

INFORMATION TO USERS

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

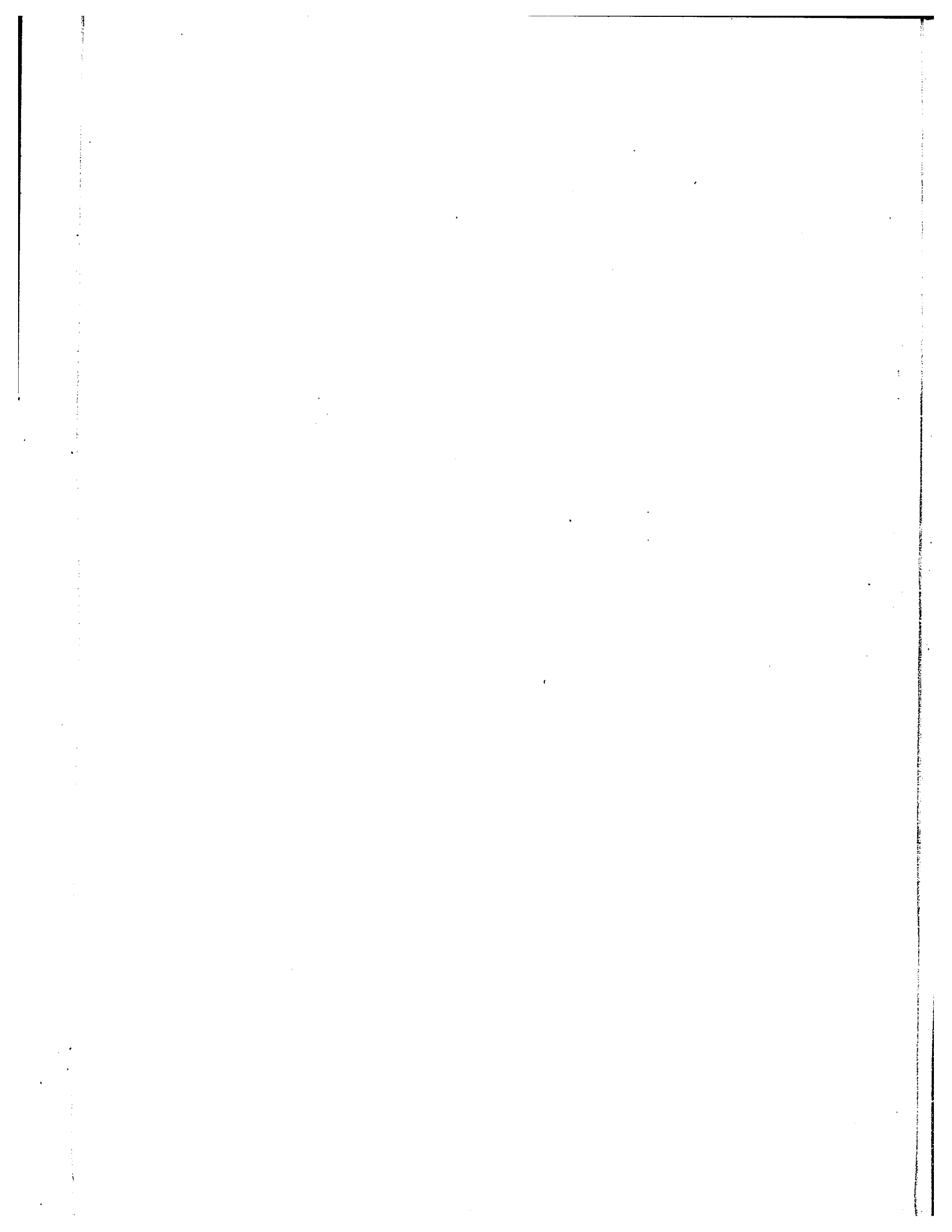
The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

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

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

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI[®]



SC

ANALYSIS OF PACKET-SWITCHED
COMPUTER-COMMUNICATION NETWORKS WITH
CONGESTION CONTROL

by

Amalendu Chatterjee, B. Sc. Engg. (Dac), M.A. Sc. (Ott.)

A Thesis

Submitted to the School of Graduate Studies

Of the University of Ottawa, in Partial

Fulfillment of the Requirements for

The degree of

Doctor of Philosophy

in

Electrical Engineering



Ottawa, Ontario, Canada
April, 1977

UMI Number: DC52418

INFORMATION TO USERS

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

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

UMI[®]

UMI Microform DC52418
Copyright 2007 by ProQuest LLC
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

ABSTRACT

This thesis addresses the problem of congestion control of packet-switched (or message-switched) computer-communication networks. Methods of analysis are given for the following cases: end-to-end congestion control with random routing and the local congestion control with random routing. Expressions have been derived for certain standard performance measures. Analysis is applied to certain examples.

DEDICATION

To my parents who, in their old age, could quietly bear
the separation from their beloved and affectionate son.

ACKNOWLEDGEMENT

I wish to express my sincere appreciation to my supervisor Dr. N.D. Georganas, for the time, guidance and assistance given to me in completing this project. In the process of writing, Dr. Georganas spent many an evening with me. I am grateful to Mrs. Jaycinth Georganas for her patience in this connection.

I wish to express my thanks to Dr. P.K. Verma of Bell Canada for his continued support and constructive criticism of my work throughout my graduate studies.

I also thank Dr. N.U. Ahmed and Dr. K.F. Schenk, Dept. of Electrical Engineering, the University of Ottawa, other members of my Advisory Committee, for valuable discussions.

