the network, define d_{ij} as

$$d_{ij} = \begin{cases} 1 & \textit{if node } j \textit{ lies before node } i \\ 0 & \textit{otherwise} \end{cases} \qquad (4.2)$$

Thus, for the reference node,

$$d_{i1} = 1, \text{ and } d_{1j} = 0, \text{ for } all \; i,j = 2, 3, ..., n.$$ (4.3)

The usefulness of this notation can be appreciated by comparing the following two representative extracts from earlier literature along with their respective restatement using d_{ij}.

Extract from Liu et al 1977 [28,§ 3]

For a network of N nodes, given P_{ij}, the portion of traffic originating at node i that is destined for node j, and a Poisson data source with parameters λ_i and $\dfrac{1}{\mu_i}$.... Let R_{ik} denote the average character rate originated from the ith node and passing through the kth node (i.e., to be served by the kth channel) then we have

$$
R_{ik} =
\begin{cases}
\dfrac{\lambda_i}{\mu_i} \sum\limits_{j=1}^{i-1} P_{ij} + \dfrac{\lambda_i}{\mu_i} \sum\limits_{j=k+1}^{N} P_{ij} & 1 < i < k \neq N \\[2ex]
\dfrac{\lambda_i}{\mu_i} \sum\limits_{j=1}^{i-1} P_{ij} & 1 < i < k = N \\[2ex]
\dfrac{\lambda_i}{\mu_i} \sum\limits_{j=k+1}^{N} P_{ij} & 1 = i < k \neq N \\[2ex]
\dfrac{\lambda_i}{\mu_i} \sum\limits_{j=k+1}^{i-1} P_{ij} & k+1 < i \leqslant N \\[2ex]
\dfrac{\lambda_i}{\mu_i} & i = k \\[2ex]
0 & otherwise
\end{cases}
$$

Expression for R_{ik} rewritten using d_{ij}

$$R_{ik} = \frac{\lambda_i}{\mu_i} \sum_{j=1}^{N} P_{ij} \; (\; d_{jk} \; d_{ki} + d_{ij} \; d_{ki} + d_{ik} \; d_{ij} \; d_{jk} \;)$$

Extract from Bux and Schlatter [7, § III.B.2]

{For a network of N nodes...}. Given the probabilities p_{ij} that a frame originating at node i is destined to node j, the probability e_{ik} that a frame originating from station i passes through station k is given by

$$e_{ik} = \sum_{j \in I(i,k)} p_{ij}$$

$$I(i,k) = \left\{ \begin{array}{ll} S - \{x \mid i < x \leq k\} & i < k \\ \{x \mid k < x \leq i\} & i \geq k \end{array} \right.$$

$$S = \{1,2,.....N\}$$

Expression for e_{ik} rewritten using d_{ij}

$$\text{---} \quad e_{ik} = \sum_{j=1}^{s} p_{ij} ((d_{\overline{ij}} + d_{jk}) d_{ki} + d_{ik} d_{jk} d_{ij})$$

It is worth noting that, in the literature, some analyses have been restricted to discuss special cases such as a symmetric traffic [28], or a balanced loop [58], whose performance does not depend upon the configuration of the nodes on the loop. Therefore, such analyses do not require the elaborate handling of relative positioning of nodes on the ring. In contrast, the present study is aimed to draw benefit from the fact that the actual data traffic on any communications network is usually far from symmetric or balanced. Thus, the present discussion considers networks with arbitrary traffic patterns among its nodes, and has to tackle the problem of incorporating the network

configuration not only in the derivation of the network performance calculations but in such a manner as to facilitate an algorithmic approach to optimization by considering the configuration to be one of the variables of the problem.

4.3. Reconfigurable Networks

To draw a physical parallel to the mathematical treatment of the network configuration as a variable of the optimisation problem, Thomas and Yasin [57] cite the following three network architectures as examples.

(i)   The most obvious parallel problem arises while planning the initial layout of a DLCN-type network, or a network reconfiguration involving physical movement of nodes from one location to another.

(ii) Star-ring architectures, proposed by Saltzer et al in 1979 [49] and Dixon in 1982 [12] as a means to facilitate node isolation for token ring network, can be modified to provide limited reconfigurability of a register-insertion network.

(iii) The dynamically reconfigurable register-insertion (DRR) network was proposed by Lee in 1979 [25] to increase network reliability by dynamically reconfiguring the network in case of link or node failures. This network architecture provides the facility for dynamic reconfiguration by a central station without any significant increase in the complexity of the network access units. A DRR network consists of multiple bi-directional links between nodes and incorporates a two-entry routing table in each node. The entries show which of the attached links are the currently selected in-link and out-link. At any instance of normal operation,

the set of in-link and out-link pairs forms a logical ring. Thus, the physical network is a general multiply connected network but the logical network is a loop network.

4.4. Related Work

A related problem is the flow assignment problem which determines the routing decisions in order to optimize a certain objective function for a given configuration. A recent paper by Tcha and Maruyama [53] discusses a similar problem of selecting the primary paths for the IBM System Network Architecture [16] amongst the available multiple paths. The analytical formulae developed by Tcha and Maruyama for solving the primary paths problem are not applicable to the configuration optimization, being researched in this thesis, for two reasons: (i) the resulting SNA network is not necessarily a ring and (ii) each node pair in SNA has a number of pre-determined data paths. For the analysis of the configuration optimisation problem for a register-insertion ring network, the first condition is unacceptable and additional constraints must be developed to restrict the resultant configuration to a ring structure. More importantly, condition (ii) is not valid for ring networks because selection of a data path for one node pair restricts the choice of available paths for other node pairs. Therefore, independent sets of available data paths for each node pair can not be formed.

4.5. Formulation of the Optimisation problem

In appendix B, the optimisation problem is simplified (although the simplification can be relaxed) by assuming a fully connected physical network, so that any two nodes can be made adjacent on the logical ring by appropriate selection of the in-link and out-link pairs. A discussion of the less-than-fully connected network appears in a

follow-up paper [56] , which is discussed in the next chapter.

As stated earlier, an optimum configuration is achieved by minimising the objective function (4.1). This is clearly an integer programming problem. For simplicity of notation and mathematical treatment, a matrix notation is used. A set of constraints is developed which is shown, by examples, to be both necessary and sufficient.

An alternative set theoretic approach is also presented. The method of solution described in the paper gives satisfactory solutions for a variety of simple problems. In certain cases, the method was found to be trapped by local minima, hence, resulting in less than optimal configurations. Elaborate book-keeping is required to find the best of several minima in a problem, but this results in the loss of the obvious simplicity and efficiency of this approach.

4.6. Simulation studies

In order to verify this intuitive approach to optimisation and the analytical results thus obtained, a simulation package for the dynamically reconfigurable network was developed in the SIMULA language on the Open University DEC-system 20 computer system. The simulation was used to compare various configurations for the same problem. The best and worst case configurations obtained by the linear programming technique are compared, in the paper, for two networks, one consisting of seven nodes and the other of eight nodes. The performance measure used for comparison is the mean transfer delay per message, which is plotted to see the behaviour of these networks under different loads. The simulation results clearly show that the optimum configuration produced by minimising the objective function is far superior to that obtained by maximising the objective function, (the worst-case configuration).

4.7. Conclusions

The paper reviewed above, investigates the sensitivity of a DLCN network's performance to its configuration. The results presented in the paper, and confirmed by simulation, clearly show that the performance of a register-insertion ring network, employing the removal by destination mechanism, depends on the relative location of its nodes on the network. It is also shown that the integer programming formulation, developed in the paper, can be used to find the optimum relative node positioning on an insertion network. Simulation studies indicate not only that the optimum configuration is far superior to the worst case configuration, but also that the former is more stable under higher loads, thereby increasing available network throughput.

5. Optimisation Of The Less Than Fully Connected Network

The previous chapter considered the problem of optimum node location in the simplified case of a fully connected physical network, and reviewed a paper by Thomas and Yasin [57] reported the findings on this subject. This chapter reviews a subsequent paper [56, Appendix C], which discusses the formation of the optimum logical ring for a less-than-fully connected dynamically reconfigurable register-insertion (DRR) network.

5.1. Introduction

In the previous chapter the benefits of the proposed optimisation of the logical ring configuration for the DRR network which employs the removal by destination mechanism were demonstrated. A method of solution using integer programming techniques was also presented, though for the simplifying case of a fully connected network.

A fully connected network is the least attractive choice for a DRR architecture for economic reasons. A network of N nodes would require $\frac{1}{2}N(N-1)$ links to be fully connected, i.e. 4500 links for a network of 100 nodes! In contrast, a single loop (i.e. DLCN) requires just N links (i.e. 100 links for a 100-node network). A more practical alternative for DRR architecture would have a reasonable number of links - higher than the minimum required to form a single loop in order to allow successful reconfiguration in the event of a link failure, but less than the fully connected case for economic reasons.

5.2. Method of Solution

In [56, Appendix C], the connectivity of a DRR network is decomposed into three components:

a)   a basic structure of evenly distributed links such that each node has the same number of links attached to it forming a multi-connected loop,

b)   those links which are in addition to the links included in (a) above,

c)   additional links which are needed to complete the structure of (a) above.

The multiplicity of links in the structure of a multiconnected loop is required, in Appendix C, to be symmetric such that if node i has a link to node $i+h$ then each node has a link to the node at distance h. The symmetric multiplicity is tackled by forming two disjoint sets of node pairs.

(A)   those pairs of nodes which are joined by a direct physical link,

(B)   those which do not have a direct physical link between them.

A pair of constraints is derived to model each set. Examples are given which show that the contraints derived in the paper result in logical ring formations consisting only of existing physical links.

The above discussion of symmetric multiplicity readily yields to the inclusion of extra links and link or node failures. To model an extra link joining a previously disjoint node-pair, the identified node pair is moved from set (B) to set (A). A link failure requires the converse operation. A node failure, making the node unavailable to any other node, can be modelled by moving all node-pairs containing the failing node to set

(a) A DRR Network

(b)
Underlying
Symmetric loop with
multiplicity = 2
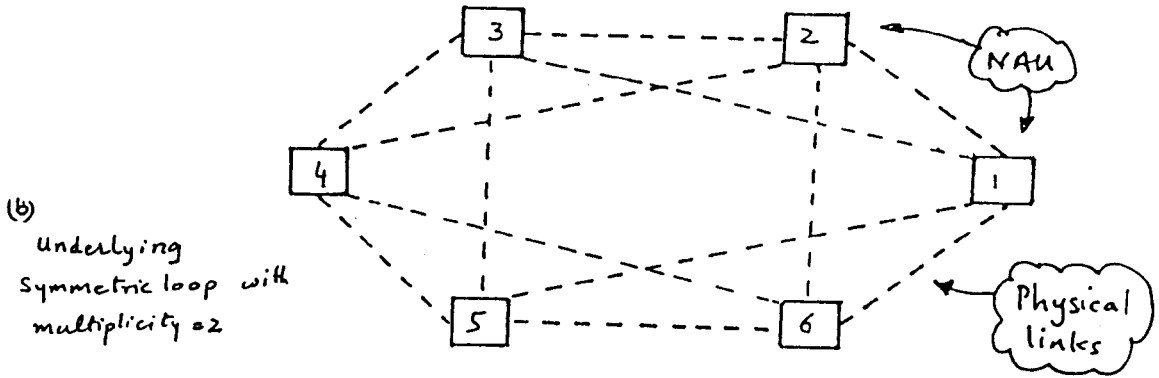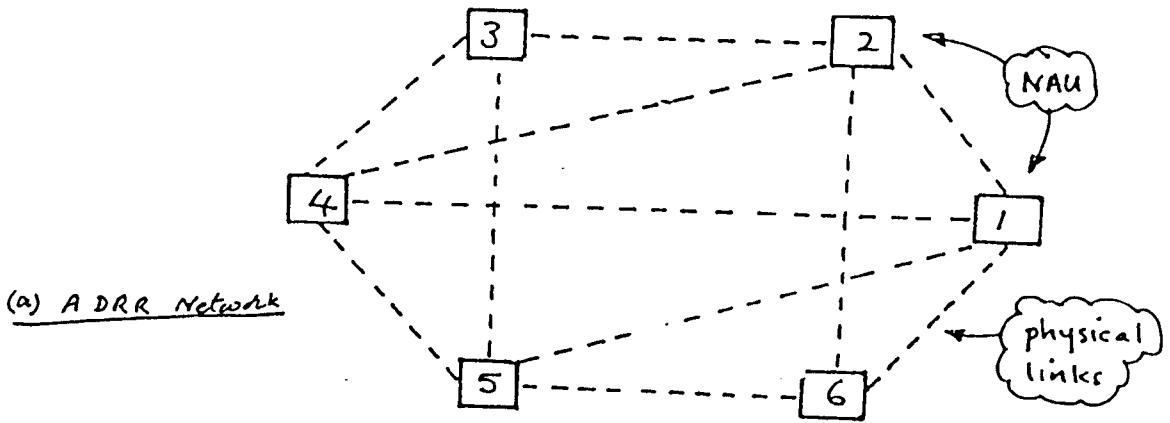
Extra link : joining node 1 to node 4

Missing links: from node 4 to node 6, from node 1 to node 3

Fig. 5.1. Decomposition of a DRR network

(B).

5.3. Conclusions

The formulation of the optimisation problem, given in the paper of Appendix C, can be used to form the optimum ring configuration for a dynamically reconfigurable register-insertion (DRR) network. This formulation can also cope with node and link failures as well as the addition of extra links. It has been concluded previously that such optimisation can result in considerable improvement in the network performance. In his paper proposing the DRR architecture [25], Lee suggests that a control node would be required to execute an algorithm to form an arbitrary logical ring, if possible, and reconfigure the network upon initial set-up of the network and upon detection of node or link failures. We conclude that for any such reconfiguration the optimum ring formation of appendix B and appendix C should be used, rather than, as suggested by Lee, an algorithm to form an arbitrary ring. It is shown by worked examples, and again confirmed by simulation, that the resultant network exhibits significant improvement in performance, is stable under higher loads, and offers higher available throughput. We also suggest, in addition to the events leading to network reconfiguration cited by Lee and summarised above, that the control node should attempt a reconfiguration if it detects a significant shift in the pattern of traffic on the network rendering the previously optimum configuration to be sub-optimal. Further research is required to identify patterns in traffic and to develop criteria to decide when a significant shift in the traffic pattern has occurred warranting a network reconfiguration.

One approach is to divide the traffic into sub-groups and attempt a reconfiguration either when a group closes down, or when a new group starts communicating. Such sub-division could be made on the basis of closed user groups which communicate mainly among themselves.

An alternative sub-division is obtained by decomposing the network traffic into a number of super-imposed service related communications. Therefore, a new service coming on stream, or the closing down of an active service would result in known shifts in the traffic pattern which could indicate the need of a network reconfiguration.

6. Performance Analysis of the Register-insertion Network

This chapter is concerned with the performance analysis of register-insertion (R-I) ring network. Here we investigate models of R-I rings which can be used in the analysis of dynamically reconfigurable R-I networks given in chapter 7. The operation of such networks is described in section 1. A survey of literature on performance analysis is reported in section 2 where four main queueing models of the R-I ring network are described and compared. The discussion in section 3 concludes that, of the four models considered, the Bux and Schlatter model is the most appropriate for the subsequent analysis of the DRR because of its accuracy, simplicity and treatment of asymmetric traffic conditions.

6.1. Introduction

A ring type local area computer network essentially consists of a number of nodes which are linked together in the form of a loop by means of a high speed communications channel (fig 6.1). The nodes consist of network access units (NAUs) and such attached devices as computers, work-stations, printers and other peripherals. A device wishing to communicate with another device on the network passes its messages onto its NAU for transmission on the network. Each message is tagged with a fixed-length header containing, amongst other information, the network identification of the NAU attached to desired destination device (referred to as the destination address). A message travels along the loop through successive NAUs until it is removed from the network.

Two removal policies proposed for the register-insertion network are, (1) removal by source and (2) removal by destination. The former method requires that a message
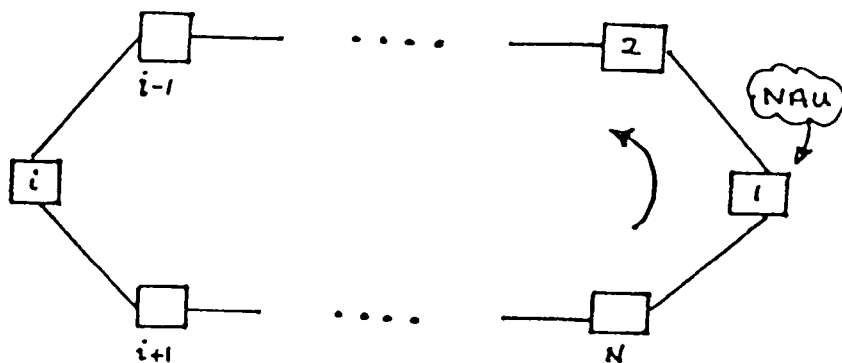
Fig. 6.1. A ring network

travel along the whole loop back to its source NAU to be removed [18]. The latter technique [48] recognizes the fact that the information contained in a message stops being useful after it has reached its destination. Removal of the message from the network by the destiantion NAU requires that messages travel only part of the loop - from the source NAU to the destination NAU. This results in a saving of the otherwise wasted network bandwidth. In the special case of symmetric traffic, in which each NAU transmits equally to all others, messages travel only half the loop, on average, thereby effectively doubling the available network throughput. There is, however, an associated penalty as each node delays incoming messages in order to check the destination address field. Various performance studies, both by analysis [7, 31, 38, 22, 20, 51, 29, 33] and by simulation [19, 8, 32, 48] show that the resultant performance of a register-insertion network with destination removal, for a balanced network, is superior to that of other local networks. The research reported on here indicates that this particular mechanism can also be used to advantage in case of un-balanced traffic conditions [55] as well as giving rise to an interesting optimization problem [56, 57].

Propagation errors and link and node failures can lead to corrupted messages. In such cases special removal mechanisms have to be invoked [41, 32]. Given that local area networks tend to offer a high degree of reliability the present study concentrates on the behaviour of networks during fault-free operation. During this time every message transmitted on the network will eventually reach its destination.

6.2. Literature Survey

The first attempt to analyse the DLCN was made by Liu et al in 1977 [28]. A queueing model is used to obtain formulae for average channel utilisation and average message transfer delay. They view a DLCN network as a cyclic network of queues. Each NAU is represented by a server with two input queues "t" and "r", as shown in figure 6.2 . The host device attached to a NAU is the source of all messages into the transmit or t-queue, whereas the messages arriving from the ring which are to be relayed onwards form the relay or r-queue.
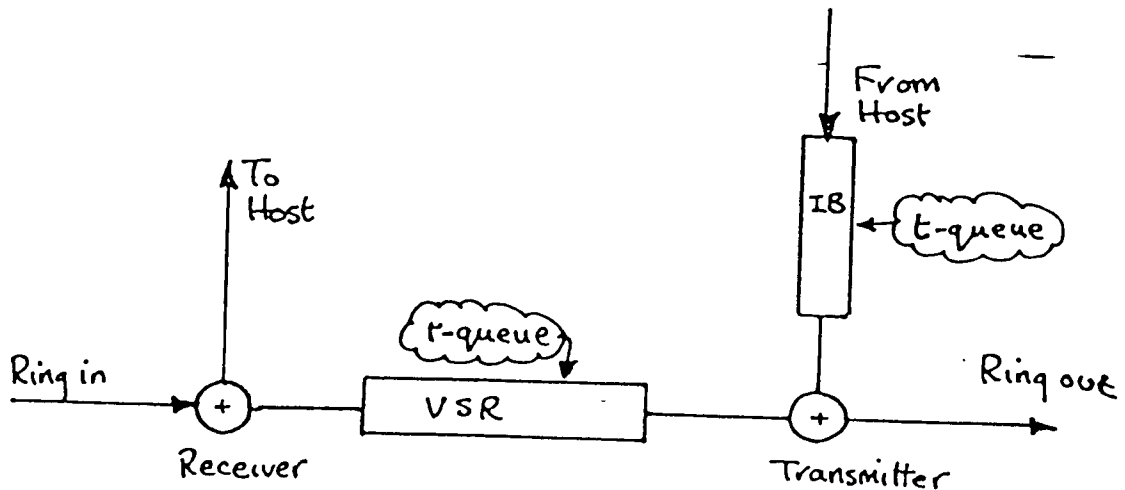
Fig. 6.2. Liu's queueing model of an NAU

Liu et al highlight the fact that the DLCN is a unique queueing system because the finite size of the buffer memory provided in the NAU causes the the priority to dynamically alternate between the r- and t-queues. Later on in their paper, however, an infinite buffer is assumed to simplify the calculation of various queueing delays by assigning a permanent high priority to the t-queue. As the transmit-queue, or the t-queue, consists of the messages generated by the local host, the above scheme is also known as the *station priority* scheme as opposed to the *ring priority* scheme where a permanent high priority is assigned to the messages waiting to be served in the relay-queue or the r-queue. Another important feature of the DLCN which has been identified in this paper is that, in contrast with the conventional *store and forward* networks where the whole message is stored at intermediate nodes before being relayed onwards, the DLCN requires that the header alone needs to be buffered at the intermediate nodes. This is known as a *check and forward* system. Hence, in a DLCN network, various sections of the same message may be simultaneously served by different NAUs, resulting in lower transfer delays. A similar approach for computer networks in general, called virtual cut through, was proposed and analysed by Kermani and Kleinrock in 1979 [23] , and is discussed later.

A comparison with simulation is also given in [28] which shows that the analytical results are consistently conservative, and that the disagreement between the two sets of results grows with increasing traffic loads. The authors cite their use of three assumptions in deriving the analytical results as possible causes for this discrepancy, which are critically reviewed below:

(i)   use of an independence assumption,

(ii)   assuming independent Poisson arrivals to the relay queue,

and

(iii) assuming infinite buffer size in calculating queueing delays.

The independence assumption. first described by Kleinrock in his classical work on Queueing Theory [24] . requires that the random processes for generating inter-arrival times and selecting message lengths are mutually independent. This results in messages reselecting their lengths randomly at intermediate NAUs. The independece assumption has been used by several other analysts in studying computer networks [28, 58, 23]. However. as pointed out by Kermani and Kleinrock [23]. the independence assumption is only acceptable as long as the network being studied does not contain long chains of NAUs with no interfering traffic. In a study of register-insertion rings published in 1983 [7]. Bux and Schlatter argue that when applied to register-insertion networks the independence assumption leads to results of very limited accuracy.

Assuming independent Poisson process for arrivals from the ring (r-queue) facilitates a mathematical treatment by isolating each NAU from the rest of the network. Obviously the ring input at a NAU is dependant on the external arrivals at the other NAUs and assuming otherwise introduces errors in the results. This is illustrated by an extreme example due to Bux & Schlatter [7] who look at the case of data distribution where one node is the source of all data and the only queueing delay occurs at the t-queue of the source NAU. Assuming arrivals to the r-queue to be independent of arrivals to the t-queue. however. introduces queueing delays in the r-queue at all intermediate NAUs. thus over-estimating the message transfer delay.

The assumption (iii). i.e. assuming an infinite buffer size (or working with MDLCN) has been shown in subsequent analyses by Thomasian and Kanakia in 1979 [58] . Liu et al in 1982 [31] and Bux and Schlatter in 1983 [7] to have no effect on the average transfer time of a message. a measure of particular relevance to the study of network

performance.

In an accompanying paper [29] Liu et al use the analytic results presented in [28] to verify that the performance of a register-insertion network employing removal by destination mechanism is superior to the other two major ring technologies, namely the slotted ring due to Pierce [43] and the token ring proposed by Farmer and Newhall [13].