Finally, a NRC Postgraduate scholarship and the typing assistance from Bell Canada are thankfully acknowledged.

TABLE OF CONTENTS

	<u>PAGE</u>
CHAPTER 1. INTRODUCTION	1
1.1 Overview and Problems	3
1.2 The Approach	8
1.3 Organization of the Thesis	9
CHAPTER 2. FLOW AND CONGESTION CONTROL	11
2.1 Flow Control	11
2.2 Congestion Control	13
2.3 Routing	15
2.4 Flow and Congestion Control in the Telephone Network	16
2.5 The Influence of Routing on Throughput	20
2.6 Flow and Congestion Control in a Packet- Switched Network	24
CHAPTER 3. REVIEW OF RELATED WORKS	27
3.1 End-To-End Control as Used in ARPA net	27
3.2 Local Control as Used in ARPA net	35
3.3 An Analytic Approach To End-To-End Congestion Control	40
3.4 Analysis of Local Congestion Control	52
3.5 Random Routing	57
CHAPTER 4. NETWORKS WITH END-TO-END CONGESTION CONTROL AND RANDOM ROUTING	62
4.1 Introduction	62
4.2 A Queueing Model For Congestion Control	66

	<u>PAGE</u>
4.3 Performance Measures	58
4.4 Equivalent Closed Queueing System And Its Analysis	71
4.5 Numerical Example	75
CHAPTER 5. NETWORKS WITH LOCAL CONGESTION CONTROL AND RANDOM ROUTING	84
5.1 Introduction	84
5.2 Requirement Of Local Congestion Control	85
5.3 The Queueing Model	92
5.4 Analysis Of The Controlled Link	95
5.5 Numerical Example	100
CHAPTER 6. SUMMARY AND CONCLUSIONS	104
REFERENCES	107
APPENDICES	

CHAPTER - 1

INTRODUCTION

Computers were first invented as problem solving devices. With the need to extend the problem-solving power to remote users, computer networks were developed and their growth increased tremendously in the last decade. Recently, because of the attempts to share computer resources, such as specialized hardware and software packages and data banks, it has been shown that existing computer systems could be better utilized when interconnected by a communication network. In this network, computer resources are distributed among the nodes (computers) and we term this configuration as "Distributed Computer-Communication Network". The major function of this type of computer-communication networks is to handle data communication. There are broad ranges of issues, such as network design, multiplexing and switching arrangements, effective buffer utilization, routing, deadlock prevention, flow and congestion control etc., to be resolved, in order to have effective data communication with high efficiency. Figure 1.1 shows the relative growth of computer installations compared to the utilization of off-line computer power [1].

In this thesis, we will consider the issue of congestion control, in conjunction with certain routing techniques in a packet-switching or message-switching mode.

The biggest experimental project of such distributed net with a packet-switched mode is the ARPA Network [2], [3]. Commercial networks of this type are coming now into operation [14], [21]. Figure 1.2 shows a general configuration of a distributed type of data communication networks. Basically, it consists of 1) a collection of HOSTS (main computers) providing service to users (terminals or other HOST computers) and 2) a subnetwork providing communication among computers or terminals (users) or both, through an interface consisting of switching nodes and transmission links.