In 1979, Thomasian and Kanakia [58] used queueing analysis to compare the performance of the two competing access control mechanisms for the register-insertion (R-I) ring, the asynchronous time division multiplexing (ATDM) and the modified distributed loop control network (MDLCN). The ATDM assigns a fixed high priority to the relay-queue (also called *ring priority*), whereas the MDLCN assigns a permanent high priority to the transmit-queue (also known as *station priority*). The authors define the notion of a balanced loop as the loop where all NAUs have same transmission pattern, and prove that, for such a system, the mean transmission delay for ATDM is the same as that for MDLCN. The conservation law of mean waiting times [24] is subsequently used to show that the mean end-to-end transmission delay is invariant for any queueing discipline using message priorities. This is an important result, also independently arrived at by Liu et al in 1982 [31] and by Bux and Schlatter in 1983 [7] using substantially different queueing models of the R-I ring. This invariance property has been used to simplify the discussion in this thesis by referring only to one access control mecanism, namely MDLCN, without any loss of generality.

Three simplifying assumptions render the actual formulae, derived by Thomasian and Kanakia, of limited application. First, the independence assumption has been used to simplify the analysis. Second, the arrival rate from the ring has been calculated for the

balanced loop alone. Finally, the underlying network has been assumed to be a *store-and-forward* type rather than the *check-and-forward* system proposed by Liu et al [31]. Thomasian and Kanakia do suggest that the application of the analysis of the *virtual cut-through* approach to the R-1 network should be studied but do not attempt it themselves.

The virtual cut-through system was proposed in 1979 by Kermani and Kleinrock [23] as an efficient alternative to message switching, also referred to as a *store-and-forward* network. In a virtual cut-through system, a whole message is buffered at an intermediate node only if its outgoing channel is busy, otherwise it is transmitted on the outgoing channel after the header section has been received for address checking. Therefore, the un-necessary delay introduced by a message switching network, which stores a complete message before relaying it, even in front of an idle channel, is avoided. The accompanying analysis shows that, in general, the network delay for a cut-through system is less than the message switching delay. The analysis assumes a balanced network, i.e. the utilisation factor for all network channels is the same, and, like former analyses, utilises the independence assumption.

It should be noted that the virtual cut-through system described above differs from the check-and-forward network proposed by Liu et al [31] in one significant aspect. In the Kermani and Kleinrock system, a cut-through does not occur if the outgoing channel becomes free after the reception of the header is completed, in which case, the whole message is buffered before being relayed to the next down-stream NAU. In contrast, the check-and-forward system initiates transmission of a message as soon as the outgoing channel becomes free. Thus, the latter system is the more efficient of the two.

In 1982, Hilal and Liu took a new approach to the analysis of the DLCN in a paper [22] presented at the Computer Networking Conference. The authors note that their earlier analysis [28] of the register-insertion ring significantly over-estimates network delays for high loads and for large networks. Therefore, a new model is developed and analysed, which is shown to be more accurate. Contrary to their earlier emphasis on the dynamically alternating priority scheme employed by the DLCN, the authors argue that the queueing discipline, i.e. whether a ring priority or a station priority scheme is employed, does not affect the average waiting times, therefore, the new model (see figure 6.3) merges the t- and r-queues into one input or i-queue. Next, the statistical dependance, of message inter-arrival times and message lengths, is taken into account by modelling an N-node network as N-queues in tandem. This model is then analysed in two steps. First, a simpler network is considered which does not have any external traffic and then the effect of symmetrically distributed traffic is considered. The resulting formulae are in good agreement with simulation results as shown by Liu et al in an accompanying paper [31].

There are three points to be made about this paper compared to the earlier analysis by the same authors [28]. First, as discussed earlier in detail, the priority mechanism, by which the transmitter of an NAU serves the two input queues, is assumed not to have any effect on the message transfer time across the network averaged over all NAUs of the network. Second, use of the independence assumption for arrivals to the r-queue appears to be the reason for over-estimation of network delays in [28], therefore, the results of analyses which employ this assumption are of little practical value. Finally, analysis of the new model, of N-queues in tandem, has been performed only for networks with symmetrically distributed traffic. This revised model does not yield to simple analysis for asymmetric traffic and, therefore, is not suitable for studying the effects of varying traffic patterns on the performance of register-insertion networks.
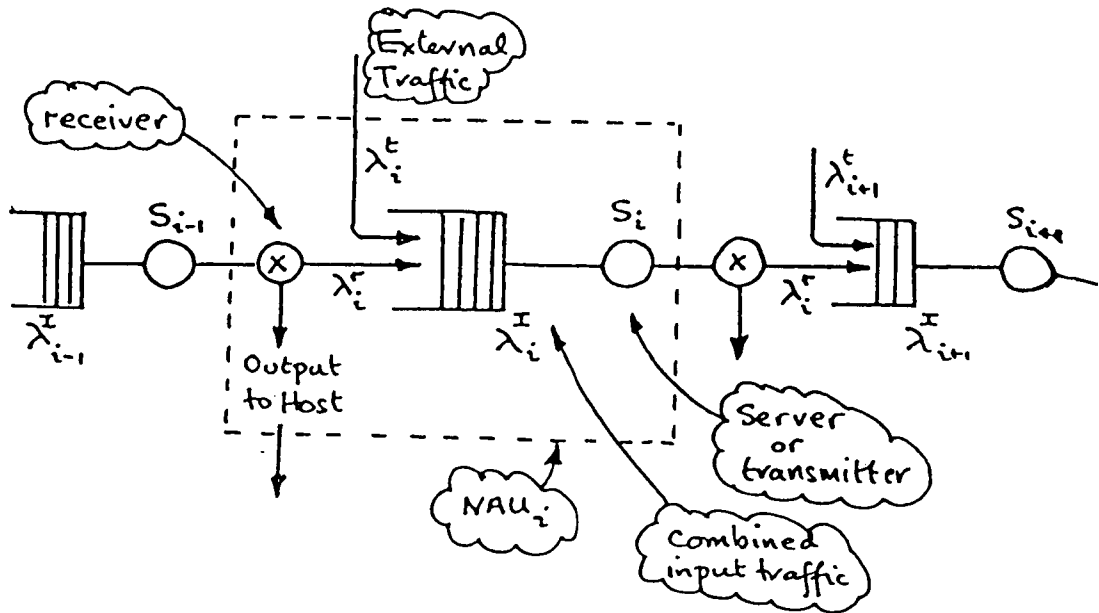
Fig. 6.3. Liu's revised model of an NAU

A new analytic approach to the performance analysis of the register-insertion (R-I) network was proposed by Bux and Schlatter in 1983 [7]. As discussed earlier, they argue that the independence assumption, used by Liu et al [28] and Thomasian and Kanakia [58] in their analysis of the R-I ring, and by Kermani and Kleinrock [23] in their analysis of the virtual cut-through system, is not applicable to the R-I ring network.

In the new model of a NAU proposed by Bux and Schlatter, see figure 6.4, a message arrives at the latency box of, say, NAU_i at the start of its transmission over arc $(i-1)$ to i, and its transmission times over various arcs are equal. The ring input λ_i^r to the i'th NAU is, then, derived as a function of the inputs to NAU_{i-1}. The resultant analysis is claimed to yield exact results for the special case where one node is the source of all data (data distribution) or where all other nodes on the network send
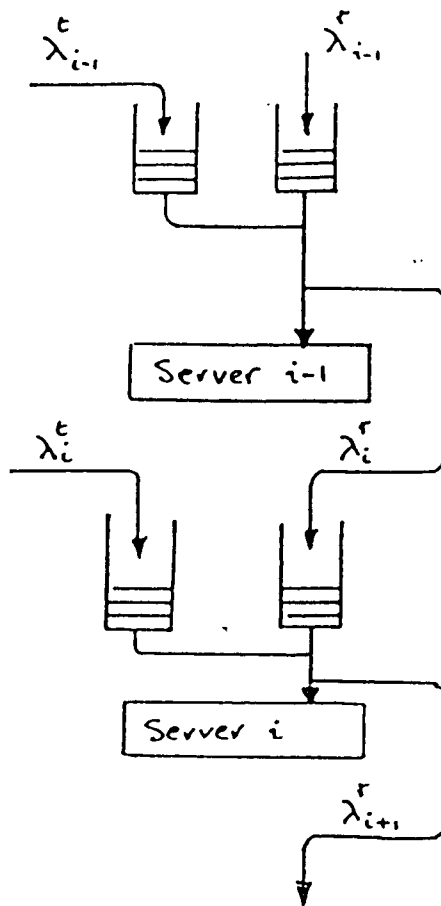
Fig. 6.4. Bux and Schlatter Model of an NAU

data to one node (data collection). It is also shown that mean transfer delay for ring-priority (ATDM) and station-priority (MDLCN) is the same. The authors conclude that prioritizing ring access does not improve the performance of the R-l ring. They also confirm that the achievable ring throughput can exceed the transmission speed of the network for special traffic patterns.

6.3. Discussion

The analytical model described by Bux and Schlatter closely models the actual physical process. A comparison with simulation given in [7] shows that the results obtained are quite accurate. In a recent performance study of multi-connected loop networks [46], Raghavendra and Silvester also found that the results of their analysis using the Bux and Schlatter model were in close agreement with the results they obtained by

simulation for various loop topologies. Also, the Bux and Schlatter model yields to a quite straight forward queueing analysis. We shall, therefore, use this model in our analysis of the dynamically reconfigurable register-insertion network.

An important point, made by Bux and Schlatter in the above mentioned study, is that the grade of service to various NAUs depends upon their geographic location. Our research shows that, because of this effect, the performance of a R-I network depends on the relative physical location of its NAUs.

In the next chapter the performce of a dynamically reconfigurable register-insertion (DRR) network is analysed and an expression for the end-to-end delay as function of the relative positioning of its NAUs on the logical ring is derived. The following chapters show that such a formulation can be used to optimize the logical configuration of a DRR network.

7. Performance analysis of dynamically reconfigurable register-insertion ring networks

In the previous chapter, the literature on the performance analysis of the register-insertion ring network was surveyed. It was concluded that the queueing model developed by Bux and Schlatter [7] most accurately models the physical operation of a register-insertion network, and yields to a straight forward queueing analysis. In this chapter, the Bux and Schlatter model is extended and used to analyse the dynamically reconfigurable register-insertion (DRR) network as proposed by Lee in 1979 [26, 25], and an expression for the average message transfer delay is derived.

This is followed by a verification by simulation which shows satisfactory agreement between the results of the analysis, performed earlier, and those obtained by simulation. An important conclusion is drawn in section 8 from this analysis - it confirms that the optimization approach of chapters 4 and 5 indeed results in improved network throughput.

For ease of understanding and to focus attention on topics specific to performance analysis, attention is drawn to some of the work reported earlier in the thesis.

7.1. Introduction

A primary motivation for the design of reconfigurable ring networks was the vulnerability of single loop networks to link and node failures [64, 49, 6, 45]. Failure of a single link can disrupt the whole network. A basic safeguard against node failure is to incorporate a by-pass relay which isolates the malfunctioning node from the network
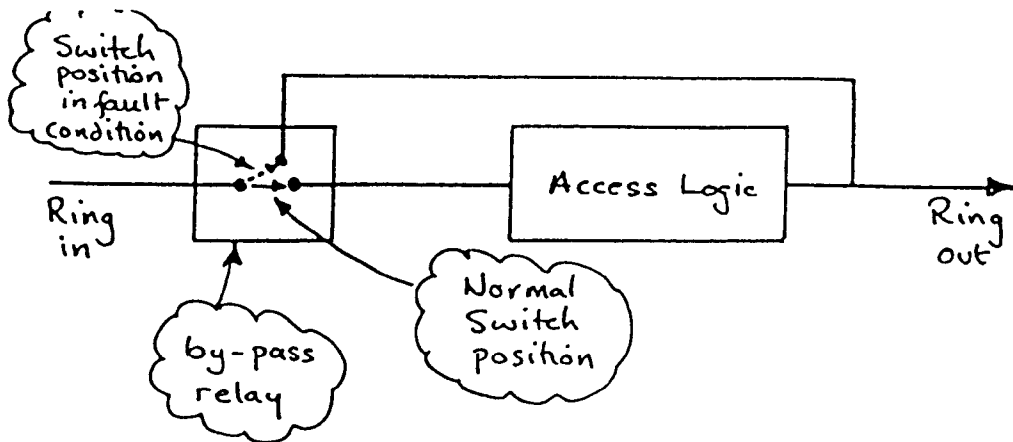
Fig. 7.1. An NAU using a by-pass relay

in the event of power failure in the node [6, 15] (see figure 7.1). Since failure of power
in a node is only one of several modes of failure which can occur, techniques have been
developed to increase the reliability of loop networks by reconfiguring the network in
the event of failure of a network component. One way to facilitate the reconfiguration
of a ring network is to arrange the physical topology of the network in a star shape
(Fig. 7.2) by looping all inter-node links back through a central location so that broken
down lines or nodes can be by-passed [49, 6].

An alternative technique to improve the reliability of a loop network is to incorporate
redundant links. The concept of distributed double-loop network (DDLCN) was intro-
duced by Wolf et al in 1978 [30]. In this network architecture, each link is duplicated
in the reverse direction, forming a complete forward loop and a complete backward
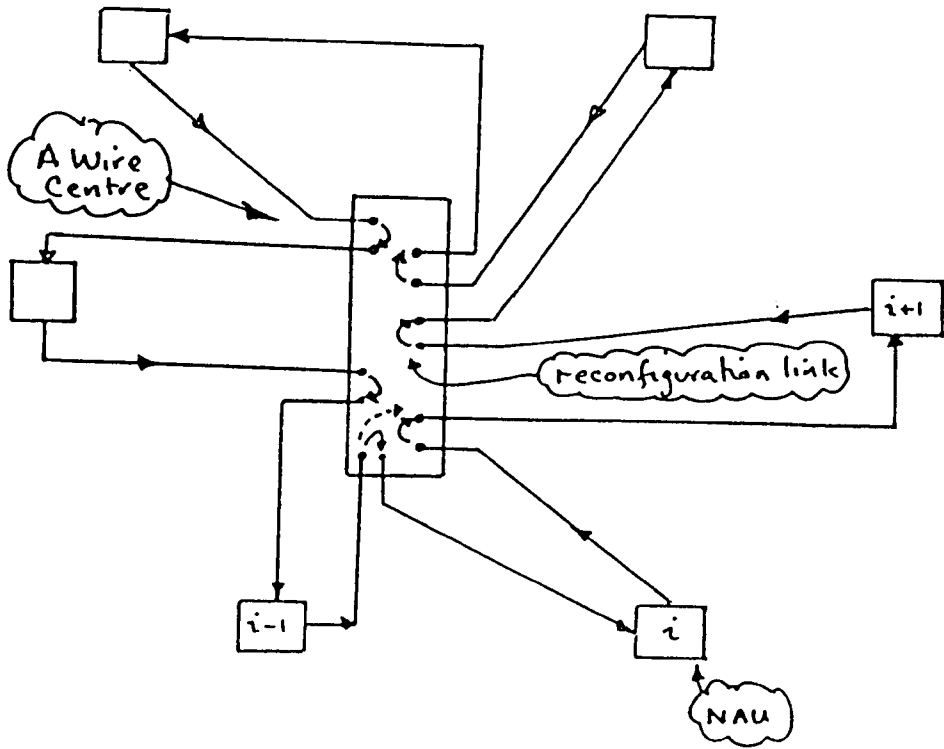
isolates node i
from the ring

Fig. 7.2. A star shaped ring

loop (figure 7.3). Both forward and reverse links are simultaneously active, the decision of which link to use to relay a particular message depends on the intended destination of the message. The mechanism of simultaneously using links in both directions not only improves the reliability of the network, but also decreases the average number nodes a message has to pass before reaching its destination node. Therefore, the DDLCN exhibits much improved performance compared with the single loop [64]. The publication of the design of the DDLCN was followed by several other multiply connected loop architectures, which primarily differ in their choice of the distance between the neighbouring nodes on the backward loop, taking the primary loop as the reference. The *daisy chain* network, proposed by Grnarov et al in 1980 [17], connects alternate nodes on the reverse loop. Raghavendra and Gerla published an analysis of double-loop networks in 1981 [45] demonstrating that the performance of a double loop can be maximized by selecting an appropriate hop length on the secondary loop (i.e. the

distance between two nodes connected by a secondary link). A survey of multi-connected loop topologies was published by Raghavendra and Silvester in 1986 [46], where it is observed that loops with higher multiplicity than doubly connected loops, such as the triply connected *chordal ring* proposed by Arden and Lee in 1981 [1], do not offer significant performance or reliability improvement over doubly connected loops. The authors also show that the optimal double loop of Raghavendra and Gerla [45] has the best reliability and performance characteristics among double loops. The survey is, however, limited to those multiconnected loops which have uniformly distributed links. Therefore, the more general multiconnected architecture of dynamically reconfigurable register-insertion network proposed by Lee in 1979 [26, 25], is excluded from the discussion.
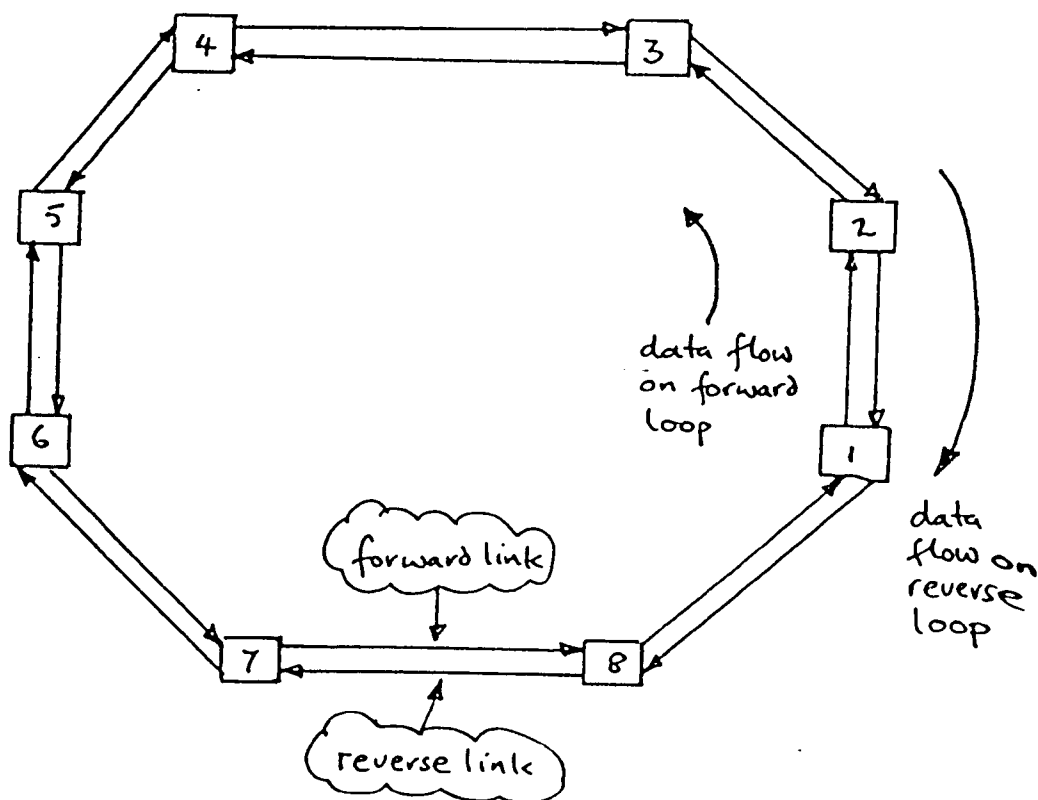
Fig. 7.3. An eight node DDLCN

The dynamically reconfigurable register-insertion network has a physical topology of a Hamiltonian graph [63] over which a subset of links is used to form a logical ring. Automatic failure recovery is achieved by finding a different logical ring which by-passess the failed component. Thomas and Yasin [57, 56, appendices B and C] show that such an architecture can easily adapt to changing traffic patterns, thereby providing high performance in a varying environment.

An important difference, between the multiconnected loop approach pioneered by DDLCN and the reconfigurable architecture proposed by Lee, is the complexity, and, therefore, cost, of the associated network access unit (NAU). Since all links of a multiconnected loop actively carry data at all times, each NAU is required to handle more than one ring-input (or relay) queues and it needs to distribute its output on more than one ring-output links. This increases the complexity of the NAU interface circuitry and requires non-trivial routing algorithms, as can be seen from the interface design for the DDLCN by Tsay et al [59]. In contrast, each DRR NAU, at any instant of normal operation, has only one active ring-input queue and only one active link for output to the ring. This simplifies the interface design and the routing is trivial. The DRR, however, requires that a two-entry routing table be incorporated in each NAU. This appears to be relatively simple to implement as suggested by Lee [25].

7.2. The dynamically reconfigurable register-insertion network (DRR)

A dynamically reconfigurable register-insertion network (DRR) consists of a number of bi-directional links connecting the nodes of the network. Each node incorporates a two-entry routing table. The entries show which of the incident links are the currently selected in-link and out-link. At any instance of normal operation, the set of in-link and out-link pairs form a ring. Figure 7.4 shows a typical DRR network during normal operation. Here, the dashed lines represent the physical bi-directional links and the
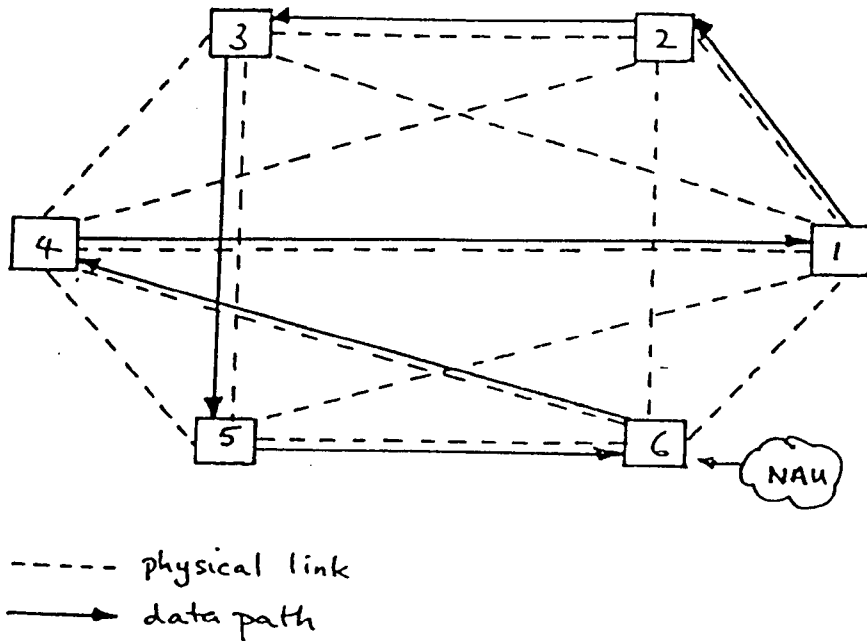
Fig. 7.4. A DRR network

solid lines show the selected data paths, data flows in the direction of the arrows. A tour around the selected data paths forms a ring contatining all nodes. In the event of failure of a link participating in an operating ring, the entries in the routing tables in the nodes are modified, if possible, by a control node to form a new ring without using the failed link. As long as the physical topology is Hamiltonian, such a ring exists [63]. An example of such a reconfiguration is shown in figure 7.5, following failure of the link joining nodes 3 and 5.

The numbering of the NAUs of a DRR network is purely arbitrary. Clearly the output of NAU_i need not be joined with the input of NAU_{i+1}, nor does the input of NAU_i necessarily joins with the output of NAU_{i-1}. For ease of reference, we define two functions over the set of NAU numbers, namely $succ(i)$ and $pred(i)$, such that $pred(i)$ refers

-74-
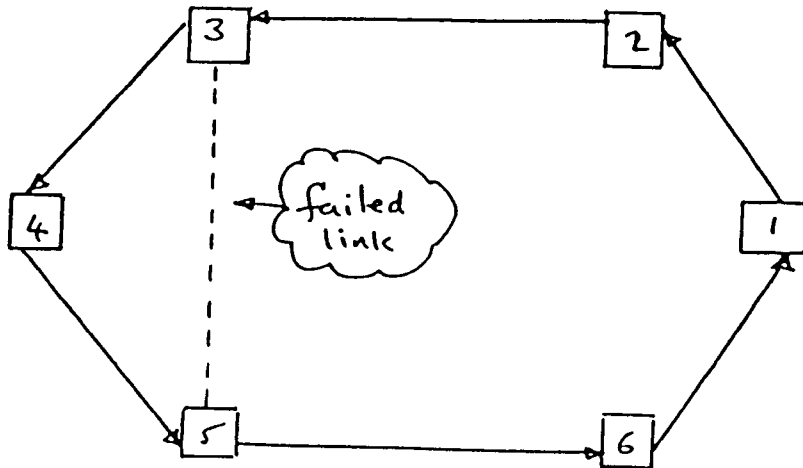
Fig. 7.5. Reconfiguration of the DRR of fig. 7.4

to the NAU whose output data forms the input for *NAU*$_i$ and *succ(i)* refers to that NAU whose input comes directly from the output of *NAU*$_i$.

7.3. Network Access Unit (NAU)

A network access unit (NAU) provides a mechanism for connecting devices to the physical medium of the network (twisted-pair wire, co-axial cable or optical fibre), and the logic necessary to allow orderly access to the communications channel (fig 7.6). It also contains the buffers required to store messages in transit while the desired output channel is busy. This situation occurs, for example, when a message is received from the ring, and which is to be relayed onto the next NAU, during the time that the output channel is busy in outputting a locally generated message. The receiver section of

the NAU {RC in fig 7.6} de-multiplexes in-coming messages onto either of its two out-put data paths: the switching policy being that the path to the output buffer is chosen if the message has reached its destination (as indicated by the message header), other-wise the path to the variable length shift register (VSR) is selected.
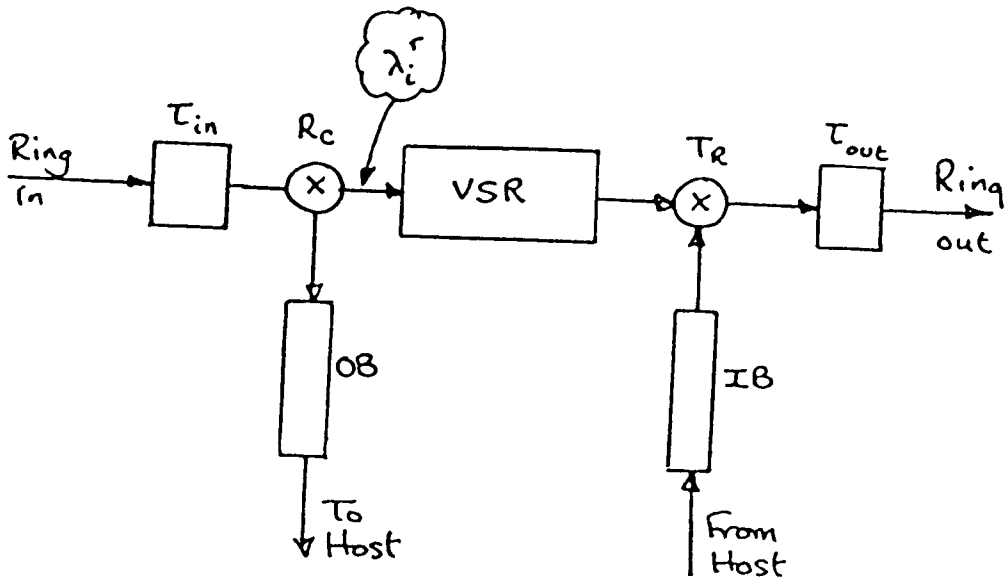
Fig. 7.6. A DRR Network Access Unit (NAU)

The transmitter section performs the less trivial task of multiplexing the input buffer and the VSR onto the ring output. Several alternatives for this switching policy (referred to as the access control mechanism) for the register-insertion ring have been investigated [14, 58, 7]: the more interesting being the distributed loop computer net-work (DLCN) and asynchronous time division multiplexing (ATDM). The ATDM mechanism assigns higher priority to the messages already on the network (i.e. the VSR input to the transmitter), while the DLCN assigns higher priority to messages

arriving from the attached device (i.e. the IB input to the transmitter) as long as there is enough space in the VSR to store the incoming messages. Since the total available length of the VSR is finite, DLCN results in an alternating priority scheme, whereby priority is given to the IB under low traffic loads and alternates between IB and VSR under high traffic loads. When analysing its behaviour, it is often the case that DLCN is modelled by the so-called modified DLCN (MDLCN) [58], which has a shift register of infinite size and, therefore, assigns a permanently higher priority to the IB. It has been shown [58, 22, 7] that the ATDM and MDLCN (and indeed any combination of the two) behave identically with regard to the average end-to-end delay per message. Thus, the discussion can be restricted to the MDLN, without any loss of generality.

The messages within each buffer (the IB or the VSR) are dealt with on the *first come first served* principle (also known as the *head-of-line* queueing discipline, abbreviated as *HOL*). In order to retain consistency of data on the network at all times, the priority schemes discussed above are non-pre-emptive. That is to say, the transmission of a message is not interrupted by the arrival of a higher priority message.

An NAU introduces a time delay in the propagation of a message. This delay can be split into those components which are fixed for a given network layout and those which vary with time. The two fixed components are denoted τ_{in} and τ_{out}. The first delay, τ_{in}, is introduced by the receiver section of an NAU. This is approximately equal to the time to receive a message header, during which the destination address contained therein can be checked. The second delay, τ_{out}, is the propagation delay caused by the finite time required for a bit to travel from the output of one NAU to the input of the next. The combined effect of τ_{in} and τ_{out} is called the latency, τ, of an NAU.

The time-varying part of the aforementioned delay consists of four quantities. Considering a message which has just arrived at the receiver section of an NAU, these quantities are:

(1) Time required to complete the current transmission, if any.

(2) Time required to transmit the messages already in the input buffer IB which are to be transmitted before this message.

(3) Delay caused by the arrival of new messages in the input buffer IB which are to be transmitted before the tagged message.

(4) Time required to transmit the messages which are already in the variable length shift register VSR.

Note that the messages in the input buffer IB have priority over messages in the variable length shift register VSR, and also that the head-of-line discipline requires that the tagged messages will not be delayed by arrival of new messages in the VSR.

Next, consider the arrival of an external message in the IB. In this case, the delay comprises of items (1) and (2) only. There is no contribution from item (3) because of the HOL queueing mechanism. Also, higher priority of the IB requires that item (4) above must also be zero.

Finally, consider a message which has just arrived at its destination. After the address checking delay, a received message is put in the output buffer (OB). Here a message waits for service from the attached device. This waiting time depends only on the functionality of the attached device and is independent of the performance of the underlying communications network. In order to isolate the effects of external devices from the following performance analyses and, hence, concentrate on the properties of the communications network, it is assumed that the output buffer (OB) is part of the

external device. Thus a message leaves the network as soon as it is inserted into the OB.

The above description of the operation of an NAU lends itself to analysis by queueing theory. The next section presents a queueing model of an NAU.

7.4. Queueing Model of a Network Access Unit (NAU)
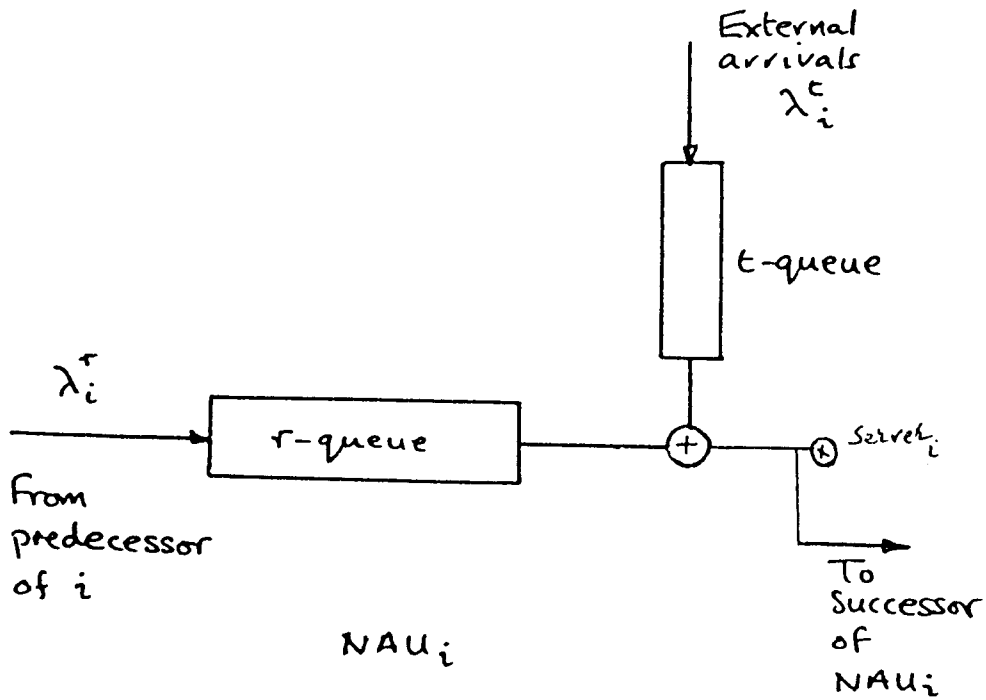
Fig. 7.7. Queueing Model of a DRR NAU

An NAU can be viewed as a single server (transmitter) serving two queues of messages, namely the IB and the VSR (Fig 7.7). The output buffer (OB) is not part of the system under study for reasons discussed above. Also, there is no queueing in the

receiver section: the receiver puts every received message in the OB or the VSR after a constant delay.

The arrival pattern of external messages at an NAU is assumed to be random. The attached devices generate new messages as needed, and thus the messages come from a non-exhaustive source. For such a system the inter-arrival rate has a Poisson distribution [52]. That is, the probability density function for the inter-arrival times is negative exponential, given by $\lambda e^{-\lambda t}$, where λ is the mean arrival rate of external messages at an NAU.

The input process from the ring is more complex. Being a combination of random processes, (the outputs of previous NAUs) this itself is a random process. In the main, it is the simplifying assumptions for modelling of the ring-input process which distinguish various analytical approaches.

In the following discussion, the superscripts t and r refer to parameters relating to the external and ring queues respectively. The subscript i refers to the i'th NAU. The following is a list of parameters used in subsequent analyses.

N   the number of nodes in the network;

C   transmission speed of the network, bits/second;

λ_i^t: arrival rate of external messages at NAU_i;

λ_i^r: arrival rate of messages from ring at NAU_i;

$E[x]$ $=\dfrac{1}{\mu}$, the mean length of messages on the network;

$E[T]$ $\quad = \dfrac{E[x]}{C}$, the mean processing (or service) time of a message;

τ $\qquad$ the constant delay introduced by a NAU in the path of each relayed message, referred to as the latency of a NAU;

$[p_{ij}]$ $\qquad$ the routing matrix, where p_{ij} is the probability that a message originating at NAU_i is destined for NAU_j;

ρ_i $\qquad$ the traffic intensity or the channel utilization defined as $\lambda_i * E[T]$;

Γ_{ij} $\qquad$ average transfer delay for messages originating at NAU_i and destined for NAU_j;

Γ $\qquad$ mean end-to-end transfer delay of the network.

7.5. Analysis of the NAU Model
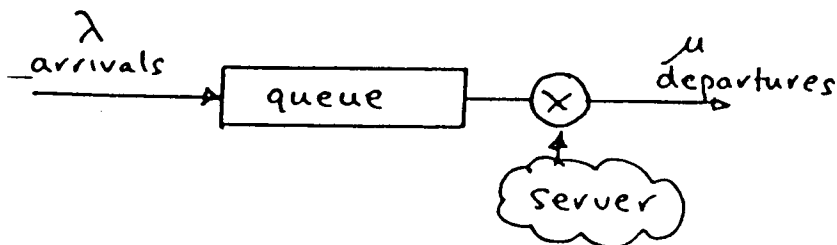
Fig. 7.8. A Simple Queueing System

Consider the queueing system in Fig 7.8. Here, the mean arrival rate of customers is λ, and the mean departure rate of customers is μ. Let p_k be the probability that there are exactly k customers in the system (queue + service). The arrival of a new customer takes the system up to the next state and the departure of a customer moves the system down one state. Since the arrival rate of customers is λ, the transition rate from state k to state k+1 is $\lambda \, p_k$. Similarly the transition rate from state k+1 to k is $\mu \, p_{k+1}$. If the mean arrival rate is greater than the mean departure rate, the queue will grow infinitely with time, resulting in an unstable system. A computer network can not be allowed to remain in an unstable state as the queueing delays will soon rise beyond acceptable limits. On the contrary, the transmission speed of the network is selected so that the mean service time per message is sufficiently less than the mean arrival rate of external messages. Such a system will eventually reach a steady-state. In a steady-state system, we must have:

$$\lambda \, p_k = \mu \, p_{k+1} \qquad (7.1)$$

The relation (7.1) is sometimes referred to as the principle of detailed balancing.

Solving (7.1) iteratively, we obtain

$$p_k = \rho^k \, p_0 \qquad (7.2)$$

where $\rho = \dfrac{\lambda}{\mu}$, is known as the traffic intensity.

The mean number of customers in the system, S, is given by

$$S = \sum_{k=0}^{\infty} k * p_k \qquad (7.3)$$

If the average system time (queueing + service time) of a message is Γ, then Little's result [27] requires that

$$S = \lambda \ \Gamma \qquad\qquad (7.4)$$

Using the expressions for S and p_k, and the fact that sum of all probabilities must be 1, we have

$$\Gamma = \frac{1}{\mu - \lambda} \qquad\qquad (7.5)$$

Next, consider a non-pre-emptive head-of-line queueing system with P number of distinct priorities and a single server. Let the mean queueing time of messages of priority p be W_p, the mean arrival rate of customers of priority p be λ_p, and the mean service time be T, then we have [24]

$$W_p = \frac{W_0}{(1 - \sigma_p)(1 - \sigma_{p+1})} \quad p = 1,2,....P \qquad\qquad (7.6)$$

where,

$$W_0 = \sum_{p=1}^{P} \lambda_p \ E[T^2] / \ 2$$

and,

$$\sigma_p = \sum_{i=p}^{P} \rho_i$$

In the MDLCN NAU model developed earlier (§ 7.2), there are two priority queues, namely VSR and IB, thus P is 2. Also, IB has higher priority than VSR. Hence, making appropriate substitutions, the expressions for the mean queueing time in the t-queue, W_i^t, and the mean waiting time in the r-queue, W_i^r, for the i'th NAU are given by:

$$W_i^t = \frac{(\lambda_i^t + \lambda_i^r) \ E[T^2]}{2 \ (1 - \rho_i^t) \ E[T]} \qquad\qquad (7.7)$$

and

$$W_i^r = \frac{(\lambda_i^l + \lambda_i^r) E[T^2]}{2 (1 - \rho_i^l - \rho_i^r) (1 - \rho_i^l) E[T]}$$ (7.8)

respectively, where

$$\rho_i^l = \lambda_i^l E[T]$$

and

$$\rho_i^r = \lambda_i^r E[T]$$

The end-to-end delay suffered by a message originated at NAU_i and destined for NAU_j, Γ_{ij}, is then given by :

$$\Gamma_{ij} = W_i^l + \sum_k W_k^r + \sum_k \tau_k + \tau_j$$ (7.9)

where the subscript k runs over all NAUs which lie between NAU_i and NAU_j.

In equation (7.9), the first term is the average waiting in the t-queue at the source, NAU_i, the second term is the sum of average waiting times in the r-queues at the intermediate NAUs, and the remaining terms represent the latency of the intermediate and destination NAUs.

The average transfer delay per message for NAU_i , Γ_i, is the appropriately weighted average of Γ_{ij} for all j, and the average transfer delay per message for the network, Γ, is the appropriately weighted average of Γ_i over all i. The respective expressions are:

$$\Gamma_i = \sum_{j=1}^{n} \Gamma_{ij} \, p_{ij}$$ (7.10)

and

$$\Gamma = \sum_{i=1}^{n} \Gamma_i \, \lambda_i^l \tag{7.11}$$

7.6. Performance analysis of a DRR network

Following the performance model proposed in § 7.4, we make the following assumptions :-

(i)   Poisson arrival rates λ_i^l.

(ii)   generally distributed message transmission times T.

(iii)   a known probability matrix P, where element p_{ij} is the probability that data originating at node i is destined for node j.

(iv)   ignoring the nodal delay and the propagation delay, that is, $\tau_i = 0$ for all nodes.

As discussed in the previous chapter on R-I ring networks and verified by simulation for the DRR network reported in chapter 8, the average transfer delay of a register insertion ring network is independent of whether a station priority scheme or a ring priority scheme is employed. Therefore, without loss of generality, we shall consider only the ring priority scheme.

Defining the traffic intensity of external messages at node i as

$$\rho_i^l = \lambda_i^l \, E[T] \tag{7.12}$$

and the intensity of the relayed traffic as

$$\rho_i^r = \lambda_i^r \, E[T] \tag{7.13}$$

it is shown in [7] that the waiting times for messages arriving in the ring queue w_i^r and in the external queue w_i^e are given by:

$$w_i^r = \frac{\rho_i^r E[T^2]}{(2 (1-\rho_i^r)E[T])} \qquad (7.14)$$

and

$$w_i^e = \frac{(\rho_i^r+\rho_i^e) E[T^2]}{(2 (1-\rho_i^r-\rho_i^e)(1-\rho_i^r) E[T])} \qquad (7.15)$$

Next, we define x_i to be the distance in number of arcs of node i from an arbitrary reference node 1, in the direction of data flow on the ring [57], then, following the notation used by Bux and Schlatter [7, Equation 8], the probability e_{ki} that a frame originating at node k passess through node i is given by

$$e_{ki} = \sum_{j \in I(k,i)} p_{kj} \qquad (7.16)$$

where

$$I(k,i) = \begin{cases} S-\{x_j \mid x_k \leqslant x_j \leqslant x_i\} & x_k < x_i \\ \{x_j \mid x_i < x_j < x_k\} & x_i < x_k \end{cases}$$

and $S = \{1, 2, ..., n\}$.

Next, define d_{ij} to be a 0-1 variable such that $d_{ij} = 0$ if $x_i < x_j$, and 1 otherwise. Thus

$$d_{ij} = 1 - d_{ji} \qquad \text{for } all \ i,j; \ i \neq j \qquad (7.17)$$

For any triplet (i.j.k) of nodes. it is clear that

$$j \in I(k,i) \quad iff \quad (d_{kj} + d_{ji})d_{ik} + d_{ji} d_{kj} d_{ki} = 1 \qquad (7.18)$$

Using equation (7.17) and the fact that

$$d_{ik} d_{kj} d_{ji} = 0 \quad for\ all\ i,j,k \qquad (7.19)$$

equation (7.18) reduces to

$$j \in I(k,i) \quad iff \quad d_{ji} - d_{ki} + d_{kj} = 1 \qquad (7.20)$$

and the expression for e_{ki} becomes

$$e_{ki} = \sum_{j=1}^{s} p_{ij} \ (d_{ji} - d_{ki} + d_{kj}) \qquad (7.21)$$

Therefore. the total traffic passing through node i due to arrivals at all other nodes in the network is

$$\lambda_i' = \sum_{k=1}^{s} \lambda_k' \ e_{ki} \qquad (7.22)$$

$$= \sum_{k=1}^{s} \sum_{j=1}^{s} \lambda_k' \ p_{kj} \ (d_{jk} - d_{ki} + d_{kj})$$

An important quantity is the total through traffic in the network. Λ'. because it represents the wasted network bandwidth due to traffic being relayed at intermediate nodes. This is given by

$$\Lambda' = \sum_{i=1}^{s} \lambda_i' \qquad (7.23)$$

$$= \sum_{i=1}^{s} \sum_{j=1}^{s} \sum_{k=1}^{s} \lambda_k' \ p_{kj} \ (d_{jk} - d_{ki} + d_{kj})$$

It is interesting to note that. using appropriate expressions for the volume elements v_{ij} and the inter-nodal distances l_{ij}. the objective function derived for the optimization problem in chapter 4. expression (4.1). is identical to the total through traffic in the network. This is an important result as it shows the importance of our minimization exercise. i.e.. the configuration achieved as a result of minizing expression (4.1) has the least through traffic on the network. thereby minimizing the wasted network bandwidth.

Assuming nodal delays. τ_i. to be zero. the transfer delay from node i to node j. Γ_{ij}. consists of the waiting time in the transmit queue of node i. w_i^t. and the waiting times in the relay queues of all intermediate nodes k. given by w_k^r.

By definition of d_{ij}. it can be easily seen that

$$d_{jk} - d_{ik} + d_{ij} = 1 \tag{7.24}$$

holds if and only if node k lies between nodes i and j when traversing along the loop from node i to node j in the direction of the flow of data on the loop.

Therefore. the transfer delay from node i to node j is

$$\Gamma_{ij} = w_i^t + \sum_{k=1}^{s} w_k^r (d_{jk} - d_{ik} + d_{ij}) \tag{7.25}$$

The average transfer delay of the network. Γ. is then calculated by averaging Γ_{ij} over all nodes weighted by the proportion of external traffic at each node:

$$\Gamma = \frac{\sum\limits_{i=1}^{s}\sum\limits_{j=1}^{s}\lambda_i' \, p_{ij} \, \Gamma_{ij}}{\sum\limits_{i=1}^{s}\lambda_i'}$$

(7.26)

or, following the notation used by Bux and Schlatter [7],

$$= \frac{\sum\limits_{i=1}^{s}(\alpha(\rho_i' + \rho_i') - \alpha(\rho_i'))}{\sum\limits_{i=1}^{s}\lambda_i'}$$

where

$$\alpha(\rho) = \frac{\rho^2 \, E[T^2]}{2(1-\rho) \, E[T]^2}$$

It can be seen from the above expression for the average transfer delay of the network, Γ, that the only configuration dependant parameter affecting Γ is the λ_i'. Therefore, by minimising Λ', Γ can be decreased.