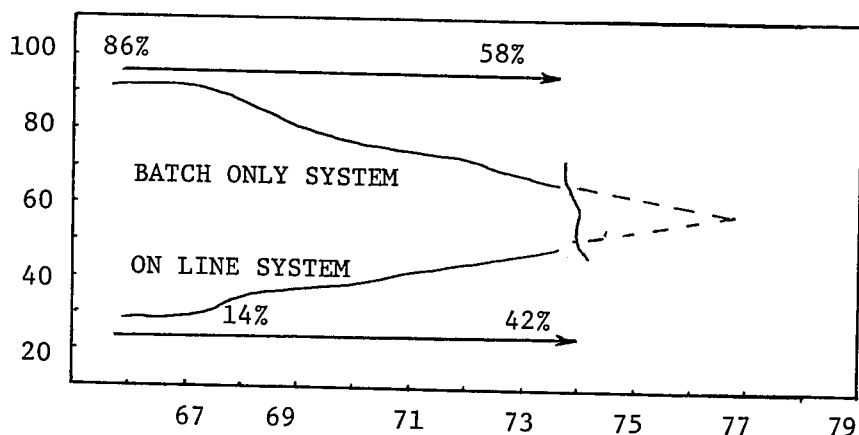


FIGURE 1.1: GROWTH OF DATA MARKET

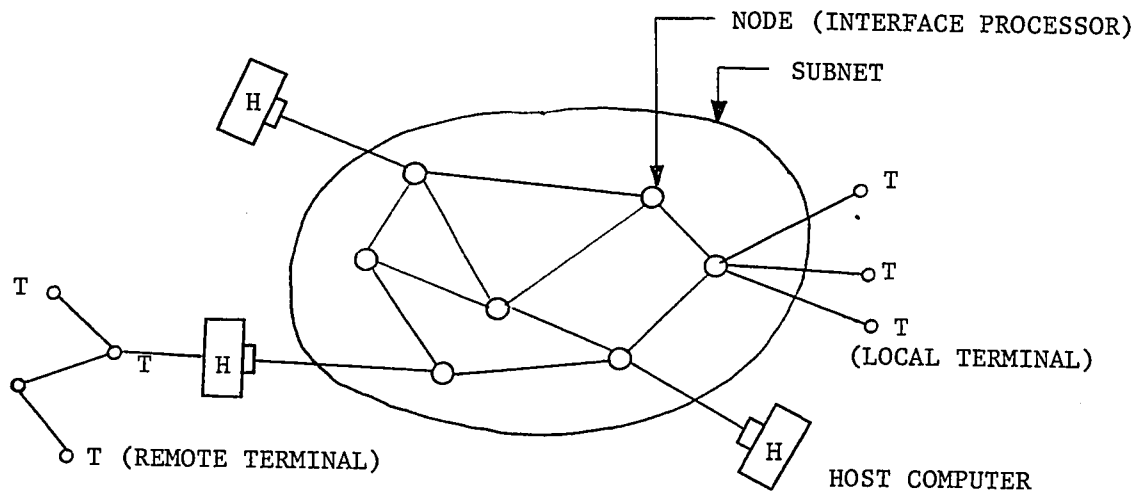


FIGURE 1.2: DISTRIBUTED COMPUTER-COMMUNICATION NETWORK

We will refer to the knowledge and the research experience gained through the ARPA technology, as we proceed with our problem of congestion control in later sections.

1.1 Overview and Problems

It is not very difficult to connect two computers and hence two users (or terminals) through a communication net for data transfer. However, when we think of the problem of effective data communication in a real time amongst a multitude of users with diverse requirements, while retaining the network resource utilization at an optimal level, we are faced with a very significant problem. Customers are always concerned with delay, throughput and cost, for their specific application while application programmers, computer centre managers or information system design groups are faced with the problem of achieving an optimal assembly and an efficient utilization of a collection of readily available hardware and software components to provide an effective spectrum of capabilities.

From the customer's point of view the traffic can be classified as follows [4], [5]:

- 1) High - Throughput Traffic (HT Traffic)
- 2) Low - Delay Traffic (LD Traffic)
- 3) Real - Time Traffic (RT Traffic)

These types of traffic are specified by the delay (elapsed time from initiation to completion of transmission), effective

throughput (transmission rate) and accuracy requirements. The conditions imposed on the communication subnet design by each of these types of traffic vary. For our purpose, we will assume that all types of traffic are present in the system. Subnetworks, as mentioned earlier, can be distinguished by the communications medium employed and the manner in which the medium is configured for communications, e.g., switching [22]. There are three switching technologies that can be used for point to point data transfer [6], [7], [8], [9], [10]:

- 1) Circuit Switching
- 2) Message Switching
- 3) Packet Switching

Circuit Switching implies the establishment of dedicated channels, or circuits, in support of communication between user and a HOST, or between HOST and HOST. A circuit is the basic physical path over which information travels. A channel is the logical (Simplex, half duplex, or duplex) path connecting user to user, user to HOST or HOST-to-HOST. Circuits may be multiplexed to support several channels. In a circuit switching network, the time required to establish a point-to-point connection is large compared to the actual time required for information transfer, especially for LD and RT traffic. Effective utilization of circuit switching requires careful matching of circuit bandwidth (which defines the rate at which information can be transmitted) and is usually measured in bits-per-second against the (usually time-varying) transmission

requirements. For this reason, circuit switching seems ill-suited to bursty traffic, in which throughput requirements vary and can result in periods of line under-utilization.

A message is defined to be a logical unit of information from the viewpoint of the user. Thus, telegrams, programs and data files are examples of messages. As a natural alternative to circuit switching, one may provide a communications utility that functions in a manner analogous to the role of the Post Office. Thus, an individual message is "dropped into" the subnetwork, which then assumes responsibility for its delivery to the destination. The use of this type of a communications utility permits increased circuit utilization. A message switching subnet may be regarded as a collection of physical circuits interconnected by switches through which individual messages are stored and forwarded, as they are routed from the source to the destination. In message switching, an individual message may pass through several circuits in travelling from the source to the destination. As a result, the opportunity exists for utilizing one circuit to carry traffic between several source destination pairs. This, in turn, permits increased circuit utilization. In general, derivation of utilization, delay times, busy intervals and idle periods of message switched systems is very involved [10]. One disadvantage of message switching, is that short messages, in the presence of longer messages, suffer more delay.

One method of reducing the delay encountered by short messages (in message-switched networks), while waiting for the transmission of larger messages, involves providing a collection of static priorities assigned on the basis of message length. This has been proven not to be so satisfactory for all traffic types [10]. Similar objectives can be achieved by subdividing messages into packets with a pre-established maximum size prefaced with suitable address information. Single packet messages can then be transmitted with a minimum of delay, while high throughput can be achieved for multipacket messages, through the simultaneous transmission of several packets. The APRANET, the TRANSPAC and the DATAPAC are examples of packet-switched networks [2], [11], [14], [21]. The analysis of trade-offs, which must be considered in evaluating circuit, message and packet-switched subnet technologies is rather involved [13], [14]. The packet-switching approach does seem to possess significant advantages since it permits simultaneous support of multimodal traffic (with reduced response times) [10]. We will address ourselves to the packet-switching technology and especially the congestion problems of the packet-switched networks assuming packets having a predetermined length, obey similar properties as messages. Flow and congestion problems in circuit-switching are not as critical as in message-switching or packet-switching networks. In circuit-switching, flow and congestion control are faced only during the call establishment phase and so data are not affected, if a call has to be abandoned during this phase because of flow and congestion problems of the network. But in the message or packet-switching case, once the packets or messages

are in the network, they cannot be abandoned during the congestion period of the network, as purging of packets will mean transmission failures and it will create inefficient utilization of the network resources. With proper flow and congestion control schemes in the packet-switched environment, the situation can considerably be improved. Routing in conjunction with congestion control determines throughput. When the network is operating at a traffic level much lower than the design limit, congestion control is not brought into effect and the throughput is determined by other system-parameters. Routing strategy is one of them. At high load, congestion does pose a serious problem and the situation may further be aggravated by some node and link failures. This is because any node or link failure will cause the network resources to be taxed beyond its design limits and this will create a pathological situation unless a proper congestion control scheme is there to take care of it.