7.7. Simulation

A simulation package, described in chapter 8 (§8.2), was used to verify the results derived in the previous section. The average message transfer delay across a 20-node network obtained from simulation, for symmetric traffic conditions, is shown in figure 7.9, together with the delays calculated by the analytical formulae. The two sets of results are in good agreement. The analysis appears to under-estimate the transfer delay for low loads. This is because the latency of each NAU has been assumed to be zero in the analysis. For a 20 node network under symmetric traffic conditions,

considered in figure 7.9, the average number of nodes traversed by each message before it reaches its destination is 10. The simulation model assumes the per NAU latency to be 24 bit-times or 0.024 milli-seconds for the assumed transmission speed of 1 million bits per second. Therefore, the contribution to the average transfer delay arising because of non-zero latency is 0.24 msec. This is a significant amount for low loads where the overall transfer delay is of the order of a few milli-seconds.
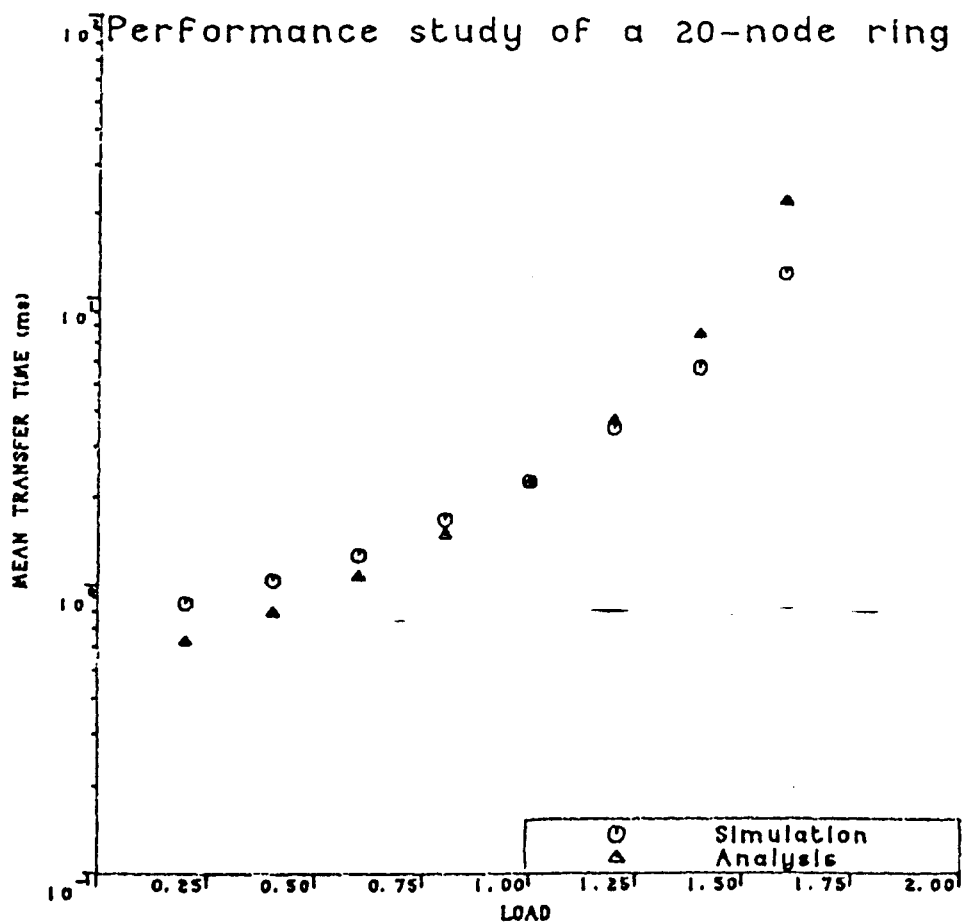
Fig. 7.9.Simulation and analysis of a 20-node DRR network

7.8. Conclusions

In this chapter, the performance of a dynamically reconfigurable register-insertion (DRR) network has been analysed using queueing theory. The queueing model proposed by Bux and Schlatter [7] for analysis of the DLCN loop is extended, by including configuration dependant parameters, to be applicable to the DRR network.

An important result, derived in this chapter, is the expression (7.25) which, when used in conjunction with (7.26), yields the average transfer delay per message of a DRR network.

It is also shown that the objective function (Eq. 4.1), proposed in Appendix B for optimisation purposes, corresponds to the total through traffic in the network. Furthermore, it can be seen from expression (7.26), that the only configuration dependant parameter in the calculation of the average transfer delay is the through traffic. Therefor, by minimising through-traffic (or the objective function of Eq. 4.1), the transfer delay of the network can be reduced. This will clearly improve the performance of the communications subnetwork.

Finally, since minimising the objective function of Eq. 4.1 reduces the through traffic in the network, the optimal configurations, obtained in appendices B and C, can handle higher external loads, thereby, resulting in improved network throughput.

8. Methods of solution

This chapter describes the numerical methods investigated to solve the optimization problems developed in chapters 4 and 5. A description of a simulation package, developed to verify the analytical results, is presented in section 2. Concluding remarks appear in section 3.

8.1. Numerical Methods

As mentioned in chapter 4, two approaches were investigated for solving the optimization problem: (1) integer programming and (2) set theory.

8.1.1. Integer Programming

The first approach was to use a general purpose integer programming package to confirm the applicability of the optimization technique. Initially, a package called NCLP [60], developed by the Open University Academic Computing Service, was used on a DEC-system-20 computer system. NCLP is suitable only for very small problems and was found un-satisfactory in handling networks of more than six nodes. Later on, a suite of mainframe integer programming packages, the Mathematical Programming System Extended (MPSX) / Mixed Integer Programming (MIP) / Generalized Upper Bounding (GUB) package [10] on the IBM 3081 computer system at Cambridge University was used. This package works as follows. First, a relaxed linear model of the problem to be solved is constructed by dropping the integer constraints from the variables. This linear problem is solved by the MPSX program. Next, the MIP program tries to find integer solutions in the neighbourhood of the optimal solution produced by MPSX. The GUB macros are used to efficiently handle the special constraints of

upper and lower bounds. The MPSX package requires a fixed format input. The preparation of such input for problems of a significant size is a lengthy, tedious process and is a potential source of error. Thus, in order to facilitate the analysis of a number of similar problems, a preprocessor, called MAGIC [11], was acquired from the Edinburgh Management Science Partnership of the University of Edinburgh. The program MAGIC is able to generate the matrix format required by MPSX by converting instructions presented in a form close to that of the mathematical model. This combination of MAGIC and MPSX/MIP/GUB allowed reasonably large size problems to be tackled, although the computer time required to solve even a 10 node problem was found to be prohibitively long. MIP is an iterative procedure, it finds local optimum solutions and, if it is not satisfied by the result, the search for the global optimum is continued in by taking some other branch of the search tree. In order to assist in tracking the long iterative procedure, MIP prints out every integer solution that it finds. If the program is allowed to run to completion, which is feasible for vey small problems only, MIP produces the optimum solution and confirms the optimality of the solution by an exhaustive search. It was found that, for the configuration optimization problem discussed in this thesis, the computer time required and the amount of print-out generated by the MIP package in search of the global optimum solution is prohibitively large for networks of more than five nodes. It is suggested, here, that the very first integer solution obtained by MIP be used as a near optimal solution. In an investigation into the routing problem for the IBM's SNA network [53], Tcha and Maruyama also had to abandon the search for the true optimum integer solution because of the excessively large computation time required. They also conclude that an integer solution, found in the neighbourhood of the optimum solution of the relaxed linear model, is a good sub-optimal solution.

Figure 8.1 shows the performance of a 10-node DRR network whose configuration corresponds to the first integer solution produced by integer programming techniques for asymmetric traffic conditions. This is compared with a worse case configuration,
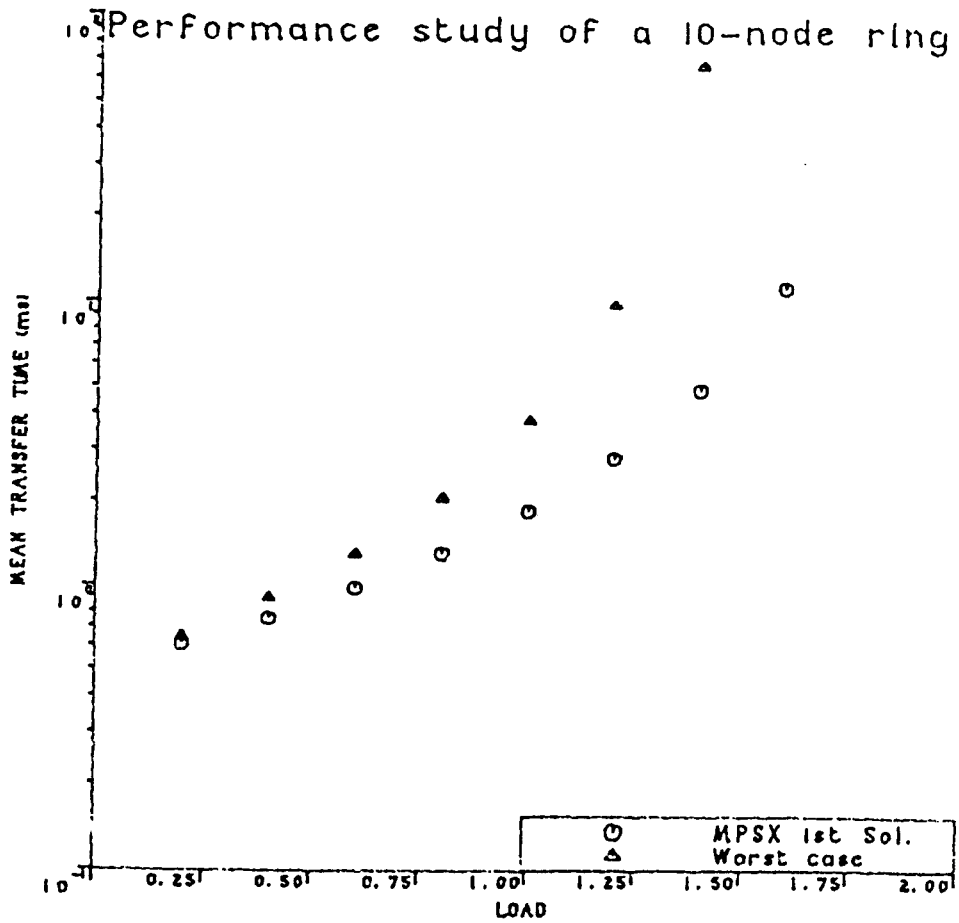
Fig. 8.1 Performance study of a 10 node DRR network

produced by reversing the order of NAUs in the aforementioned solution, as suggested
in appendix B. It is obvious from this comparison that the optimised configuration
exhibits much improved performance, particularly under heavey traffic loads. The per-
formance of an arbitrarily chosen configuration, as suggested by Lee [25], is most likely
to be somewhere between the two extremes shown in the figure. There are obvious
performance benefits to be had by adopting the optimisation methodology presented in
this thesis.

8.1.2. Set Theoretic Approach

As the integer programming methodology is very expensive in the use of computing resources, an alternative method of solution was sought. It was found [57] that, for a ring topology, the integer 0/1 variables d_{ij} must obey the following transitive relationship:

$$d_{ij} = 0 \quad \text{and} \quad d_{jk} = 0 \quad \textit{implies} \quad d_{ik} = 0 \quad \text{for } \textit{all } i, j, k \tag{8.1}$$

A Pascal program called CONFIG was developed to solve the optimization problem by deciding the polarity of d_{ij}s for minimizing the objective function (4.1) and subject to the constraints (7.1). In case of a conflict, the choice is based upon the d_{ij} which has the highest absolute value of the co-efficient in the objective function. A more detailed account of the operation of CONFIG appears in [57] where it is also shown that CONFIG can get trapped in local minima, thereby producing sub-optimal results. A further limitation of CONFIG is that it can solve only those problems which are based on a fully connected physical topology. Incorporation of the constraints developed in appendix B, to deal with missing links, is a non-trivial problem which is the basis for further research.

8.2. Simulation Package

A discrete event simulation program using the SIMULA language [3] on DEC-20 computer system, was developed to verify the analytical results. The simulator, SIMDRR, considers the configuration of the network being simulated to be one of the parameters of the problem. It reads the given configuration as input data and routes messages from source node to the destination node via the given configuration, see figure 8.2. Thus, it is possible, using this simulator, to compare the performance of a given network under a given traffic pattern with different configurations.
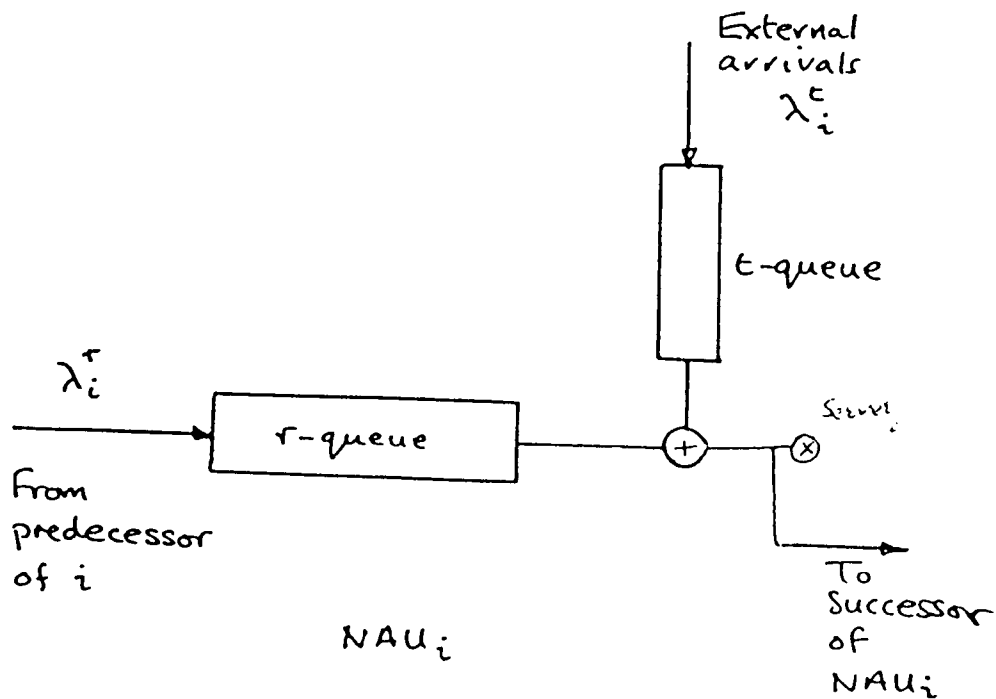
Fig. 8.2. Simulation model of a DRR network

In order to be able to simulate both the station priority (i.e. MDLCN) and the ring priority (i.e. ATDM) protocols, the simulation program is divided into three parts. The middle part has two versions, one for each protocol. By using the appropriate middle part (RSP or RRP), either of MDLCN or ATDM can be simulated. Various simulations were carried out which confirm the analytical findings of Thomasian and Kanakia [58], Liu et al [31] and Bux and Schlatter [7], that the performance of a register-insertion network is identical for both protocols. The results of simulation runs for a sixteen node network under symmetric traffic conditions for MDLCN and ATDM are plotted in figure 8.3.

Among other data, the inputs to the simulator include the relative activity of each node (i.e. $\frac{\lambda_i'}{\sum_i \lambda_i'}$) and the probability matrix $[p_{ij}]$. Arrival of external messages at each
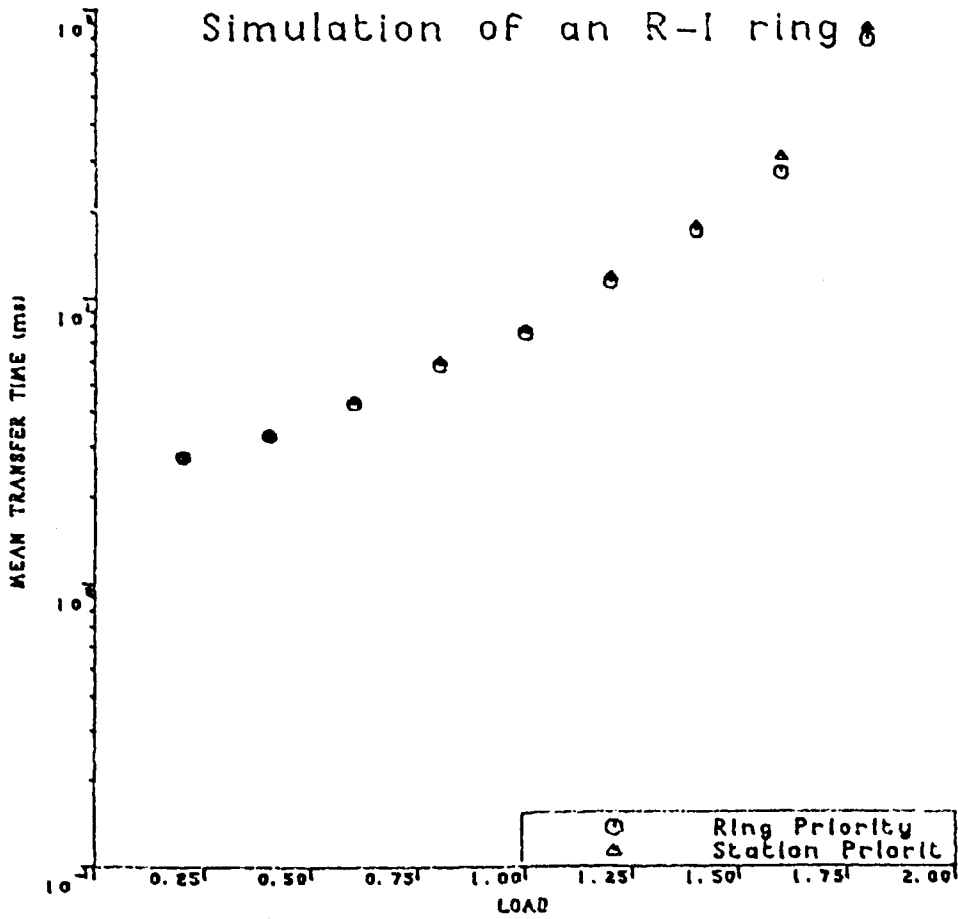
Fig. 8.3. Simulation of ring- and station-priority schemes

node is assumed to be an independant Poisson process. the arrival rate is determined from the load (i.e. throughput/transmission speed) on the network and the mean message length. Message lengths are generated by random sampling of the negative exponential function.

All messages are tagged by a message number and the identity of the source node, so that it is possible to trace the progress of a message through the system. A message arrival is deemed to have occured when a new message is placed in the input buffer of an NAU and a message leaves the system as soon as it arrives at the relay queue of the destination NAU.
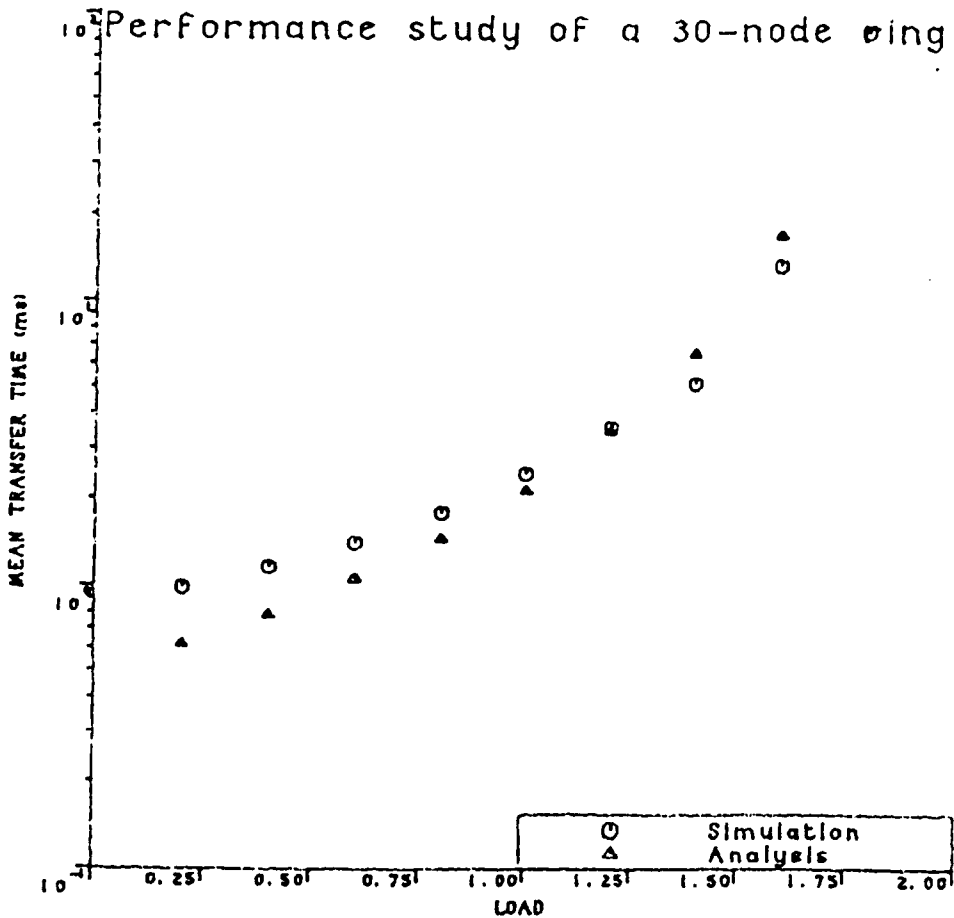
Fig. 8.4. Analysis and simulation of a 30 node DRR network

Each NAU is modelled as a *check and forward* server, so that messages are delayed at intermediate nodes for a time equal to the latency per node, which is the time taken by a node to receive and check the destination address contained in the header of the message. After this delay, an identical message, i.e. of similar length and with similar source and destination addressess, is put in the relay queue of the next NAU, as determined by the given configuration. The transmission of a message keeps the network access unit (NAU) busy for a time equal to the message length divided by the transmission speed of the network. The end-to-end transfer time of a message is determined by the sum of the total delay incurred during its life-time and the message transmission time.

Simulation was also used to verify the results obtained by queueing analysis reported in chapter 7. A reasonable agreement was found between the two sets of results for various network configurations. A comparison of simulation and analytical results for a 30 node DRR network under symmetric traffic is given in figure 8.4.

Among the statistics gathered by the simulator are the maximum and average station/ring queue lengths for each node and for the network, and the average end-to-end transfer time per message.

8.3. Conclusions

The integer programming approach is expensive in computing resources. It is, therefore, suggested that the first feasible solution produced by MIP (or any similar integer programming package) be taken as a near optimal solution. It is our feeling that, because of the way the problem is structured by giving weighting to those node pairs which have heavier traffic un-balance, there is little to be gained from trying to find the true optimum solution.

The set theoretic approach is very efficient and can be used to obtain quick near-optimal solutions for fully connected networks.

A simulation package has been developed to verify the analytical results. This package is specifically designed to simulate the dynamically reconfigurable networks. It can simulate different configurations for a given network subject to a given traffic pattern. A number of diagnostic outputs can be produced such as the life history of a message in the network.

9. Conclusions

This chapter summarises the research reported on here, and identifies various directions for further research. Sections 1 and 2 present a summary of the results, obtained during the course of the research reported in the body of this thesis, in the fields of low cost local area network (LAN) management and configuration optimisation. Suggestions for future research work are summarised in section 3.

9.1. LAN Management

In the first part of this research, i.e. LAN management, a survey of literature revealed that the term *LAN management* is used to encompass a diversity of functions. Also, the difference between the functions carried out by automatic management (i.e. by computerised management packages) and those carried out by human managers is often obscure. The contribution of our research in this area is the identification of those management functions which can be incorporated economically on low cost local area networks. Moreover, an investigation into the trade-off between centralized and distributed implementation of the said functionality, identified a minimum functionality which must be incorporated in each network access unit (NAU) in order that a DLCN type network can accommodate a central network management station.

9.2. Configuration optimisation

One of the functions required of a management facility, of §9.1 above, is to collect network usage statistics in the form of a volume matrix indicating the (relative) amount of data transfer taking place between each pair of communicating nodes. Further research into the use of the so collected statistics resulted in the following

contributions to the field of configuration management for DLCN networks:-

i)   it is shown that the performance of a network can be improved by configuring the network in a particular way;

ii)   an analytical measure for the comparison of different configurations for a network under given traffic conditions has been proposed;

iii) it has been verified, both by analysis and simulation, that the performance measure of ii) above leads to an optimal configuration;

iv)   a mechanism is derived which allows a network configuration to be included, as a parameter, in the analysis of a network;

v)   queueing analysis of the dynamically reconfigurable register-insertion (DRR) network is performed;

and

vi)   a simulation model is developed, and implemented, for the DRR network.

9.3. Further research

There are several related areas of future work. Firstly, analysis of the pattern of traffic on a network is required which should define criteria for a reconfiguration attempt. Second, further investigation is required to extend the results of this research to the more general case of concatenated local networks. This is expected to lead to a better composition of each section of a concatenated LAN. Further work into the special nature of the optimisation problem may lead to more efficient algorithms for arriving at the optimum configuration. Finally, certain criteria, in addition to those proposed in the thesis, are desirable to identify when a sufficiently near-optimal solution has been obtained.

References

1. Arden, B. W. and Lee, H., "Analysis of chordal ring," *IEEE Trans. on Computers* C-30(4) pp. 291-295 (April 1981).

2. Berchanski, J. A., "some aspects of network user access procedure design," *Proc. COMNET'81: Networks from the user's point of view*, pp. 547-557 IFIP, (1981).

3. Birtwistle, G. M., Dahl, O-J, Myhrhaug, B., and Nygaard, K., *SIMULA begin*, Aurbach Pub. Inc., Philadelphia, USA (1973).

4. Brownbridge, D. R., Marshall, L. F., and Randell, B., "The Newcastle connection or Unixes of the world unite !," *Software - Practice and Experience* 12 pp. 1147-1162 (1982).

5. Bull, G. M. and Huckle, B. A., "An interface for interactive users within a hetrogeneous computer network," *Proc. of the EURO-IFIP'79 Conf.*, pp. 405-409 (1979).

6. Bux, W. and Closs, F., Kummerle, K., and Muller, H. R., "A reliable token-ring system for local-area communication," *Proc. Nat. Telecom. Conf.*, pp. A2.2.1-A2.2.6 IEEE, (November 1981).

7. Bux, W. and Schlatter, M., "An approximate method for the performance analysis of buffer insertion rings," *IEEE Trans. on Comm.* COM-31(1) pp. 50-55 (January 1983).

8. Clark, D. D. and Svobodova, L., Mamrak, S. A., Maurath, P., Gomez, J., and Janardan, S., "Guest layering distributed processing support on local operating systems," *Proc. COMPCOM'80 Spring*, pp. 438-444 (February 1980).

9. Cornelius, P., "EURONET-DIANE USER LOGFILE : A user's structured view of the similarities and differences between EURONET-DIANE hosts at the operating system and application system level," EUR 7878, Commission of European Communities (1983).

10. Corp., IBM. "Mathematical programming system extended (MPSX) Mixed integer programming (MIP) program description." SH20-908-1, IBM (August 1973).

11. Day, R. E., *Matrix generator instruction converter (MAGIC)*, Edinburgh Management Science Partnership, Edinburgh, Scotland (1982).

12. Dixon, R. C., "Ring network topology for local data communication." *Proc. COMPCON Fall'82*, pp. 591-605 IEEE, (September, 1982).

13. Farmer, W. D. and Newhall, E. E., "An experimental distributed switching system to handle bursty computer traffic." *Proc. ACM Symp. on Problems of Optimization of Data Comms.*, pp. 1-33 (October 1969).

14. Goscinski, A., Walasek, T., and Zielinski, K., "Performance study of ring structured local area networks using register insertion." *Angewandte Informatik*, (3) pp. 120-126 (March 1985).

15. Grant, J. S., "Multilink - an open network for personal computers." *Computer Comm.* 8(1) pp. 27-34 (February 1985).

16. Gray, J. P. and McNeil, T. B., "SNA multiple system networking." *IBM Systems Journal* 15(2)(1979).

17. Grnarov, A., Kleinrock, L., and Gerla, M., "A highly reliable distributed loop network architecture." *Proc. 1980 Int. Symp. on Fault Tolerant Computing*, pp. 319-324 (October 1980).

18. Hafner, E. R., Nenadal, Z., and Tschanz, M., "A digital loop communication system." *IEEE Trans. on Comm.* COM 22 pp. 877-881 (June 1974).

19. Halkyard, M. S. and Naik, V. K., "Simulation of buffer insertion ring local area network." Proc. GLOBCOM'83 (1983).

20. Hamacher, V. C. and Loucks, W. M., "Packet transfer delays in local area computer networks." *INFOR* 23(3) pp. 248-274 (August 1985).

21. Heap, S. and Hall, R. C., "Total network management." *Proc. Vacation School on Switching and Signalling in Telecom. Networks*, pp. 14/1-14/32 (1984).

22. Hilal, W. and Liu, M. T., "Analysis and simulation of the register insertion protocol," *Proc. 1982 Computer Networking Conf.*, pp. 91-100 IEEE, (December 1982).

23. Kermani, P. and Kleinrock, L., "Virtual cut-through : a new computer communication switching technique," *Computer Networks* 3 pp. 267-286 (September 1979).

24. Kleinrock, L., *Queueing Systems : Computer Applications*, John Wiley and Sons, New York (1976).

25. Lee, R. P., "The architecture of a dynamically reconfigurable insertion-ring network," Report RJ2485(32434)/3/13/79, IBM, San Jose, California (March, 1979).

26. Lee, R. P., "Dynamically reconfigurable logical ring network," *IBM Technical Disclosure Bulletin* 21(8) pp. 3349-3350 (January 1979).

27. Little, J., "A proof of the queueing formula $L = \lambda W$," *Operations Research* 9(2) pp. 383-387 (March 1961).

28. Liu, M. T., Babic, G., and Pardo, R., "Traffic analysis of the distributed loop computer networks (DLCN)," *Proc. Nat. Telecom. Conf.*, pp. 31:5/1-31:5/7 (December 1977).

29. Liu, M. T., Pardo, R., and Babic, G., "A performance study of distributed control loop network," *Proc. of 1977 Int'l Conf. on Parallel Processing*, pp. 137-138 (August 1977).

30. Liu, M. T., Tsay, D. P., Chou, C. P., and Li, C. M., "Design of distributed double loop computer network (DDLCN)," *J. of Digital Systems* 5(1/2) pp. 3-37 (Spring/Summer 1981).

31. Liu, M. T., Hilal, W., and Groomes, B. H., "Performance evaluation of channel access protocols for local computer networks," *Proc. COMPCON Fall'82*, pp. 417-426 IEEE, (September 1982).

32. Liu, M. T. and Rouse, D. M., "A study of ring networks," pp. 1-39 in *Ring technology local area networks,* ed. E. B. Spratt,IFIP (1984).

33. Loucks, W. M., Hamacher, V. C., Preiss, B., and Wong, L., "Short packet transfer delays in local area rings," *IEEE Trans. on Computers* C-34(11) pp. 1006-1014 (November 1985).

34. Ltd., RTD, *Clearway user guide,* Real Time Developments Ltd., Farnborough, U. K. (October 1982).

35. MacCalla, E., "peripheral networks for the small business office," *Proc. Mini-Micro West Computer Conf.,* pp. 24/1-24/6 (1983).

36. Mamrak, S. A., "The design and development of resource-sharing services in computer communications networks : A survey," pp. 183-219 in *Advances in Computers Vol. 16,* (1977).

37. Mamrak, S. A., Maurath, P., Gomez, J., Janardan, S., and Nicholas, C., "Guest layering distributed processing support on local operating systems," , pp. 854-859 IEEE, (1982).

38. Mirabella, O. and Salvo, A., *Throughput improvement in a ring LAN,* IEEE (1983).

39. Olsen, R., Seifert, W., and Taylor, J., "Tutorial: RS232C data switching on local networks," *Data Communications,* pp. 231-238 (September 1983).

40. Penny, B. K. and Baghdadi, A. A., "Survey of computer communications loop networks: Part 1," *Computer Comm.* 2(4) pp. 165-180 (August 1979).

41. Penny, B. K. and Baghdadi, A. A., "Survey of computer communications loop networks: Part 2," *Computer Comm.* 2(5) pp. 224-241 (October 1979).

42. Pierce, J. R., "Network for block switches of data," *Bell System Tech. J.* 51(6) pp. 1133-1145 (July/August 1972).

43. Pierce, R. A. and Moore, D. H., "Network operating system functions and microprocessor front-ends," *Proc. COMPCON'77 Spring,* pp. 325-328 (February 1977).

44. Popek, G., Walker, B., Kleine, C., Rudisin, G., Thiel, G., Chow, W. J., and Edwards, D., "LOCUS: A network transparent, high reliability distributed system," *Proc. of 8th SOSP,* (December 1981).

45. Raghavendra, C. S. and Gerla, M., "Optimal loop topologies for distributed systems," *Proc. 7th Data Comm. Symp.,* pp. 218-223 IEEE, (October 1981).

46. Raghavendra, C. S. and Silvester, J. A., "A survey of multiconnected loop topologies for local computer networks," *Computer networks and ISDN systems,* (11) pp. 29-42 (1986).

47. Rayner, D., "User interfaces in open data communications networks," *Proc. Working Conf. Command Languages,* pp. 405-417 IFIP, (1979).

48. Reames, C. C. and Liu, M. T., "Design and simulation of the distributed loop computer network (DLCN)," *Proc. 3rd Annual Symp. on Computer Appl.,* pp. 124-129 (January 1976).

49. Saltzer, J. H. and Pogran, K. T., "A star-shaped ring network with high maintainability," *Proc. LACN Symp.,* pp. 179-190 (May, 1979).

50. Saltzer, J. H. and Clark, D. D., and Pogran, K. T., "Why a ring ?," *Proc. 7th Data Comm. Symp.,* pp. 211-217 IEEE, (November 1981).

51. Schlatter, M., "Comparative analysis of local area communication systems," *Mitt. Angen. (Switzerland),* (pt.31) pp. 9-13 (March 1981).

52. Tannenbaum, A. S., *Computer networks,* Prentice Hall Inc. (1981).

53. Tcha, D-W. and Maruyama, K., "On the selection of primary paths for communication networks," *Computer networks and ISDN systems,* (9) pp. 257-265 (1985).

54. Thomas, P. G. and Yasin, M. M., "Enhanced management domain for low-cost local area networks," *University Computing* 8(1) pp. 46-49 (Spring 1986).

55. Thomas, P. G. and Yasin, M. M., "Performance analysis of a dynamically reconfigurable register-insertion network," Report 86/16 (1986).

56. Thomas, P. G. and Yasin, M. M., "Optimum ring formation in a dynamically reconfigurable insertion ring network," Internal Report 86/13 (1986).

57. Thomas, P. G. and Yasin, M. M., "Optimum node location in a ring LAN," *University Computing (to be published),* (Spring 1987).

58. Thomasian, A. and Kanakia, H., "Performance study of Loop networks using buffer insertion," *Computer networks* 3 pp. 419-425 (December 1979).

59. Tsay, D. P. and Liu, M. T., "Interface design for the distributed double-loop computer network (DDLCN)," *Proc. 1979 Nat. Telecom. Conf.,* pp. 59.3.1-59.3.16 (November 1979).

60. University, The Open, *M351: NCLP,* The Open University, Milton Keynes, U. K. (1985).

61. Wilbur, S., "Local area network management for distributed applications," *Computer Communications* 9(2) pp. 100-104 (April 1986).

62. Wilkes, M. V. and Needham, R. M., "The Cambridge model distributed system," *ACM Operating Systems Review* 14(1) pp. 21-29 (Jan 1980).

63. Wilson, R. J., *An introduction to graph theory,* Longman Inc. (1985).

64. Wolf, J. J., Bruce, W. W., and Liu, M. T., "Analysis and simulation of the distributed double-loop computer network," *Proc. of Computer Networking Symp.,* pp. 82-89 (1979).

APPENDICES

APPENDIX — A

Enhanced Management domain for local area networks

By   P.G. Thomas
          M.M. Yasin

Enhanced management domain for low-cost local area networks

P. G. Thomas & M. M. Yasin, *The Open University*

Abstract The computing environment of a university is characterized by its collection of different, and often incompatible equipment, and its large number of users with different computing needs and experience. Local area networks (LANs) offer low-cost access to the diverse equipment by a wide variety of user. Up to now, however, the emphasis in the design of such LANs has been on switching needs, and the network management software has been primarily concerned with input/output functions.

This paper examines the requirements for, and practicability of, extending the network management domain to include users and attached computer systems. The purpose is to add functionality to a basic message-passing network. Such functionality should be capable of being added to an existing LAN with a minimum of change and should be of low cost.

Introduction

There are many commercially available local area networks (LANs) which offer a large number of users low cost access to a variety of equipment.[1] The purpose of the work reported here is to investigate how much networks can be realistically extended to provide enhanced management services at low cost. There are two directions to the investigation. The first is to determine what is feasible without changing the underlying LAN, and the second is to investigate which of the feasible options are cost effective.

The areas which could usefully be enhanced in a LAN include:

(a) improved user-interface providing information about the characteristics and status of attached systems;
(b) uniform log-in and log-out to attached systems;
(c) network-wide mail service;
(d) file transfer;
(e) a help facility which, during an established session, does not disturb on-going communication; and
(f) establishing simultaneous parallel connections to different systems.

In the next section we outline the nature of these facilities and indicate the kinds of implementation problems they give rise to. The next section compares a centralized approach with a distributed solution. Next we describe an experimental LAN used to investigate the management issues.

Enhancing a network

User interface

Our starting point for enhancing the management domain of a LAN is the information which should be maintained by the LAN which can be of help to the individual user.[14] Simple examples include the characteristics and status of the systems attached to the network. Thus, if two or more attached systems offer equivalent services then the user can, for example,

Correspondence: Dr P. G. Thomas, Computing Discipline, Faculty of Mathematics, The Open University, Milton Keynes MK7 6AA.

choose the less heavily loaded system based on statistics held by the network. Needless to say, such choices should be easy to make, and this implies that the network should present the information in an easily digestible form.

One of the benefits claimed for LANs is that they can offer the individual access to a wide variety of computing facilities. In a heterogeneous network there will be a number of processors and peripheral devices to choose from — all with their own idiosyncratic method of access.[2,3] In particular, the log-on process can vary enormously from one machine to another. Some systems require the user to press the RETURN key to alert the system,[4] others have clever automatic baud-rate detection mechanisms.[5,6] Just as there are different log-on processes there is a bewildering assortment of log-off procedures. In an ideal world a network should provide a uniform interface to the attached devices. One way of providing such a mechanism is via user profiles. For example, if the network properly authenticates each user access to the network it could automatically log on to the target system using the user's identification and password. This mechanism would not only save the user time but would also relieve him from the need to remember the appropriate procedures for the different machines on the network.

The user profile mechanism can be put to many useful purposes. For example, a profile can check the availability of a selected target machine before connecting the user and so avoid the frustration of being (temporarily) connected to a non-responsive system.

A non-trivial problem arises when one considers the help that a LAN could give a user when logging off from an attached system. Typically, having finished a session, the user must type such a command as BYE, LOGOUT or LOGOFF to release the resources allocated to him during the session. Although some systems automatically release an inactive session after a timeout period, others do not. Even after the attached system has logged out there remains the logical connection between the user's network interface unit (NIU) and the target system's NIU, which

46

MISSING

PRINT

must be cleared so that the resources within each NIU and the terminal port of the host system may be released for use by others. If the user forgets to break this connection, some networks perform a timeout on an inactive connection, others clear a connection if they sense the user's terminal being switched off, still others retain the link until an explicit command is issued to clear-up.

An interesting situation develops when the network operates a time-out mechanism but the host system does not. In this case it is difficult to terminate an idle session once the logical connection has been cleared, as any further attempts at opening the connection to the same host will probably attach to a different host input port. (Multi-user systems identify terminal sessions by the input port from which the session has been activated. A network, however, usually assigns host input ports arbitrarily, except perhpas for matching certain characteristics such as baud-rate.)

We propose a log-out command directed to the network, together with a time-out mechanism operated by the network. The network management software takes the responsibility of appropriately closing all relevant sessions before clearing the network link.

An additional feature of the user profile is the terminal protocol translation from a network standard protocol to that of user's terminal.[7] The translation tables for all the terminal types recognized by the network would be kept at a central place, so that introduction of new terminal types is performed only once for all the network rather than being actioned on each host.

File transfer

As the network is designed to make all attached computer systems equally accessible, a user's choice of a system would be either arbitrary or based on availability of certain utilities. Although individual users tend to work exclusively on one particular system, there are other factors which influence the choice of a system for a particular session. It may be that the preferred system has a heavy load and hence a slow response, or the system may not be available at all (due to system maintenance, say). Also, there may be certain utilities which are available only on one system. For example, a user may prefer to use a particular editor on one system while programming in a language available only on another system. In all such cases, the user needs the capability to transfer files among the various systems on the network.

It must be remembered, however, that many low cost LANs are designed primarily for terminal handling, and that indiscriminate file transfer can seriously degrade performance.

Help and multiple sessions

A useful concept at this stage is the idea of a network manager. The manager is part of the network and responsible for maintaining the services offered by network. Ideally, the manager will know about:

    all current connections on the network;
    all attached systems;
    users;

and will keep statistics about the use of the network. Such a manager need not be centralized; indeed shall show that many of its functions could, should, be distributed.

It will sometimes be desirable for a user to enquire about network status or to open a session with another system without disturbing the on-going session. Thus, the network manager should be able recognize a network attention command, be able suspend a session while communicating with the user and be able to multiplex sessions on one user NIU.

Network mail

Many host systems offer a mail facility in which a user can leave messages for another. Such systems work satisfactorily when the communicating users have regular access to one machine. In a network it likely that two users may wish to communicate but not regularly use the same host. A network-wide mail system seems the obvious answer in which the network keeps track of the mail and ensures that a user receives his mail when he logs on to the network rather than having to log on to a specific host for this purpose.

Statistics

A network manager, that is, a logical resource dedicated to the management of the network, could collect various statistics for analysis of network performance and forecasting future requirements.[2,8] In particular, it could record the traffic pattern on the network, for example the average volume of data transferred between each pair of nodes over a certain period of time. This information can then be used to optimize the network configuration as discussed Thomas & Yasin.[15]

The network manager could also keep accounting information about the utilization of resources on the network.

Implementation

To implement these facilities, the network manager needs to store user profiles of all network users, peripheral profiles of all connected systems, and network mail information. It would be economical

store all this information at a central location to avoid the cost of multiple copies (if kept on each NIU), or the complexity of distributed file update (if kept in segments at various places). Such a scheme, however, creates the problem of reliability, as the operation of the network would depend on the proper functioning of the network manager. A solution to this problem is to use either stand-by managers or divide the tasks among co-operative managers — which can take over the job of the faulty manager (or, at least, partially offset the impact of its loss). Our emphasis on low-cost solutions has led us to opt for the following. The management functionality is designed as an optional add-on to the network, so that, in the event of loss of the manager, only the additional functionality is lost while the basic network remains operational, thus providing graceful degradation.

On broadcast-type networks, for example those using a bus, it is possible for a central NIU to examine every data packet on the network and take appropriate action.[9] Such a scheme is also the case with ring networks where the messages travel around the whole loop back to the source node and are then removed by the source.[10] On such networks almost all functionality can be centralized into an optional manager station for an otherwise perfectly functional network.

Our interest lies primarily in ring networks with destination removal, in which the messages travel only part of the loop from source to destination.[10] Here it is not possible for a central manager to monitor all traffic. Also, the manager cannot detect a network attention command embedded in an active session or monitor various sessions for inactivity. Such capability must be provided in each NIU, which can then refer to the central manager for further action.

Experimental set-up (Figure 1)

The Computing Discipline of the Faculty of Mathematics at the Open University has an experimental register-insertion ring network based on Multilink network interface units.[12] It incorporates destination removal of messages as in the DLCN network.[11] Multilink NIUs provide basic functions which make this network suitable for testing the ideas presented in the previous sections. Individual NIUs can open multiple simultaneous logical connections (up to 64), and thereby provide the capability of supporting multiple sessions. The NIUs can be remotely programmed. A central manager can, therefore, control and interrogate individual NIUs. There is a class concept built into the naming convention used by Multilink such that, if a particular node is unable to accept a call to open a connection, the call is passed further along the ring to any other node with the same name. Such a scheme is particularly useful for implementing co-operative network managers.
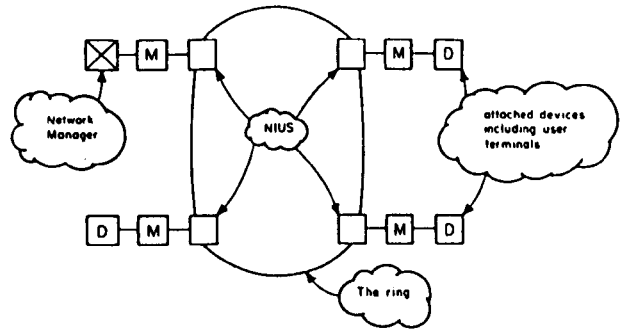

Fig. 1. Experimental set-up.

As previously discussed, a central node in a removal by destination ring is unable to monitor all the traffic on the network. Thus the detection of network attention commands and inactive sessions has to be implemented within each NIU. Since Multilink NIUs, as supplied, do not have such a capability, we included a microcomputer (an MTX512)[13] between the user terminal and the NIU to emulate the required functionality.

The MTX micro performs the following activities:

it detects network attention commands and refers them to the central manager for further action;
it performs housekeeping essential to multiplex sessions; and terminal protocol translation using a table down-loaded from the central manager;
it records statistics, in particular the volume of traffic per unit time, for various logical connections; this information is sent to the network manager at regular intervals;
it operates a time-out mechanism on inactive logical connections; and
it detects log-out commands.

In this set-up the centralized network management software provides the network's user interface, the original user interface being suppressed by the MTX microcomputers. It is also capable of providing a one-stage log-on mechanism (although some users may have reservations about the security implications of keeping user identifications and passwords at one place). When informed, by the MTX micro, either of the time-out on a logical connection or of a network log out command being issued by the user, the manager closes all relevant sessions on the host systems before clearing the network connections, thus providing the desirable effect of a clean log out.

To effect file transfer between any two computer systems, we suggest providing a file transmit/receive utility on each host. The network management software logs on to both source and destination systems and initiates the required utilities. It then remotely creates a connection between the source and destina-

tion NIUs, without being noticed by either host. The file transmit and receive utilities operate a hand-shake mechanism, so that data is not lost while appropriate connections are being set-up.

Finally, the manager collects statistics about the network usage, which can be used for forecasting network demand and optimizing network usage.

When the network manager fails, the MTX micros detect the fact and enter a transparent mode. In this mode all data to and from the user terminal is passed on to the Multilink NIU without any intermediate processing. Thus the network returns to the basic functionality provided by the Multilink network. Meanwhile, any messages sent to the network manager while it is not operational disappear into the network (in practice, any unclaimed messages are removed from the network by the source).

Discussion

We have studied the basic concepts involved in enhancing the functionality of a low cost LAN through the provision of a central manager and additional NIU functionality. We have deliberately kept the functionality provided by the MTX microcomputers to a minimum, so that this could be built into enhanced versions of the NIU. The design allows the basic operation of the network to be maintained in the event of the failure of the central manager.

There are three main directions for further research work. First, to apply the concept presented above to existing LANs, and to implement them into newer versions of respective NIUs. We have experimented only to test the practicability of the approach mentioned in this paper, further software development is needed to provide an enhanced network. Second, as shown in this paper, by providing certain minimum functionality in NIUs, low cost networks can provide extensible management capabilities. This poses the question of standardization of the required minimum LAN functionality in order to open the networks for third-party management add-ons.

Finally, further research needs to be undertaken in the use of statistics for effective resource management. We are currently studying the use of network

statistics for the optimization of DLCN type local area networks; this work is reported in another paper.[15]

References

1 Cheong, V.E. & Hirscheim, R.A. *Local Area Networks: Issues, Products and Developments* (John Wiley & Sons, 1983).

2 Cornelius, P. EURONET–DIANE USER LOGFILE: a user's structured view of the similarities and differences between the EURONET–DIANE hosts at the operating system and application system level. *Commission of European Communities Report No. EUR7878* (1983).