The purpose of this undertaking is to propose an analytic model for congestion control in a computer communications network. Basically there are two ways to approach this problem. They are: 1) Simulation
and 2) Mathematical Analysis.

This thesis presents an analytic solution to the congestion problem under random routing. We have adopted an analytic approach in preference to using a simulation technique since analytical techniques present a generalized view and a more

perceptible relationship between the parameters involved.

Using the well known "independence assumption", Kleinrock, analyzed many aspects of message-switching networks [6], [10]. We will follow a similar approach.

1.2 The Approach

Routing strategy is an integral part of any packet-switching network design. A routing technique may be defined as a procedure adopted to choose the actual path for a packet or message from amongst a set of available alternatives. Even if the communication net has traffic within its handling limit, because of bad management of the available routes, the traffic at some part of the net may be "congested", thus causing more delay and less throughput, whereas other parts may be less utilized. This becomes critical when the network is subjected to unbalanced traffic.

Two parameters R and W , which are functions of both time and space, govern the routing procedure to be adopted to satisfy certain performance criteria. R is a matrix whose elements r_{ij} denote the paths available for going from node i to node j , which may change because of congestion and branch or node failures. W is another matrix with elements w_{ij} denoting arrival rates, whose value is different at different nodes at different times. The function of the routing techniques is to best allocate resources, R to accomplish the work, W . By "Best" we mean the minimization of average packet transmission time and hence maximization of throughput through the network.

If this cannot be accomplished because of node or link failures and erratic demands on resources, the average time for a packet to reach destination increases lowering the throughput. This state of the network condition in the presence of link and external customers, explained later, will be defined as congestion. There are different classifications of the routing techniques [15], [16], [17], [18]. In our analysis we will be mostly talking of fixed, random, alternate and dynamic routing.

A mathematical analysis of flow and congestion control in a packet-switching mode with fixed routing has been done in [19], [20]. In this thesis, we will remove the limitation of fixed routing in the model of end-to-end and local congestion control schemes, to show the usefulness and the effectiveness of random routing compared to fixed routing in terms of network congestion. We believe that these results can be used to provide an insight into the case of alternate or dynamic routing for network design, which are not easily amenable to mathematical analysis.

1.3 Organization of the Thesis

This thesis consists of six chapters. In the second chapter we gain experience from the telephone network to handle flow and congestion control in the data communication network. The main purpose of this chapter is to look for a unified definition of flow and congestion for both telephone and data networks. This unified definition will form the basis for our further investigation of the problem.

In chapter 3 we review all the work in this area and find that because of the lack of proper mathematical tools, very few work has been done. We identify these problems and find some way to proceed in chapters 4 and 5.

In chapter 4, computer-communication networks with end-to-end congestion control and random routing are modelled. From this model random routing is compared with fixed routing in terms of defined performance measures.

In chapter 5, it is first observed that end-to-end congestion control alone, where there is common buffer usage, may not provide complete solution to the congestion problem. Here we model the network with local congestion control and random routing and try to establish a criterion for how end-to-end control in conjunction with local control can potentially solve the congestion problem in the network.

In chapter 6, we summarize our results and identify future research problems for a complete solution of the flow and congestion problem.

CHAPTER - 2

FLOW AND CONGESTION CONTROL

Any system where a given amount of resources is going to be shared by a number of contending independent and unrelated users, requires the adoption of a control scheme for its proper functioning. Controls may also be required to optimize performance under varying load conditions [22], [23], [24].

This Chapter will specifically concentrate on Flow and Congestion Control explaining their respective functions and giving means of realization.

We will find it convenient to regulate the traffic of either a circuit-switched or a packet-switched network by two control techniques, without excessive loss of throughput and excessive delay. These are: 1) Flow Control and 2) Congestion Control. There might be different ways and means of implementing these techniques, depending upon the traffic pattern and the design requirements.

2.1 Flow Control

In general terms, flow can be defined as a physical displacement of goods, commodities or information of significance or value. A flow network is composed of nodes, which act as sources, sinks or tandem points and transmission facilities.

It is generally a requirement that a flow network provide an orderly flow, free from loss or mutilation (or duplication in case of information transmission) to an acceptable degree and that the division of resources be equitable in case of congestion.

In any flow network, a scheme to allocate resources amongst users is required. Further, if there is a finite possibility of the total instantaneous demands placed on the resources exceeding the available resources, a scheme to resolve contention must also be adopted. A flow control scheme is thus an algorithm adopted by the network to determine how resources will be allocated between (independent and unrelated) users and to how contention will be resolved between the bidders, when resources are exceeded.

Thus by the definition of Flow Control, it is apparent that its provision is essential to every multi-user network. Some samples of the alternative ways in which the basic two functions of Flow Control could be realized are furnished below:

Allocation of Resources:

- (i) First Come First Served
- (ii) Last Come First Served
- (iii) Establishment of Priority Classes

Resolution of Contention:

- (i) Blocked and Lost
- (ii) Blocked and Delayed

2.2 Congestion Control

Congestion is a well experienced, if imprecisely defined phenomenon. For the purpose of a communication network, it is convenient to define congestion as a situation where the network throughput starts reducing in the face of increasing traffic. The purpose of adopting a congestion control is to prevent or minimize loss of throughput, as the load on the network increases.