3 Bull, G.M. & Huckle, B.A. An interface for interactive users within a heterogeneous computer network. *Proceedings of EURO-IFIP'79 Conference* (1979), 405–409.

4 *User Reference Manual DEC-20 Introduction* (Academic Computing Service, The Open University, November 1983).

5 *Cromemco CDOS User Guide* (Cromemco Inc., 1980).

6 *Guide to IDBP (Intel Data Base Processor)* (Intel Corporation, 1983).

7 Welendez, W.A. & Petersen, E.L. Tutorial on the upper layers of the OSI Reference Model. *Commission of European Communities Report No. 84/62963* (June 1984).

8 Way, D.E. Managing a LAN. *Telecommunications* (January 1984), 79–80.

9 Thurber, K.J. & Freeman, H.A. *Tutorial: Local Computer Networks*, 2nd edn. (IEEE Computer Society Press, 1981).

10 Liu, M.T. & Rouse, D.M. A study of ring networks. *Proceedings of Ring Technology Local Area Networks Conference* (1984), 1–39.

11 Reames, C.C. & Liu, M.T. Design and simulation of distributed loop computer network (DLCN). *Proceedings of 3rd Annual Symposium on Computer Applications* (January 1976), 124–129.

12 Grant, J.S. Multilink — A ring for connecting individual outstations. *Networks '84: Proceedings of European Computer Communications Conference* (July 1984).

13 *MTX Series Opertors Manual* (MEMOTECH Ltd, 1983).

14 Rayner, D. User interface in Open Data Communications Networks. *Proceedings of Working Conference on Command Languages* (1979), 405–416.

15 Thomas, P.G. & Yasin, M.M. Optimum Node Location in a Ring LAN *Internal Report No. 86/12* (to be published).

APPENDIX — B

Optimum node location in a register-insertion network.

By   P. G. Thomas
            M. M. Yasin

To be published in University Computing,
Spring 1987.

Optimum node location in a register-insertion ring network

P.G.Thomas   and   M.M.Yasin

Computing Discipline, Faculty of Mathematics, The Open University

Milton Keynes, MK7 6AA, United Kingdom

Abstract

The Distributed Loop Computer Network (DLCN) has been the subject of several performance studies. The achievable throughput of DLCN incorporating removal by destination protocol has been shown to exceed the transmission speed of the physical medium for a balanced traffic load. It has been noted by some researchers that its performance varies with the pattern of traffic among its nodes. This paper examines the problem of locating the relative physical position of nodes in a DLCN LAN which minimises the traffic flow in the network. A formulation of the general case of n nodes is given in the form of an integer programming problem. An alternative approach is also considered. The results are verified by simulation studies.

1. Introduction

Much research and development effort in the field of local area networks (LANs) is concerned with one of two competing network topologies: bus and ring (or loop). It has been shown by various performance studies that the ring topology, using a register-insertion technique, performs better than other access mechanisms [3,4,5,7,10]. For networks using register-insertion ring technology, two mechanisms have been proposed for the removal of messages from the ring:

(i)   removal by source [13];

(ii) removal by destination [10,14].

The Distributed Loop Computer Network (DLCN), proposed by Reames and Liu [10], is an example of the latter. It permits more than one variable length message to be present on the ring by providing variable length delay-insertion registers in each node. A node wishing to transmit a message can create an empty slot on an otherwise busy ring by swithcing its own delay register into the ring. However, this increases the the size of the ring and, therefore, adversely affects the end-to-end delay characteristics of the network. This effect is bounded by limiting the maximum size of the delay register in the nodes. The finite size of an insertion register also prevents a node from monopolizing the network. A node wishing to transmit its own data must first accumulate enough space in the delay register to accomodate the messages coming from the ring.

A significant feature of DLCN is that the messages are removed from the network by the destination node. Thus during error-free operation, the

2

messages travel only part of the network - from the source node to the destination node. A consequence of this factor, which has been the subject of performance studies, is that the effective network throughput can be much greater than the transmission speed of the underlying physical medium. In the extreme case of every node transmitting only to its successor, the available throughput of a network of n nodes is n times the transmission speed.

A less researched aspect of the DLCN protocol is its sensitivity to the actual distribution of traffic among its nodes [3,5,6]. However, a study of this effect has been reported by Bux and Schlatter [4]. The latter have concluded that DCLN can provide unequal grades of service for different stations depending on their geographic location. Since messages are removed from the network by the destination node they do not travel the length of the ring. Also, only one message can travel along an arc at one time, and messages which travel long arcs consume network resources. By minimising the number of arcs between a source of data and its destination, therefore, throughput of the network can be improved.

For illustrative purposes, consider the six node ring shown in figure 1a. If the traffic from node 3 destined for node 2 is substantially higher than that in the reverse direction, and nodes 2 and 3 do not have significant communication with other nodes on the ring, then the configuration given in figure 1b is obviously advantagous than that of figure 1a.

Figure 1a & 1b

3

An alternative way of looking at this phenomenon is that, for a given traffic pattern, the performance of a DLCN ring depends on the relative location of the nodes on the ring. In this paper we develop criteria for the measurement of the performance of different configurations of nodes, and define a function which can be used to maximise the network performance. It is shown that the criterion developed can be used to configure a network to provide a substantial improvement in performance.

It is to be noted that similar reasoning can be applied to networks which use removal by source node where the destination node truncates the message, probably using the truncated message to pass on acknowledgements for the messages received earlier. Since the length of the original message is, in general, significantly larger than that of the truncated part carrying acknowledgements, the network suffers from an unbalanced load similar to that of DLCN. Thus, the optimisation discussed in the previous paragraph can improve the throughput of such a network by decreasing the number of arcs traversed by the larger messages.

2. Physical Vs Logical Topology

At this point, it is necessary to distinguish between the concepts of physical and logical topologies of a network as applied to a register-insertion ring. The logical topology of a network is the directed graph consisting of the set of nodes as vertices and the directed data paths as arcs. In the case of a register-insertion network, the logical topology is a ring or loop. Each node on the ring has exactly one predecessor and one successor. That is, in any configuration, a node will have one input arc and one output arc, such that travelling each arc once in the direction of flow

4

of data will result in a loop containing all the nodes currently in the network.

The physical topology of a network is determined by the set of the physical links between the nodes on the network. A subset of the physical links determines a logical topology. By selecting a suitable subset of the physical links a logical ring can be formed. Thus, the physical topology of a register-insertion network need not be a ring. In general, any physical network which is Hamiltonian [21] can be configured as a logical ring, using suitable routing tables in the nodes. Such an architecture has been proposed by Lee [16] to improve the reliability of the ring by providing redundant links between pairs of nodes. His Dynamically Reconfigurable Register-insertion network (DRR) consists of multiple bi-directional links among nodes and incorporates a two-entry routing table in each node. The entries show which of the attached links are the currently selected in-link and out-link. At any instance of normal operation, the set of in-link and out-link pairs forms a ring. An important feature of the DRR is that the routing tables within nodes can be set remotely by a control node. In the event of the failure of a link or a node, the control node executes an algorithm to form another edge-distinct cycle, if one exists, and sets up the appropriate routing tables.

The built-in redundancy of links implies that, in general, more than one edge distinct cycle can be formed. Each such cycle corresponds to a different configuration of relative node locations on the ring.

For example, the physical network of figure 2a consists of thirteen bi-directional links among six nodes. A selection of data paths which form a

5

logical ring is shown in figure 2b, the solid lines represent data paths whilst the physical links are shown as dotted lines. Figure 2c shows an alternative selection of data paths, thus forming a different relative configuration of nodes on the ring for the same physical network.

Figure 2a, 2b & 2c

This flexibility of ring configuration, coupled with the fact that the network throughput varies significantly with the traffic pattern among the nodes, gives rise to the problem of finding the optimum ring formation which maximises the network performance for a given traffic pattern. A new ring formation may be needed not only because of link or node failures, but also if there is a significant change in the traffic pattern on the network.

3. Abstraction of the problem

The purpose of the study is to develop a method for finding the optimum relative node location in a ring LAN for a given traffic pattern. To simplify the problem, we assume that an arbitrary logical ring topology can be chosen as a feasible solution. This implies that either the individual nodes can be physically moved to adjust to the solution, or the physical topology of the network is a fully connected network. A subsequent paper [18] examines the effect of additional constraints which arise when a less than fully connected network is considered.

6

Suppose that the average amount of data transmitted across the network between every pair of nodes in the network is known (perhaps measured in bits per unit time). The problem is to determine the relative positions of nodes which minimises the traffic across the arcs of the ring <u>assuming that</u> <u>there are no geographical constraints on the position of any node.</u>

Let v_{ij} denote the average volume of traffic from node i to node j during unit time, i,j = 1,2,...,n, where n is the number of nodes in the network. Further, let l_{ij} be the distance, measured by the number of arcs between node i and node j, in the direction of traffic flow on the ring. Then, the work done by the network in transporting the traffic between node i and node j is the product $v_{ij}l_{ij}$. We can now say that the optimum node location is that which minimises the total work done by the network. That is, we want to optimise the following objective, or cost, function:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} v_{ij}l_{ij} \qquad (1)$$

[Our analysis of the dynamically reconfigurable register-insertion ring network [19] shows that this sum represents the aggregate through traffic on the network. In [19] we show that minimzing the cost function described above also decreases the mean message transfer delay of the network.]

Using node 1 as a reference node, let x_i be the distance, measured by the number of arcs, between node 1 and node i, in the direction of movement of data on the ring, and let d_{ij} be a 0-1 variable such that:

7

$$d_{ij} = \begin{cases} 0 & \text{when } x_j >= x_i \\ 1 & \text{when } x_j < x_i \end{cases} \tag{2}$$

for which, by definition,

$$d_{1j} = 0 \quad \text{and} \quad d_{i1} = 1 \qquad i,j = 2,..,n \tag{3}$$

This is a way of saying that every node (excluding the reference node) comes after node 1 in the direction of movement of data on the ring. Since we are considering a ring topology, the definition of the d_{ij} leads to the relationship:

$$d_{ij} = 1 - d_{ji} \qquad \text{for all } i,j, \ i \neq j \tag{4}$$

and

$$l_{ij} = \begin{cases} x_j - x_i & \text{when } x_j > x_i \\ x_j - x_i + n & \text{when } x_j < x_i \end{cases}$$

Hence,

$$l_{ij} = x_j - x_i + nd_{ij} \qquad (i \neq j) \tag{5}$$

and the objective function (1) can be written:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} v_{ij}(x_j - x_i + nd_{ij}) \tag{6}$$
$$\boxed{j \neq i}$$

or, alternatively as:

$$\min \quad \underline{u}^T(\ (v - v^T) \ \underline{x} + n \ v_D \ \underline{u} \ ) \tag{7}$$

8

where V is the (n * n) matrix whose elements are v_{ij}, and where we define $v_{ii}= 0$. Here, $\underline{x}$ is the vector of distances $[x_1, x_2, \ldots x_n]^T$, $\underline{u}$ is the unit vector $[1, 1, \ldots 1]^T$ and V_D is the matrix whose (ij) element is the product $v_{ij} * d_{ij}$. We shall also require the matrix D whose (ij) element is d_{ij}.

As d_{ij} is 1 when node j lies before node i in the direction of data flow $(x_j < x_i)$, the number of arcs between node 1 and node i (x_i) is given by:

$$x_i = \sum_{j=1}^{n} d_{ij} \qquad \Rightarrow \quad \underline{x} = D\,\underline{u} \qquad\qquad (8)$$

Substituting (8) into (7) yields:

$$\underline{u}^T \left((V - V^T) D + n\, V_D \right) \underline{u} \qquad\qquad (9)$$

If we now define

$$A_{ij} = v_{ij} - v_{ji} \qquad \Rightarrow \quad A = V - V^T \qquad\qquad (10)$$

then (9) simplifies to

$$\underline{u}^T \left(A\, D + n\, V_D \right) \underline{u} \qquad\qquad (11)$$

9

Expression (11) can be changed into a more useful form. In what follows we shall use the notation $\bar{M}$ to stand for the strictly upper triangular part of the matrix M. Thus, we can write

$$\underline{u}^T V_D \underline{u} = \underline{u}^T (\bar{A}_D + \bar{V}) \underline{u}$$

However, $\underline{u}^T \bar{V} \underline{u}$ is a constant independent of d_{ij}, and thus expression (11) becomes

$$\underline{u}^T (A D + n \bar{A}_D) \underline{u} \qquad (12)$$

Equation (4) implies

$$D = \bar{D} - \overline{D^T} + \{ \overline{(u u^T)} \}^T$$

so that the first term of (11) can be re-written as

$$\underline{u}^T A D \underline{u} = \overline{\underline{u}^T} A (\bar{D} - \overline{D^T}) \underline{u} + \underline{u}^T A \{ \overline{(u u^T)} \}^T \underline{u} \qquad (13)$$

Also,

$$\underline{u}^T A \overline{D^T} \underline{u} = \underline{u}^T \bar{D} A^T \underline{u} \qquad (14)$$

Using (14) and (13) in (12), and noting that the last term of (13) is a constant, yields

$$\underline{u}^T \{ (A - A^T) \bar{D} + n \bar{A}_D \} \underline{u} \qquad (15)$$

10

as the objective or cost function to be minimised. The co-efficient of d_{ij}

for $j>i$, is easily found to be

$$\left(\sum_{k=1}^{n} A_{ki} - \sum_{k=1}^{n} A_{kj}\right) + n \, A_{ij} \tag{16}$$

Expression (15) defines the objective function to be minimised to obtain the optimum connectivity of a network with destination removal and unidirectional links. The following example illustrates the use of expression (15) but shows that it does not completely solve our problem and that constraints must be added.

Example 1

Suppose that the following matrix, V, represents the traffic pattern on some four node network. The elements of the matrix have been chosen deliberately to be simple; there is no loss of generality since the elements are supposed known and are, therefore, constants of the problem. In this example the non-zero elements can be viewed as representing a high density of traffic.

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Here, v_{ij} represent the data flowing between node $\underline{i}$ and node $\underline{j}$, in some suitable units. From equation (10), we get the matrix A:

11

$$\begin{bmatrix} 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & -1 \\ 0 & -1 & 0 & 1 \\ 1 & 1 & -1 & 0 \end{bmatrix}$$

For a four node network, the objective function (15) becomes

$$\min \sum_{i=1}^{4} \sum_{j=1}^{4} \left(\sum_{k=1}^{4} A_{ki} - \sum_{k=1}^{4} A_{kj} + 4A_{ij} \right) d_{ij}$$

which leads to

$$\min \quad 5d_{23} - 2d_{24} + 5d_{34} \tag{17}$$

As the d_{ij} are 0-1 variables, the solution to this objective function is easily seen to be

$$d_{23} = d_{34} = 0 \quad \text{and} \quad d_{24} = 1 \tag{18}$$

(Recall that $d_{i1} = 1$ and $d_{1j} = 0$ are already known - by (3)). The interpretation of the solution is that:

node 3 comes after node 2 (since $d_{23} = 0$)

node 4 comes after node 3 (since $d_{34} = 0$)

node 4 comes before node 2 (since $d_{24} = 1$)

Pictorially this can be viewed as:

Figure 3

----

which is clearly not a ring. That the solution does not represent a ring can also be deduced from the values of the x_i's given by equation (8):

$$x_2 = d_{21} + d_{23} + d_{24} = 2$$

$$x_3 = d_{31} + d_{32} + d_{34} = 2$$

$$x_4 = d_{41} + d_{42} + d_{43} = 2$$

which seems to imply that each of the nodes 2,3 and 4 are two arcs away from node 1! The definition of the x_i's breaks down if the network is not a ring.

This example illustrates that contraints have to be added to the objective function to enforce a ring solution.

4. Constraints

There are several ways in which constraints can be added to the formulation. Our first way comes from the observation that all the x_i have to be distinct (i.e. have different, integer values), otherwise two or more nodes will be at the same distance from node 1. We can formulate this requirement by demanding that the distance between any two nodes be at least one:

$$x_j - x_i + nd_{ij} \geq 1 \qquad \text{for all } i,j, \; i \neq j \tag{19}$$

and that the distances from 1 to each node take the integer values from 1 to n inclusive:

$$x_j - x_i + nd_{ij} \leq n \tag{20}$$

Once again we use equation (8) to replace the x's in (19) and (20):

$$\sum_{k=1}^{n} d_{jk} - \sum_{k=1}^{n} d_{ik} + nd_{ij} \geq 1 \qquad j > i \tag{21}$$

$$\sum_{k=1}^{n} d_{jk} - \sum_{k=1}^{n} d_{ik} + nd_{ij} \leq n \qquad j > i \tag{22}$$

An alternative way of enforcing a ring architecture upon the solution is to impose the transitive relation:

$$\text{if} \quad d_{ij} = 0 \quad \text{and} \quad d_{jk} = 0 \quad \text{then} \quad d_{ik} = 0 \quad \text{for all } i,j,k \tag{23}$$

5. Methods of solution and results

For a network of many nodes, together with a more general volume matrix, a systematic method of solution is required. The formulation given above is clearly an integer programming problem with objective function (15) subject to the constraints (21) and (22) where the variables, the d_{ij}, are all 0-1 integer variables.

14

Our first approach to the solution of this integer programming problem was
to utilize a linear programming package named NCLP [12] running on a DEC-
system 20 computer. A pre-processor was written in Pascal to produce input
compatible with the sparse input mode of NCLP from a given volume matrix. A
post-processor used the solution from NCLP to form the solution ring
configuration and compute the total cost according to equations (6) and (8).
This method, when applied to Example 1 gives the following solution:

$$d_{23} = d_{24} = d_{34} = 0$$

with a cost of 6 units.

Example 2

The NCLP package was used to find the optimum node configuration of a
register insertion ring LAN with seven nodes labelled 1 to 7, and the
following volume matrix:

$$
\begin{pmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
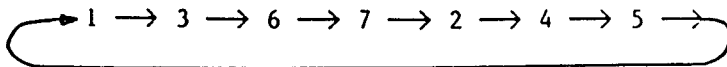0 & 1 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
$$

15

The objective, produced by the pre-processor, is to minimise

$$7d_{24} - d_{26} - 7d_{27} + 6d_{36} + 7d_{45} - d_{46} + 6d_{56} + 8d_{67} \qquad (24)$$
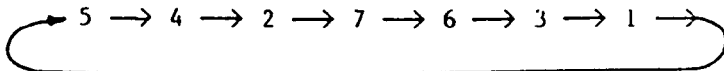
The solution is:

$$x_2 = 4, \quad x_3 = 1, \quad x_4 = 5, \quad x_5 = 6, \quad x_6 = 2, \quad x_7 = 3$$

with ring configuration

and a total cost of 15 units. For comparison purposes, the post-processor also gives the worst-case configuration:

with total cost of 40 units.

Since an integer programming package can be expensive in computer time especially for networks with greater than 10 nodes say, an alternative method of solution was sought. We have developed a package called CONFIG which uses a set theoretic approach. Here, the coefficients of the objective function are taken in order of absolute value, highest first, to determine the polarity of the corresponding d_{ij}.

To illustrate the approach, consider the problem of minimizing function (17) of Example 1. Here, d_{23} and d_{34} both have the maximum modulus coefficient of

5. Arbitrarily choosing d_{23} first, it is set to zero to minimise its effect. Next d_{34} is examined and again is (temporarily) set to zero. Having allocated values to both d_{23} and d_{34} it is possible to deduce, using the transitive constraint (23), that d_{24} should also be set to zero (and implying that d_{42} is 1). The next step is to examine the coefficient of d_{24} in the objective: as this coefficient is negative we would ideally like to set d_{24} to 1. However, this conflicts with the value of d_{24} already demanded by the setting of d_{23} and d_{34} earlier. To maintain d_{24} at zero means that the value of the objective is increased by 2 - the absolute value of the coefficient of d_{24}. However, to set d_{24} to 1 means that either d_{23} or d_{34} must be changed to 1. In either case the value of the objective would be raised by 5. Therefore, we conclude that the smallest increase in the value of the objective is when d_{24} is zero. As all the d_{ij} in the objective have now been determined the calculation is complete.

This process attempts to ensure that the minimisation takes maximum account of the largest coefficient. The d_{ij} with smaller coefficients are "adjusted", if possible, to conform to the demands of the larger coefficients. It is possible, of course, that this process will, at some stage, require the re-examination of all previous decisions and would, therefore, lead to lengthy computations.

17

Example 3

This example, of an eight node ring, illustrates the use of CONFIG on a more substantial example. Here is the volume matrix:

$$
\begin{pmatrix}
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 & 0
\end{pmatrix}
$$

The objective is to minimise:

$$2d_{23} - 3d_{24} - 4d_{25} + 7d_{26} + 3d_{27} - 3d_{28} - 5d_{34} + 10d_{35} - 3d_{36} + d_{37}$$

$$- 5d_{38} - d_{45} - 6d_{46} + 6d_{47} - 8d_{48} - 5d_{56} - d_{57} + 9d_{58} + 4d_{67} - 10d_{68}$$

$$+ 10d_{78}$$

The solution is:

$$x_2 = 3, \quad x_3 = 1, \quad x_4 = 7, \quad x_5 = 2, \quad x_6 = 6, \quad x_7 = 4, \quad x_8 = 5$$

with a total cost of 72 units. The worst case cost is 104 units.

18

6. Simulation studies

Simulation is a useful technique in the investigation of network performance. We have employed this technique to confirm the results of the previous section. We simulated the rings used in examples 2 and 3 to study the behaviour of the extreme case configurations under different load conditions. We have assumed poisson arrival of external messages at the nodes. The arrival rate at node i is determined by the ratio of the number of 1's in row i of the respective volume matrix to the total number of 1's in that matrix. The probability that a message originating at node i is destined for node j is determined by the value of the volume element (ij) divided by the sum of all elements of row i. The message length is assumed to be exponentially distributed with mean 500 bits. Each message is appended by a 24 bit header. Figures 4a and 4b show the results of our simulation. The graphs show the average message transfer time between source and destination under varying traffic loadings.

Figures 4a and 4b

Our results show that there is a considerable improvement in performance of the optimised configuration. It is also clear that the optimised configuration is more stable under higher loads.

7. Conclusions

The linear programming package NCLP is designed for small problems and we could only test out LAN configurations for up to seven nodes. However, even with small systems it was apparent that the computational efficiency of such a solution could be prohibitive for real systems. However, the value of this approach is:

(i) it will yield a solution to the problem;

(ii) it can be used to check other methods of solution.

For larger networks, we propose using mainframe integer programming packages such as MPSX [20]. Also, combining parts of the network into pseudo-nodes with combined input and output traffic, can decrease the size of the problem into manageable sub-problems. These pseudo-nodes can be optimised individually as small rings, viewing the rest of the network as a pseudo-node which is the source of all data into the sub-network and the sink of all data out of the sub-network. Of course, such a strategy may not lead to the global optimum solution.

Although the CONFIG approach is simpler and more efficient, it is applicable only to the case of fully connected networks. Even then, it breaks down if there are several local minima.

Our results, confirmed by simulation studies, indicate that there are benefits to be gained from optimizing ring configurations. That such a course of action should be embarked upon can be seen in the newly emerging dynamically reconfigurable network topologies [5,15,17].

20

Acknowledgement

21

References

[1] Liu, M T and Rouse, D M, A study of Ring Networks, Proc of Ring Technology LANs, IFIP, 1984, 1-39.

[2] Ragavendra, C S and Silvester, A, Double loop network architectures - A Performance Study, IEEE Trans on Comm, vol COM 33, 2, 185-187, 1985.

[3] Liu M T, Babic G, Pardo R, Traffic Analysis of DLCN, Proc of Nat Telecom Conf, 1977, 31:5-1/7.

[4] Bux, W and Schlatter, M, An approximate method for the performance analysis of buffer-insertion rings, IEEE Trans on Comm, vol COM 31, 1, 50-55, 1983.

[5] Dixon R C, Ring Network Topology for Local Data Communications, Proc COMPCON Fall'82, Sept 1982, 591-605.

[6] Mirabella, O and Salvo, A, Throughput Improvement in a Ring LAN, IEEE 1983.

[7] Hilal W and Liu M T, Analysis and Simulation of Register Insertion Protocol, Proc of 1982 Comp Networking Symp, Dec 1982, 91-100.

[8] Liu M T and Hilal W, Performance Evaluation of Channel Access Protocols for LCNs, Proc COMPCON Fall 82, Sept 1982, 417-426.

[9] Thomasian A and Kanakia H, Performance Study of Loop Networks using buffer Insertion, Computer networks Vol 3, Dec 1979, 419-425.

[10] Reames C C and Liu M T, Design and Simulation of DLCN, Proc of 3rd Ann Symp on Comp Appl, Jan 1976, 124-129.

[11] Goscinski A, Walasek T, Zelinski K, Performance Study of Ring Structured LANs using Register Insertion, Angus Int No.3, March 1985, 120-126.

[12] M351, Numerical Computation, The Open University Press, 1985.

22

[13] Hafner, E R, Nenadal, Z, and Tschanz, M, A Digital Loop Communication
    System, IEEE Trans On Comms, vol COM 22, June 1974, 877-881.

[14] Grant, J S, Multilink - An Open Network For Personal Computers, Comp
    Comms, vol 8, 1, 27-34.

[15] Saltzer, J H and Pogran, K, A Star Shaped Ring Network with High
    Maintainability, Proc Local Area Comms Network Symp, May 1979, 179-190.

[16] Lee, R P, The architecture of a dynamically reconfigurable
    insertion-ing network, IBM Research Report RJ2485(32434)/3/13/79.

[17] Thomas, P G and Yasin, M M, Enhanced Management domain for low-cost
    local area networks, University Computing, Vol 8, 00-00, 1986.

[18] Thomas, P G and Yasin, M M, Optimum Ring Formation In a Dynamically
    Reconfigurable Insetion ring network, Internal report 86/13,
    (To be published)

[19] Thomas, P G and Yasin, M M, Performance Analysis of the Dynamically
    Reconfigurable Insertion ring network, (In preparation).

[20] ----, Introduction to Mathematical Programming System - Extended (MPSX)
    Mixed Integer Programming (MIP) and Generalized Upper Bounding (GUB),
    IBM Corp., 4th Ed., 1973.

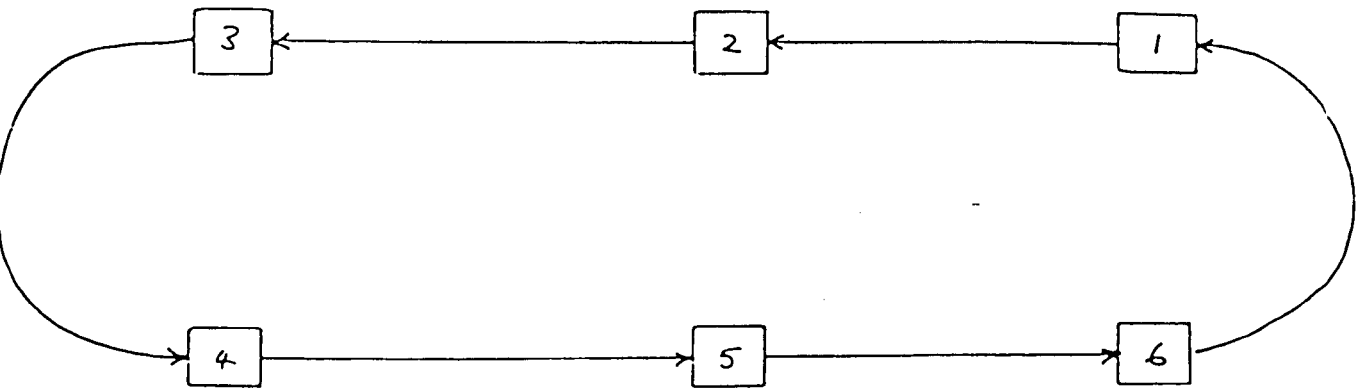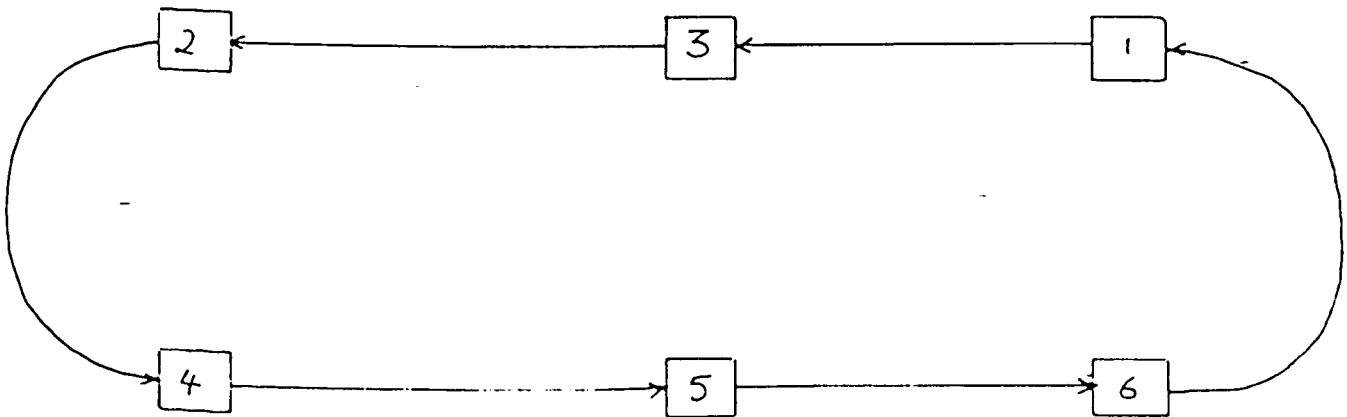[21] Wilson, R J, Introduction to Graph Theory, 3rd Ed.   -
    Longman, 1985, 35-38.
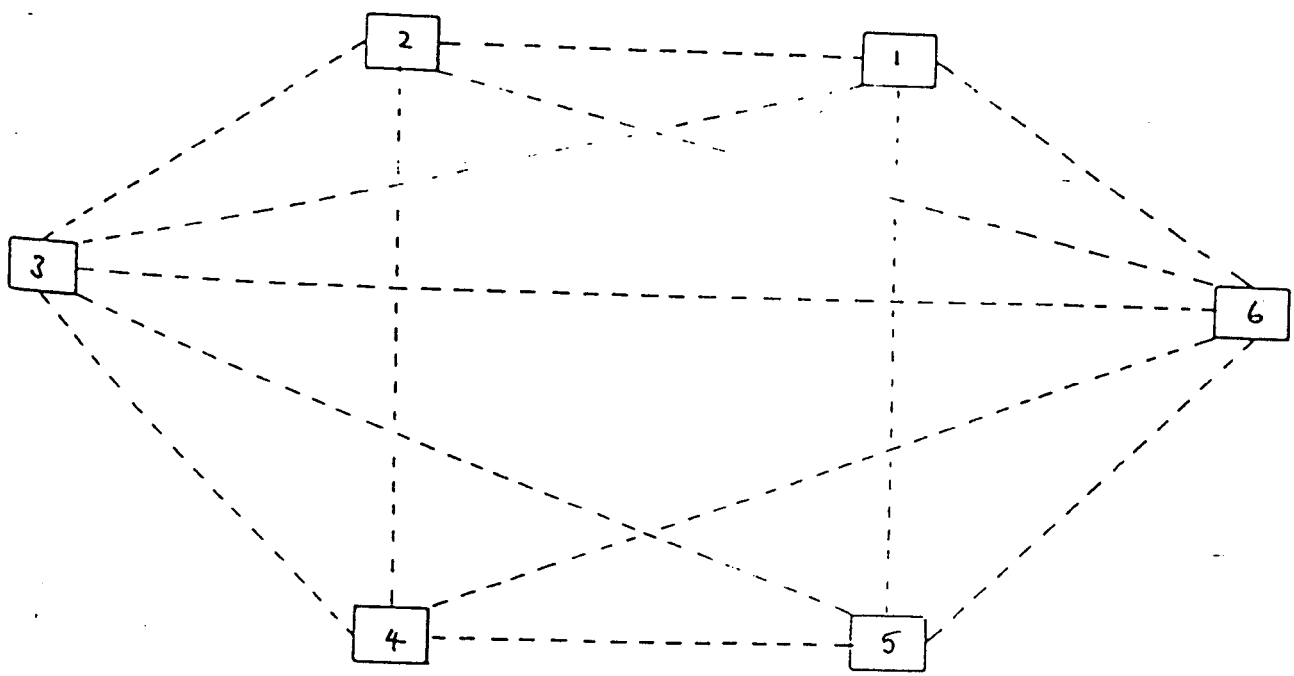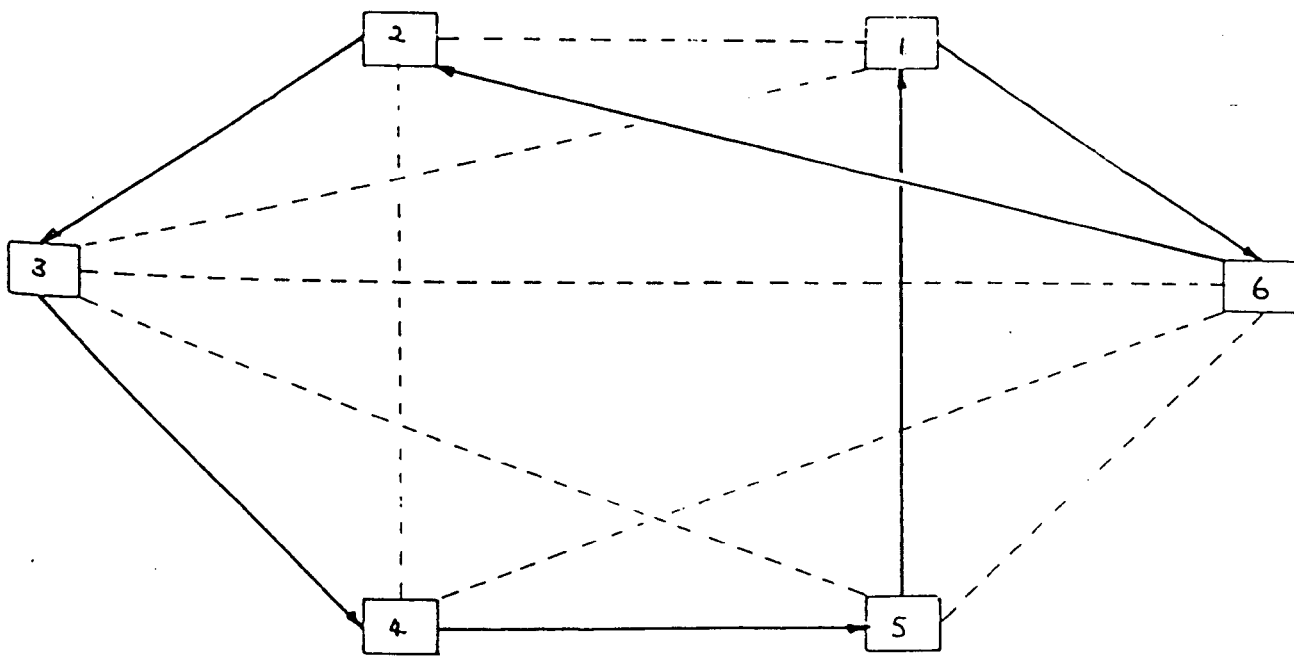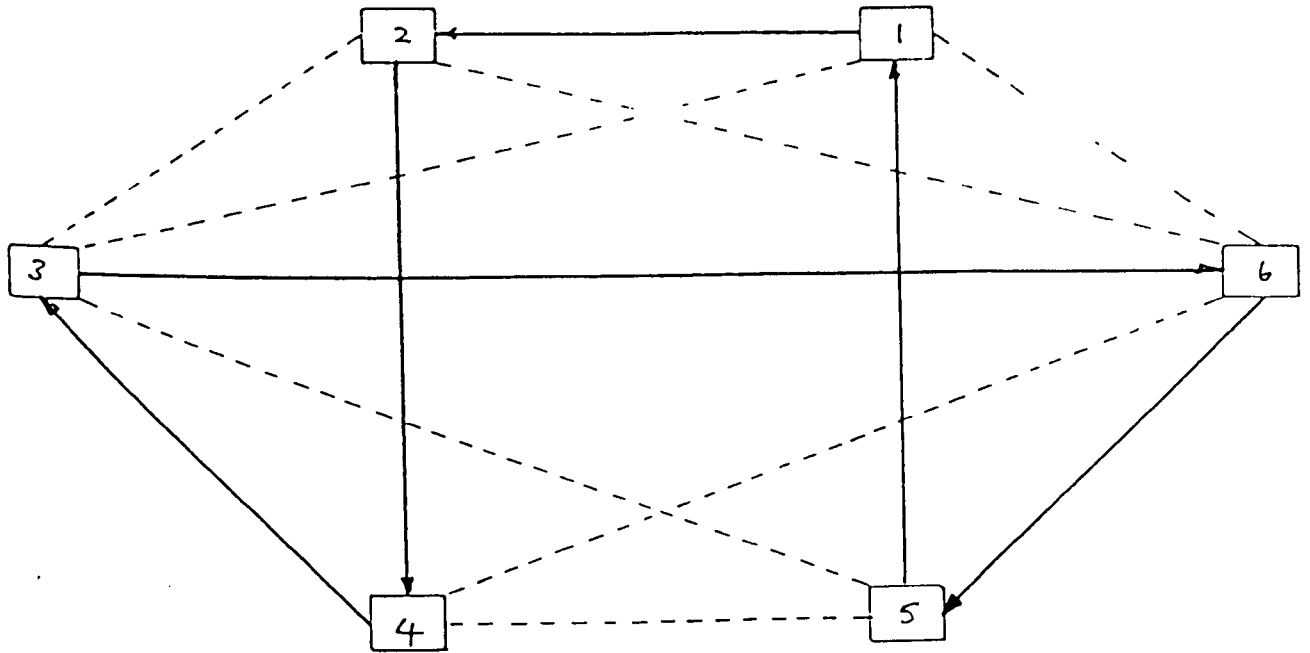
23

Figure 1a.

Figure 1b.

Figure 2a

Figure 26

Figure 2c
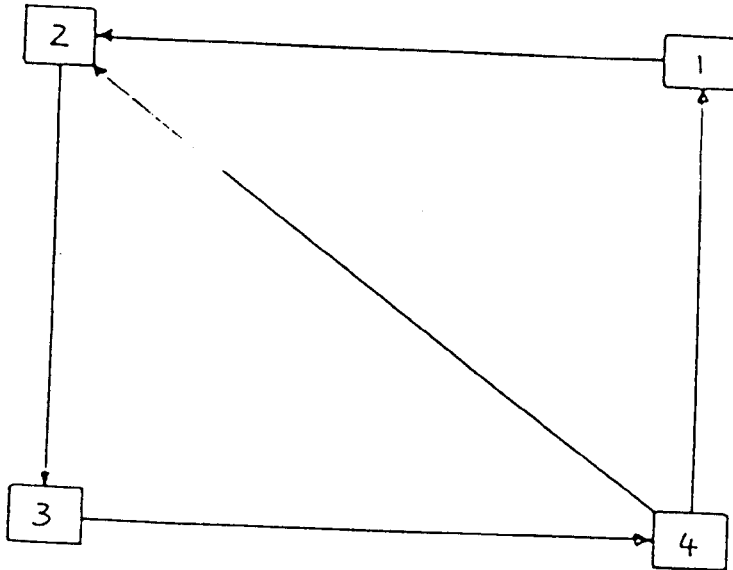
Figure 3

Figure 4a

Simulation of Example 2

_- 143 -_

Figure 46

Simulation of Example 3

MEAN TRANSFER TIME (ms)

LOAD
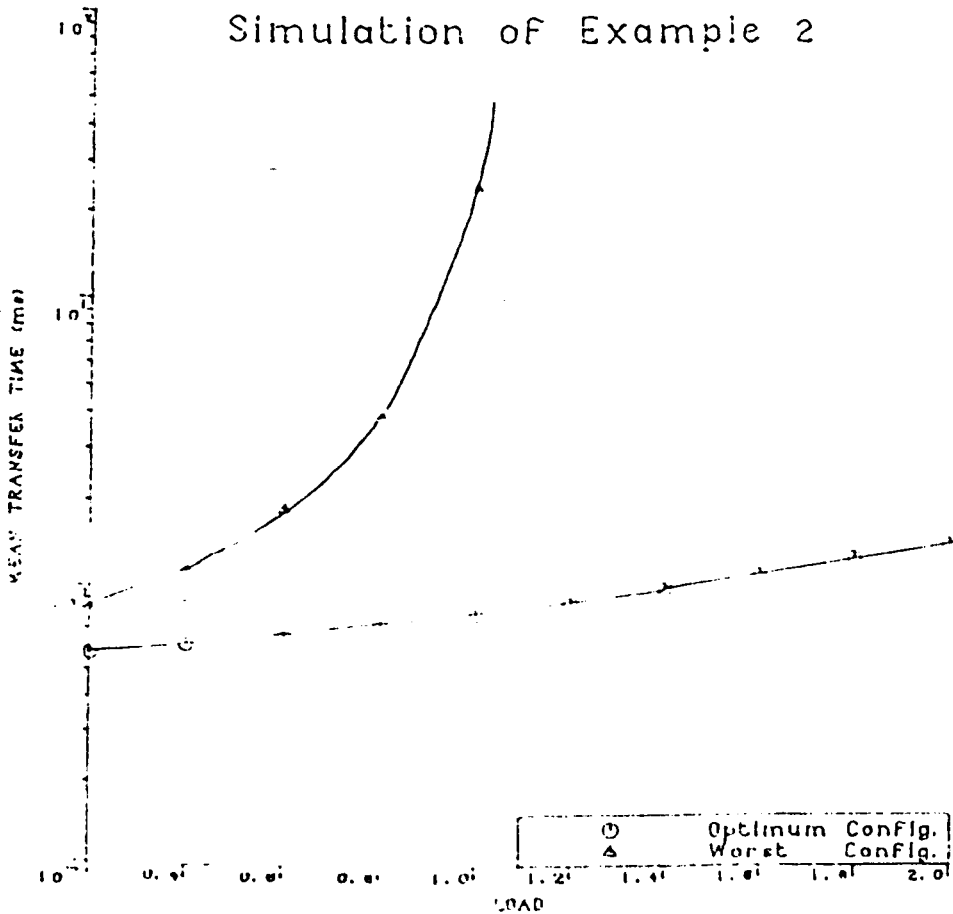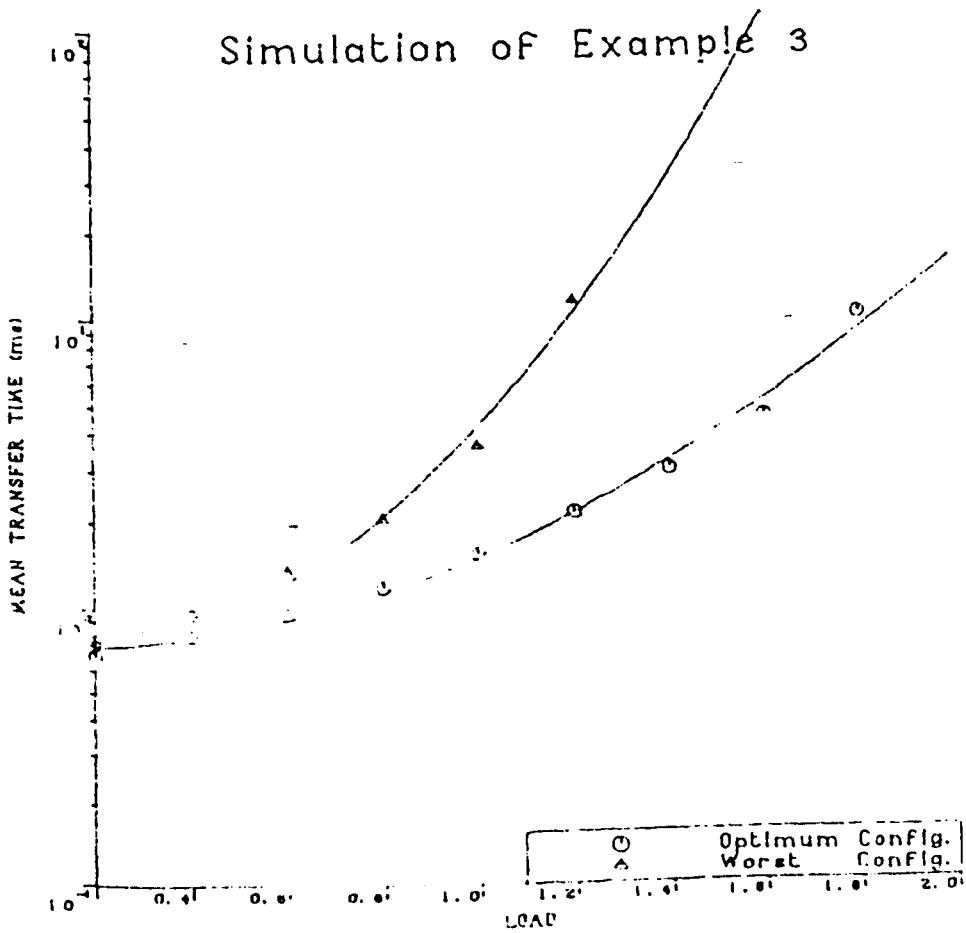
Optimum Config.
Worst Config.

APPENDIX — C

Optimum ring formation in a dynamically reconfigurable insertion ring network

By   P. G. Thomas

M. M. Yasin

Optimum ring formation in a dynamically reconfigurable insertion ring network

P.G.Thomas and M.M.Yasin

*Computing Discipline, Faculty of Mathematics, The Open University,
Milton Keynes, U.K.*

Abstract

A delay-insertion ring network with removal by destination protocol exhibits sensitivity to the relative location of its nodes. A dynamically reconfigurable insertion ring network has a physical topology of a general Hamiltonian graph, where one or more logical rings may be configured by setting up appropriate routing tables in the nodes. This paper extends the notion of optimum relative node location developed in [8] to forming the optimum ring configuration for such a network.

1. Introduction

Local area computer networks (LANs) provide a high bandwidth communications channel for the interconnection of computers and user terminals spread over a limited geographic area. The ring topology for these networks is of considerable interest because it offers low cost communication.

A ring LAN provides a uni-directional flow of data among its nodes. Each node has exactly one in-coming data link (the in-link) and one out-going data link (the out-link). A node continually re-transmits data received on its in-link onto its out-link, thereby eliminating any routing problems which exist for computer networks with greater connectivity. Exceptions to this normal mode of operation occur when either a node wishes to transmit its own data onto the ring, or data on the ring reaches its destination node. These exceptions are resolved by the transmission and removal policies of the LAN.

Three common transmission policies are :-

(1)   the slotted ring,

(2)   the token ring,

(3)   the insertion ring.

In this paper we are concerned with insertion rings which have certain advantages over the other two policies.

2

In a slotted ring [1] one or more empty slots of fixed data length traverse the ring, each preceded by a full/empty flag. A node wishing to transmit waits for an empty slot to arrive on its in-link, changes the header flag to indicate full status, and fills the slot with its own data. Since the slot(s) on the ring are of fixed length, there is a computational overhead of dividing the message into packets at the source node and re-assembling the packets at the destination node. There is also a waste of channel capacity because the slot containing the last part of a message may be partly empty.