There are two basic procedures involved in any congestion control:

- a) Detection of Congestion and
- b) Institution of a suitable Control procedure.

A simple case illustrating the occurrence of Congestion is shown in Fig. 2.2.1. Identification of performance corresponding to Point A on the curve would coincide with the occurrence of congestion. In actual practice, measurement of congestion in a real network may not be an easy task. Measurement of round trip delays on 'pilot' packets, or the measurement of buffer fills at a node, may provide identification of congestion to an acceptable degree of accuracy.

The effectiveness of a particular congestion control mechanism could best be measured by a comparison of throughput obtained without any control (curve a) and one obtained when control is being exercised (curve b). A single point measure may

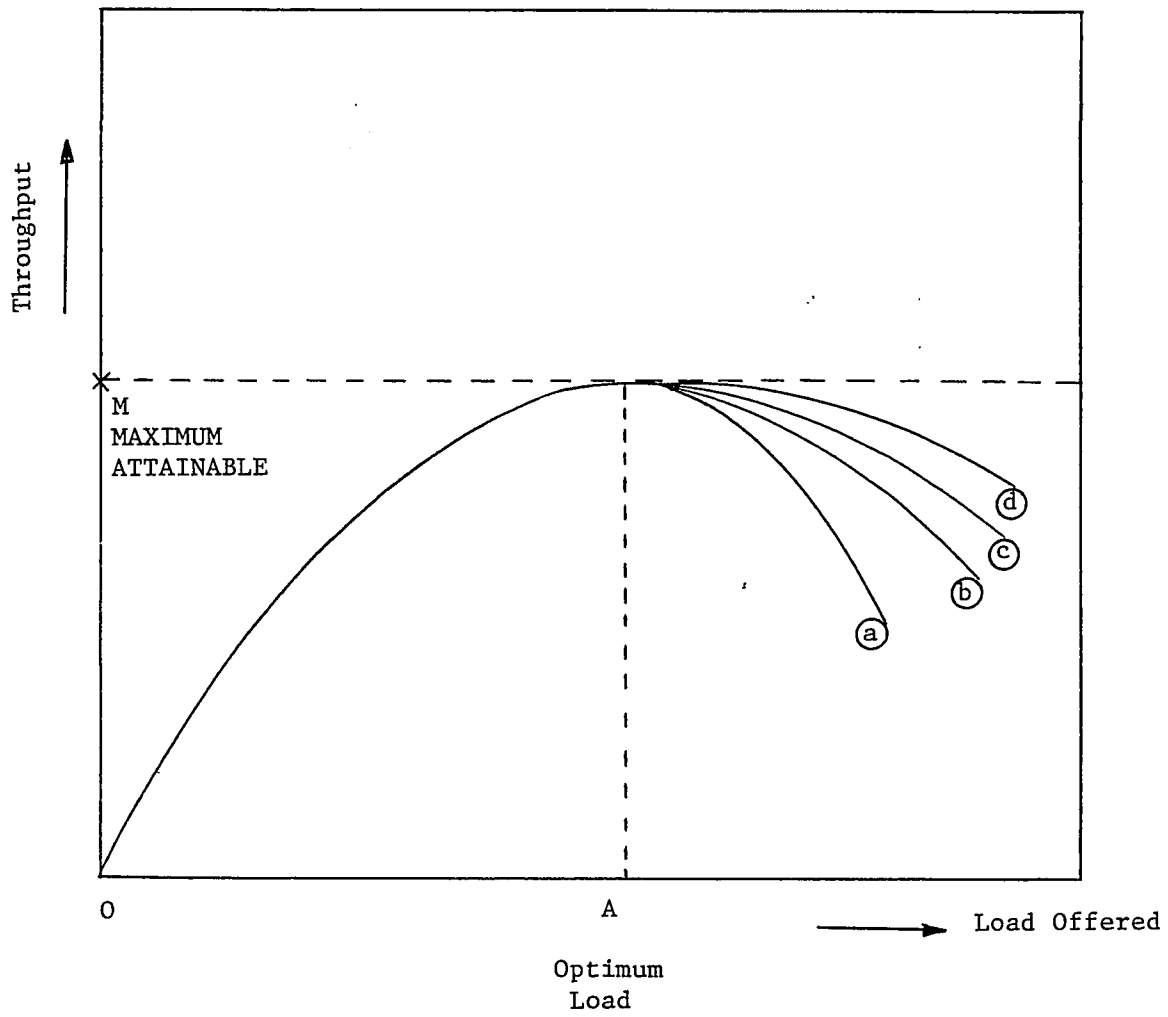


FIGURE 2.2.1: THROUGHPUT AND OFFERED LOAD RELATION IN A TELECOMMUNICATION NETWORK

generally be sufficient. Such a measure could be defined as the percentage reduction in throughput, with a certain percentage of traffic enhancement over and above the traffic corresponding to the maximum throughput, when no congestion control was instituted. Better and more efficient congestion control techniques may improve the throughput further up, as shown in curves c and d of Figure 2.2.1.

2.3 Routing

The word routing is synonymous with providing a path or paths, consisting of communication links, between the source-destination pairs in a network. Routing can be either fixed or multiple.

Fixed routing implies the availability of one and only one possible path of communication links between one source destination pair. Failure or unavailability of any link or node in this path, will automatically result in traffic between this source-destination pair being denied access to the network, or rejected, while in progress to the destination.

Multiple routing implies a scheme, where more than one path is available between one pair of source and destination nodes. Multiple routing can be achieved in several ways, such as a simple hierarchy of alternate routing based on a time out criterion, a hierarchy or routing based on origin and direction of traffic or a fully adaptive routing providing an optimum path for every new packet depending upon the net-

work condition at that time. The reason for providing multiple routing in a network is to increase its throughput, by increasing the call completion probability under certain load conditions. Other examples of multiple routing are random and dynamic routing. We will deal with random routing in greater details in later chapters.

Use of a single congestion control scheme, with varying routing strategies, will also result in the realization of different throughput vs. incident load curves. Optimum routing is a function of the incident load. In the following section, an example is presented illustrating the flow and congestion control techniques adopted for the telephone network.

2.4 Flow and Congestion Control in the Telephone Network

A telephone network is the prime example of a network, where considerable attention has been paid to provide suitable flow and congestion controls for operation during normal and overload conditions. Mechanisms to allocate switching resources at the central office and the various signalling information guiding the calling subscriber regarding the progress of the call establishment (advising him to abandon it if necessary), are examples of a flow control scheme providing for both allocation of resources and resolution of contention.

The congestion control schemes adopted in the telephone network are based on the properties of the hierarchial switching system used for call routing, and have been formulated on the basis of network simulation and actual experience. The hierarchy of call routing in North America is well known. With the hierarchy adopted, calls between one end office and another are completed through as low a hierarchial chain as possible.

The following three methods of congestion control are adopted in the telephone network [25], [26].

1. Reduction of Alternate Routing
2. Directionalization of Traffic Flow
3. Prevention of Feedback Amplification of sender delays and reduction of intra-office switching delays.

The cancellation or reduction of alternate routing may appear at first sight to mitigate against the call carrying capacity of the network. This is indeed true under normal conditions, but under overload conditions, as the offered load increases, the proportion of messages that are alternately routed increases also. This results in a higher trunk usage per message, so that the number of messages that can be carried is less than the number of messages carried by an equivalent network without alternate routing.

Directionalization of traffic flow is a means adopted to give preference to traffic moving from an office of higher rank to one of lower rank at the expense of traffic moving from lower rank to higher rank. The rationale for doing this becomes apparent, when one considers the fact that the preferred traffic is more likely near its destination than the non-preferred traffic. In other words, use of this scheme results in a nearly completed call being given further preference, to enhance the probability of its completion, resulting in a higher overall completion rate. The net result, with regard to the whole network, is to maximize the completion rate under varying offered load, and make the network behave like a single server (which never congests but only degrades in performance).

Senders are devices which receive, store and forward the address digits of an attempted call. Sender congestion tends to propagate from one office to another under overload conditions and amplify in the process. A reduction of 'time-out' interval during a period of sender congestion results in reduction in congestion feed-back and improves the general performance.

Several conclusions of a general nature could be drawn from the above. The first and the foremost of these is that several practices, which tend to increase the call carrying capacity of a network during normal operating conditions, tend to further congest the network during overload conditions.

Chief factors amongst these are the procedures, which tend to 'delay' the progress of a call with a view to enhancing its completion probability. While perfectly valid under normal conditions, such delays actually lower the total number of completed calls during an overload condition, thus degrading the network performance. Thus, while the emphasis during normal conditions is to introduce delay, if necessary, by providing longer timeouts etc., the emphasis during overload conditions is to reduce delays as much as possible in call completion, thereby giving a preferential treatment to all calls which can be completed fast.

The fundamental mechanisms, which cause congestion are similar in both the telephone network and the packet-switched computer network. There are two important differences however. The telephone network is a circuit-switched network, engineered by and large on a blocking and loss basis, whereas a packet-switched computer network is traffic engineered on a blocking and delay basis. Further, in a circuit-switched network, all the effects of delay or congestion are experienced only during call set-up. On the other hand, a computer network, which is generally a store and forward network, will provide a degradation in terms of reduced throughput or higher delay, throughout the duration of a call, consisting of several blocks or packets. Because of the above two reasons, congestion in a packet-switched computer network can be more serious as it will enhance

delays and deteriorate throughput of calls already in progress and a greater design effort is required to mitigate its effects under overload conditions. Besides, unlike the situation in a circuit switched network, where a call in progress (during the establishment phase) can be abandoned at any stage of its progress, messages in progress cannot be purged out in a computer network under overload conditions. The only place where a message could be throttled in such a case would be right at source. It also follows, for the same reason, that the initiation of congestion control should take place somewhat ahead of its actual anticipated occurrence in a packet-switched network.

2.5 The Influence of Routing on Throughput

What influence does routing have on the network throughput? The simplest kind of routing as discussed in the previous section is fixed routing, in which there is only one path that a particular call may take. In the analysis here we are postulating that reliability of operation is not the criterion for having a routing scheme other than fixed. For, even in the case of fixed routing, a higher reliability can be attained by simple facility restoration. The fundamental reason for adopting a routing scheme is, therefore, to enhance throughput or reduce delay. It is in this light that the influence of routing will be considered in this section.

The influence of routing on throughput will be considered qualitatively under three loading conditions:

- a) Light
- b) Moderate
- c) Heavy

A light loading situation can be defined as the situation under which an insignificant amount of calls will be rejected under the route-busy condition. It also follows that irrespective of the adoption of the hierarchical or the adaptive routing, the shortest path will be adopted. Under light loading situation both the fixed and the multiple routing schemes should therefore produce the same throughput. (This is with the assumption that little or no overhead is required for administering a multiple routing scheme - such as the overhead bits required for managing an adaptive routing scheme).