A token ring [2] accommodates variable length messages by using a token to permit transmission. A token circulates around the ring when the ring is idle. A node receiving the token passess it onto its successor, unless it has its own data to send. In the latter case, the node first transmits its own data and then releases the token to its successor. Thus, only one node can be transmitting on a token ring at any time.

An insertion ring [3] permits more than one variable length messages to be present on the ring by providing delay-insertion registers in each node. A node wishing to transmit can create an empty slot on an otherwise busy ring by switching its own delay register into the ring. However, this increases the size of the ring and, therefore, adversely affects the end-to-end delay characteristics of the network. This effect is bounded by limiting the maximum length of the delay registers in the nodes.

3

There are two commonly used alternative policies for the removal of messages from a ring during fault free operation. In the first policy, the source node removes its message form the ring, and this is known as removal by source. Here, messages travel completely around the ring, arriving back at the source to be removed. In the second method, namely removal by destination, the destination node removes the message from the ring. Here the destination node can either delay all messages whilst checking the destination address fields [5], or it can perform the address checking 'on the fly' by, (i) truncating the rest of the message as soon as it recognises its own address in the destination field, and (ii) relying on another node to remove the truncated message from the ring [6]. There are variations on these two techniques, including the use of the truncated message for sending acknowledgements of previously received messages [4].

Thus, on an insertion ring network using the removal by destination protocol variable length messages travel only part of the loop (from the source node to the destination node). By minimising the number of nodes lying between a source of data and its destination, throughput of the network can be improved.

A major problem with a single loop network is its vulnerability to node and link failures. For example, failure of a single link can disrupt the functioning of the whole network. There have been several approaches to increasing the reliability of such a network by using: a by-pass switch [3], double loops [10,11], and the physical topology of a Hamiltonian graph [4].

4

In the last mentioned approach, each of the nodes has several links attached to it such that one or more edge distinct cycles may be formed. Each node has a routing table with two entries. The entries show which of the attached links are the currently selected in-link and out-link. At any instance of normal operation, the set of in-link and out-link pairs forms a ring. In the event of the failure of a link or a node, a control node runs an algorithm to form another edge distinct cycle, if one exists, and sets up the appropriate routing tables.

Arising out of the discussion on the collection of network statistics in [7], it is shown in [8] that the performance of an insertion ring with destination removal can be improved for a given traffic pattern by selecting the correct relative node locations on the ring. In the present paper, we generalise this concept to finding an optimum ring configuration in a Hamiltonian network described in the previous paragraph.

2. The Optimization Problem

Consider a ring of n nodes numbered 1 to n, for which a volume matrix V is known. The elements $v_{i,j}$ of V denote the volume of data transferred from node i to node j in unit time. Let $l_{i,j}$ be the distance between node i and node j measured by the number of arcs in the direction of data flow on the ring. We define work done by the network in transferring this traffic per unit time as the following product-sum:-

5

$$\sum_{i,j} v_{i,j} * l_{i,j} \qquad\qquad (1)$$

Using node 1 as the reference node, let x_i be the distance $l_{1,i}$, and let $d_{i,j}$ be a 0-1 variable such that

$$d_{i,j} = \begin{cases} 0 & \text{when } x_j >= x_i \\ 1 & \text{when } x_j < x_i \end{cases} \qquad\qquad (2)$$

In [8], it is shown that

$$l_{i,j} = x_j - x_i + n * d_{i,j} \qquad (i \neq j) \qquad\qquad (3)$$

and that the objective function to be optimised is:

$$\text{minimize } \underline{u}^T \{ (A - A^T) \bar{D} + n \bar{A}_D \} \underline{u} \qquad\qquad (4)$$

where

$\underline{u}$ is the unit vector $[1,1,\ldots\ldots,1]^T$,

$A = V - V^T$,

D is the matrix of elements $d_{i,j}$,

A_D is the matrix whose (i,j) element is the product $a_{i,j} * d_{i,j}$

and where the notation $\bar{M}$ stands for the matrix formed from the strictly upper triangular part of the matrix M.

6

It is also shown in [8] that, in order for the resultant topology to be a ring, the solution must be constrained by the following :

$$x_j - x_i + n * d_{i,j} >= 1 \qquad \text{for all } i,j \ (i \neq j) \qquad (5)$$

and

$$x_j - x_i + n * d_{i,j} < n \qquad \text{for all } i,j \ (i \neq j) \qquad (6)$$

The above constraints specify that the distance between any two nodes is at least 1 and not greater than n.

These constraints assume that any two nodes can be placed adjacent to each other on the ring. This implies that either the nodes can be physically moved or a physical link exists between each pair of nodes. In the latter case the physical topology is a fully connected network.

In the rest of this paper, we shall consider Hamiltonian networks which are less than fully connected.

7

3. Symmetric Multiplicity

Consider a set of n nodes numbered 1 to n. We define a network with multiplicity m as one in which each node has a link to its m consecutive successors, and by symmetry, to m of its consecutive predecessors. Hence, a network with multiplicity 1 is a single loop. Figure 1 shows examples of networks with multiplicities 2 and 3.

In a network which is less than fully connected, there exist pairs of nodes which do not have a direct link between them. Suppose that nodes i and j are such a pair. The distance between node i and node j must be at least 2, since there must be at least one intermediate node. Also, the maximum possible distance between node i and node j reduces to n-1 by similar reasoning. Thus, constraints (5 and 6) must be modified to:

$$x_j - x_i + n * d_{i,j} >= 2 \qquad (7)$$

and

$$- x_j - x_i + n * d_{i,j} <= n-1 \qquad (8)$$

Next, consider a network with multiplicity m. Here, each node i has a direct link to nodes i+1, i+2, ..., (i+m), all additions taken modulo n. Also, node i has a direct link to nodes i-1, i-2, ..., (i-m) modulo n. Thus constraints (5 and 6) are applicable to all such pairs of nodes and lead to:

8

$$x_j - x_i + n * d_{i,j} >= 1 \qquad \text{for } (j-i+n) \bmod n <= m \qquad (9)$$

and

$$x_j - x_i + n * d_{i,j} <= n \qquad \text{for } (j-i+n) \bmod n <= m \qquad (10)$$

For all other pairs of nodes, constraints (7 and 8) apply:

$$x_j - x_i + n * d_{i,j} >= 2 \qquad \text{for } (j-i+n) \bmod n > m \qquad (11)$$

and

$$x_j - x_i + n * d_{i,j} <= n-1 \qquad \text{for } (j-i+n) \bmod n > m \qquad (12)$$

Finally, consider the more general case where the multiple links have arbitrary hop lengths. That is, node i is directly connected to nodes (i±1), (i±h_1), (i±h_2), . . . , (i±h_m) mod n. Here, the constraint pair (5 and 6) applies to the set of pairs of nodes (i,j) such that { j = i±1 or i±h_1 or i±h_2 or ... i±h_m all mod n }, and the constraint pair (7 and 8) apply to all other pairs of nodes.

9

4. Faults and Additional Links

Any missing links, caused by link failure or otherwise, can be accommodated in this model by choosing constraints (7 and 8) for the appropriate pairs of nodes. Also, links which are additional to the symmetric multiplicity described above can be included by choosing constraints (5 and 6) for nodes connected by these links.

5. Results

We have used an integer programming package NCLP [12] to solve a variety of problems. The following examples illustrate the effect of the multiplicity constraints.

In the examples we look at a network of six nodes with the following volume matrix:

$$
\begin{vmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0
\end{vmatrix}
$$

where an entry of 1 indicates heavy traffic and 0 indicates otherwise.

10

Example 1

For a fully connected network with six nodes, that is, with multiplicity m=3, the optimum configuration is found to be:

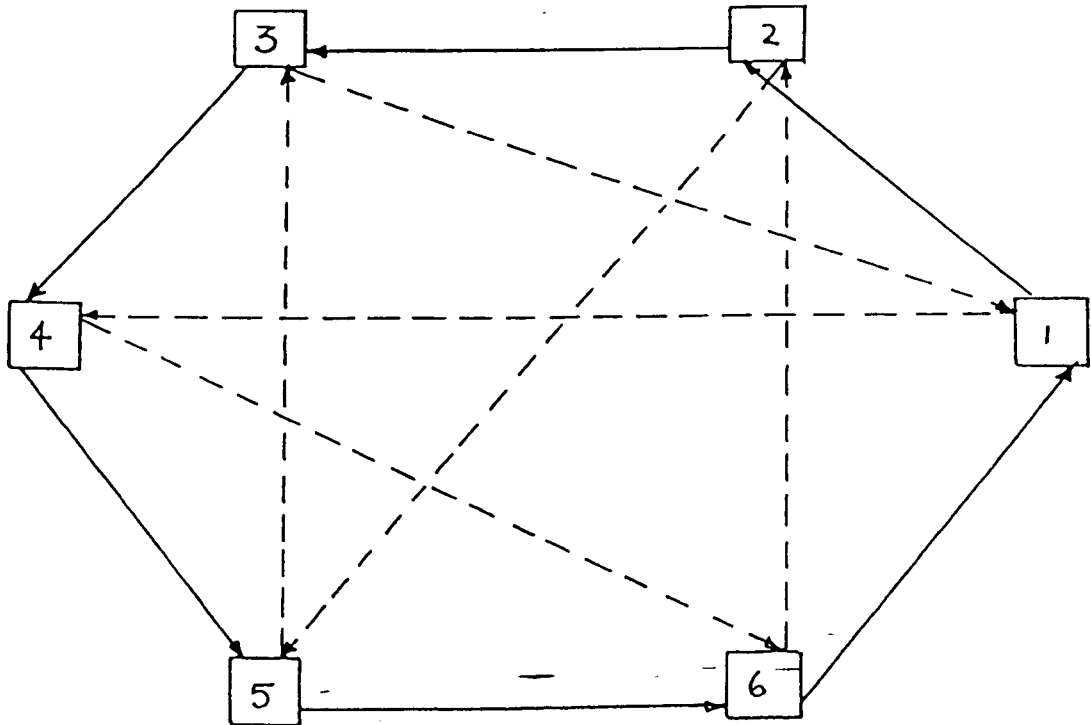– – – – data path

——— Solution ring

11
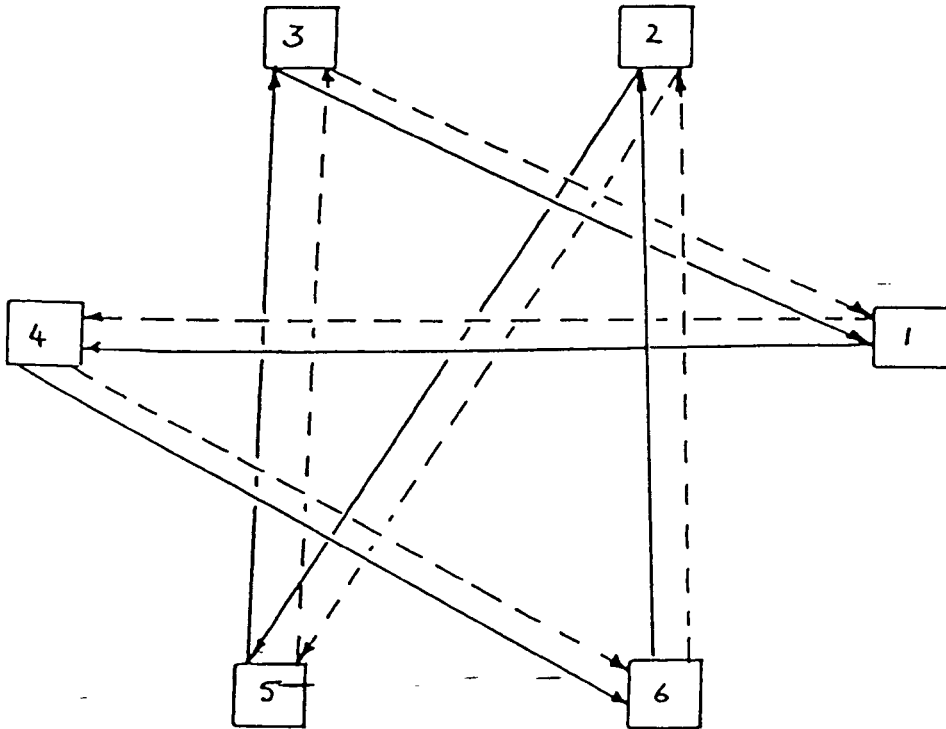
Example 2

For a doubly connected network, with multiplicity m=2, the optimum

configuration is restricted to the following

- - - - - data path

———————— Solution ring

12
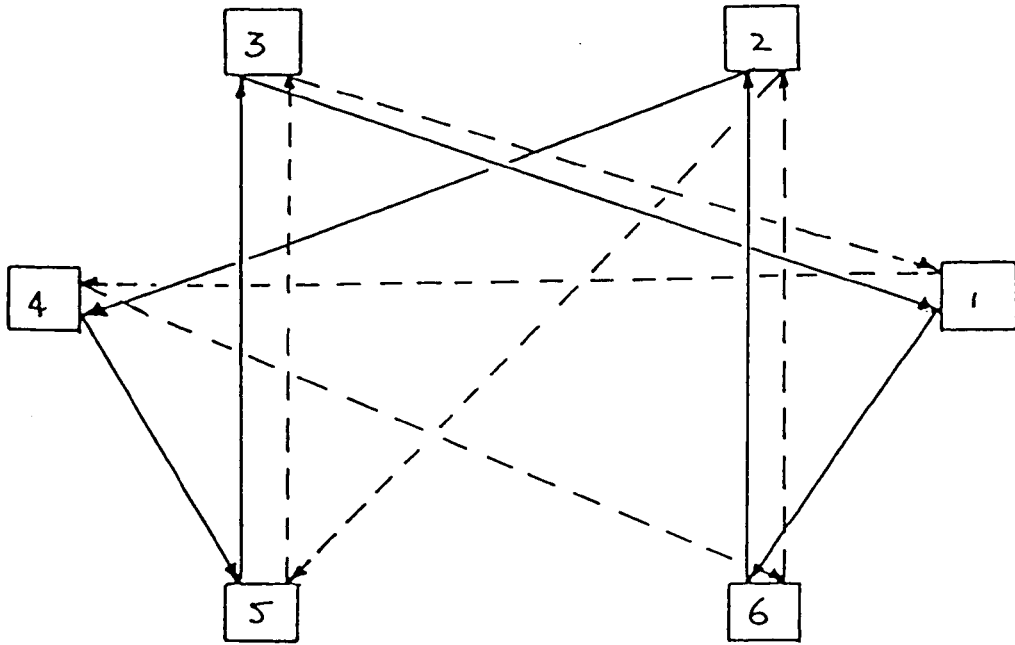
Example 3

For the trivial case of single loop, with multiplicity m=1, the optimum

configuration is the same as the physical configuration

13

6. Discussion

The distributed double loop network architecture [5] and its derivatives [10, 13] provide improved reliability by using two pairs of in- and out-links. In order to improve network throughput, both pairs of in- and out-links at each node are simultaneously active. Nodes run a routing algorithm to choose between the two out-links for transmission of each message. They also have to cope with two simultaneously active input streams from the ring in addition to the input from the attached device. Thus, the nodes, or network interface units, in such networks are quite complex and expensive.

In contrast, in the dynamically reconfigurable network architecture considered in this paper, improvement in reliability is achieved by providing a number of bi-directional links at each node such that the resultant physical topology is a Hamiltonian graph [9]. The number of redundant links at each node does not have to be the same, thus a more critical node can have higher than average number of links. The routing decisions are taken by a central node at the network initialisation on power-up or as a result of link or node failures. The individual nodes require a relatively simple enhancement to include a two-entry routing table, indicating which of the links incident on a node is to be the in-link and which one is to be the out-link. The set of in-link and out-link pairs forms a ring.

Because of the redundant links, more than one such ring formations are possible. In this paper, we have derived the criteria and methodology to

14

choose that ring formation which maximises network throughput for a known data traffic pattern among the nodes on the network. It is also shown that this approach can accomodate any number of link and node failures as long as at least one edge-distinct cycle can be formed from the functioning nodes and links.

For a network with a varying traffic pattern, the control node would update the volume matrix at frequent intervals and perform a reconfiguration whenever there is a significant shift in the traffic pattern.

15

References

1. Pierce, J R

    Network for block switches of data

    Bell Systems Technical Journal, Vol 51, No 6, July/Aug 1972, 1133-45

2. Farmer, W D and Newhall, E E

    An experimental distributed system to handle busy computer traffic

    Proc. ACM Symp. on Problems in the Optimization of Data Comms.,

    October 1969, 1-33.

3. Hafner, E R , Nenadal, Z and Tschanz, M

    A digital loop communications System

    IEEE Trans. on Comms. Vol COM 22, June 1974, 877-881.

4. Lee, R P

    The architecture of a dynamically reconfigurable insertion-ring

    network

    IBM Research Report RJ2485(32434)/3/13/79, 1979.

5. Reames, C C and Liu, M T

    Design and Simulation of the Distributed Loop Computer Network (DLCN)

    Proc. 3rd Ann. Symp. on Computer Appl., June 1976, 124-9

16

6.   Grant, J S

      Multilink - An open network for personal computers

      Computer Comms, Vol 8, No 1, 27-34.

7.   Thomas, P G and Yasin, M M

      Enhanced management domain for low-cost local area networks

      University Computing, Vol 8, No 1, 1986, 46-49.

8.   Thomas, P G and Yasin, M M

      Optimum node location on a ring LAN

      Internal Report 86/12 (To be published)

9.   Wilson, R J

      Introduction to Graph Theory

      3rd Ed., Longman, 1985, 35-38

10.   Ragavendra, C S and Silvester, J A

      Double loop network architecture - A performance study

      IEEE Trans. on Comms., Vol COM 33, No 2, Feb. 1985, 185-7

11.   Liu, M T , Tsay, D-P , Chou, C-P and Li, C-M

      Design of the Distributed Double Loop Computer Network (DDLCN)

      Journal of Digital Systems, Vol V, No 1/2, 1981, 3-37.

12.   M351, Numerical Computation

      The Open University Press, 1985.

13.   Raghavendra, C S and Gerla, M

       Optimal loop topologies for distributed systems

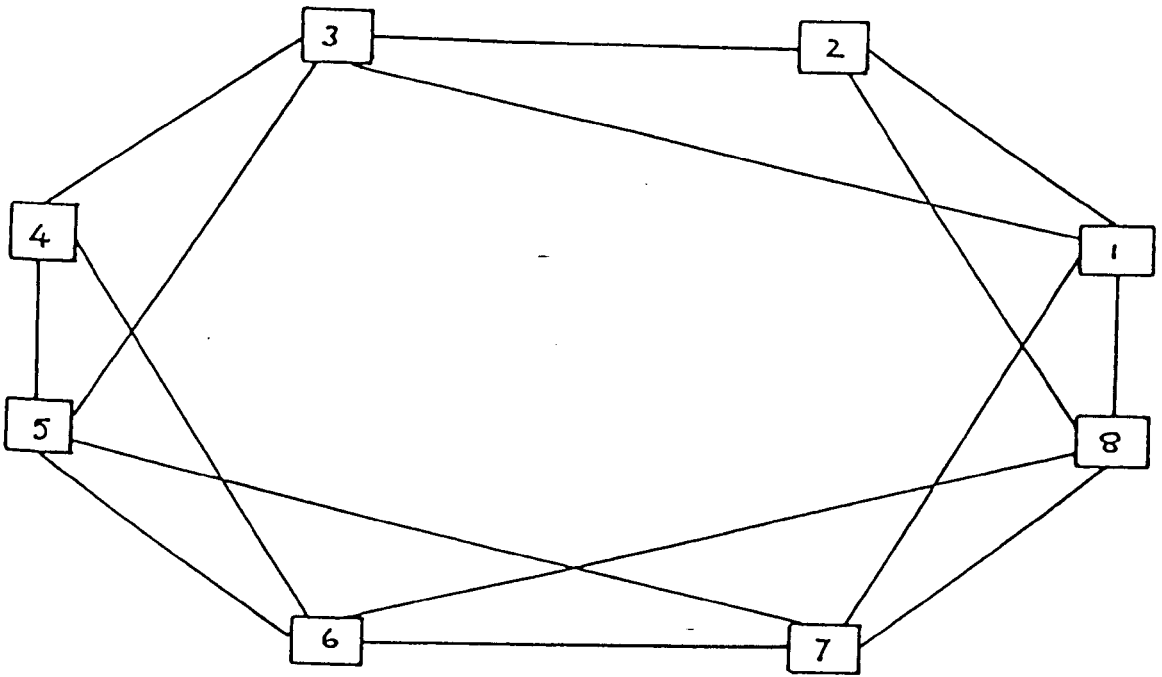   Proc. of Data Communications Symposium, October 1981, 218-223.
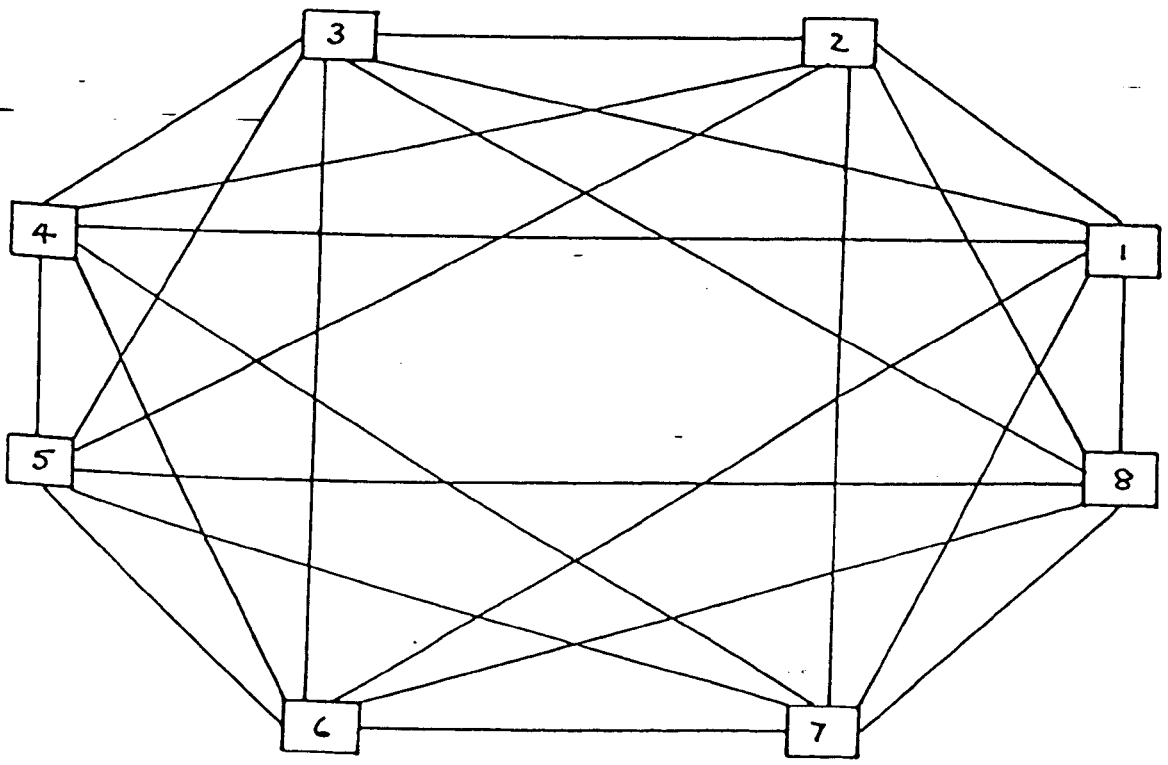
18

Figure 1(a)   n = 8, m = 2

Figure 1(b)   n = 8, m = 3