Under moderate loading conditions, a larger percentage of calls, that would have been rejected due to the only available path being busy in a fixed routing scheme, would be completed through an alternate path in a multiple routing scheme and the provisioning of alternate or hierarchical routing will therefore increase the system throughput. The point to note here is that all calls, that are completed through other than the shortest path, generally use a longer number of links per completed call.

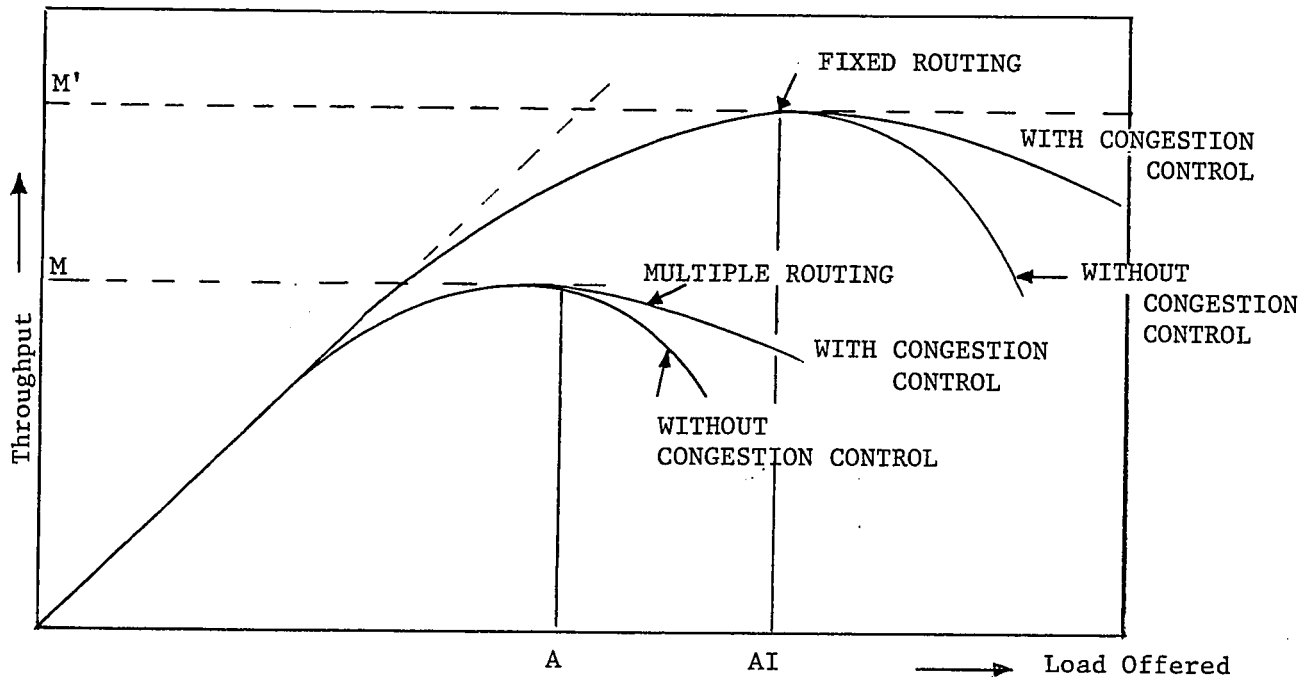


FIGURE 2.5.1: EFFECT OF ROUTING ON THROUGHPUT

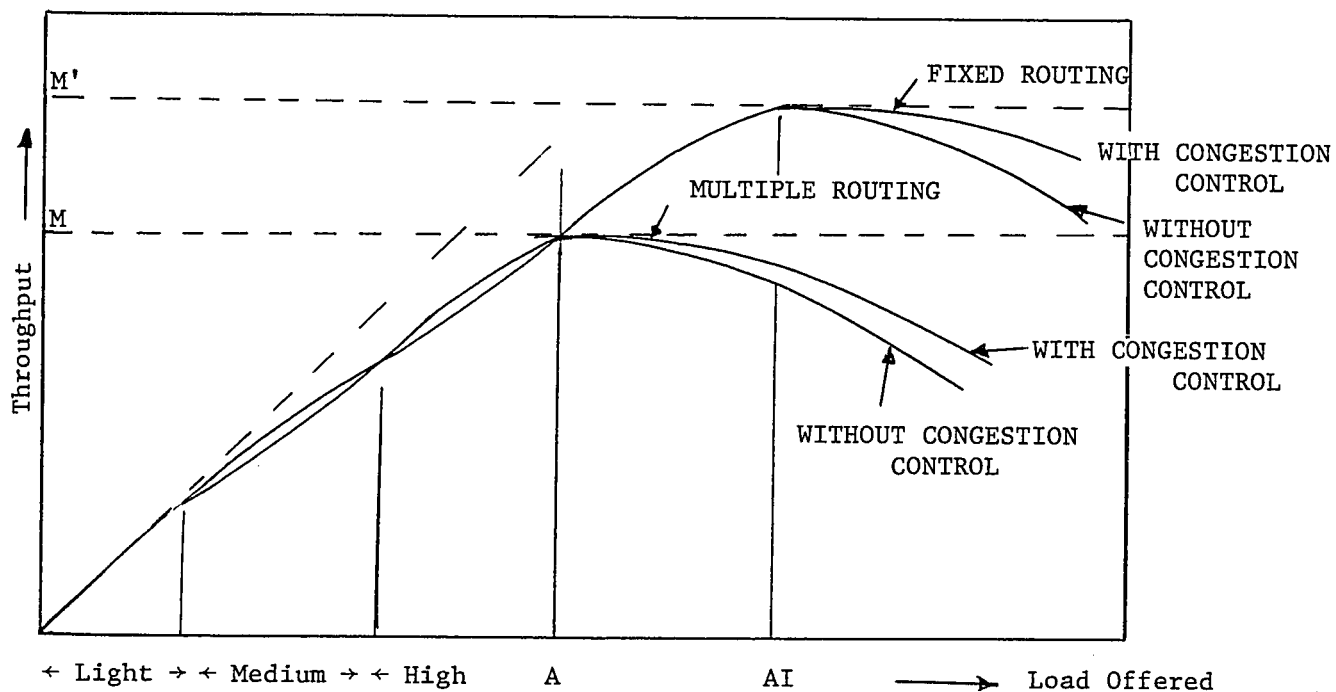


FIGURE 2.5.2: EFFECT OF ROUTING ON THROUGHPUT AT DIFFERENT LOAD CONDITION

Under a heavy load situation the above factor actually results in a lower number of calls completed in a situation with multiple routes compared to the situation when a fixed route only was available. As discussed before, under heavy loading conditions, the throughput of all real networks will fall, as traffic is increased. It also follows that, because of the reason given above, i.e., deployment of a larger number of links per completed call, the throughput of a system with a larger number of routing options falls more rapidly, under heavy loading conditions. The strategy for optimal routing should, therefore, be alternate routes during light and moderate loads and fixed routing under a heavy load. Since the ultimate capacity of a multiple routing scheme will be less than under a fixed routing scheme, the multiple routing throughput curve will peak earlier than the fixed routing throughput curve. Further, this peak will be smaller than the throughput attainable under the fixed routine scheme at the same load.

A typical relationship between throughput and load offered is shown in Figures 2.5.1 and 2.5.2. The differences in the shapes between the two figures, at high and medium loads might be due to different factors, such as topology, link length and types of routing schemes employed.

2.6 Flow and Congestion Control in a Packet-Switched Network

The purpose of the previous sections was to examine flow and congestion control from a general and qualitative point of view. In particular, the influence of congestion control techniques and routing on throughput was examined and it was shown that similar relationships exist between throughput, delay and incident load for both the telephone network and a packet-switched computer communication network. In the following, flow and congestion control techniques, applicable largely to a packet-switched network, are examined.

A general definition of flow control has previously been given. There is no fundamental difference in implementing a flow control technique, either in a telephone network or in a computer-communication network. As allocation of resources and resolution of contention are the same in both cases, we will not discuss more about this here.

There is one fundamental distinction between the congestion control mechanisms exercised in a circuit-switched network and that in a packet-switched network. In a circuit-switched environment, congestion does not degrade the performance of an established connection, rather it results inordinate delays in establishing the initial connection or in a lowering of the call completion rate. On the other hand, a packet-switched network can create degraded performance during the course of a call, because the successive packets

constituting the call can be separately delayed by varying amounts, depending upon the dynamic state of the network. For this reason, initiation of congestion control should take place somewhat ahead of its actual anticipated occurrence in a packet-switched network. Further, in the exercising of the control mechanism itself, a circuit-switched network can apply a congestion control rather abruptly, compared to a slow control essential for a packet-switched network. To substantiate our statement and to show the effectiveness of different techniques, we will resort to mathematical analysis in later sections.

Three ways of controlling congestion in a packet-switched network have been reported in the literature. These are:

- 1) End-to-End Control,
- 2) Local Control,
- and 3) Isarithmic Control.

The definitions and mode of operation of these techniques can be found in [3], [27], [28], [29], [47]. We will differ from the usual definitions in the sense that we will call these operations as congestion control instead of flow control or flow and congestion control. None of these techniques can stand alone and prevent congestion completely.

End-to-end congestion Control is exercised by placing a limit on the maximum number of packets that can be present at any one time in a "logical link" of a source-destination pair.

Local Congestion Control is a technique, in which the arrivals of packets in a particular node are not permitted, when that node is occupied beyond a certain level in the sense of buffer fill.

Isarithmic congestion control, as simulated by Davies [27] is a technique, in which the total number of packets, to be present at one time in the network, is kept constant. Analysis of the Isarithmic technique is based almost entirely on simulations.

In this thesis we will concentrate our attention to the end-to-end and local congestion control [20], and will show later how the combination of the two may potentially solve the congestion in the network.

CHAPTER - 3

REVIEW OF RELATED WORK

We have presented the definitions of flow and congestion control in the previous chapter. In the following chapters, we will address the problems of congestion control exercised on an end-to-end or local basis. A general network with random routing, will be used in both cases. In this chapter, we will review the work done so far in the areas of flow and congestion control as defined above. Before we do that, we will discuss some of the concepts used in the early ARPA net for end-to-end control and local control for effective data transfer. It may be mentioned here that the concept of congestion control, as defined by us is not quite the same as the end-to-end control and local control, used in the ARPA terminology.

3.1 End-to-End Control As Used In ARPA Net

In ARPA terminology this is also called source/destination or HOST/HOST flow control. The early form of this control procedure was to use the concepts of "links" and 'Request for Next Message (RFNM)' [3]. A link is nothing but a unidirectional logical connection between a pair of HOSTS. The HOSTS may make further use of links as an addressing mechanism to specify particular process-to-process, subscriber-to-subscriber or terminal-to-terminal connections. Each end of the connection is named a 'socket'. A process is in com-

munication with another process, after exchanging a sequence of messages through the links. The two processes agree upon a link, for use in connecting a pair of sockets, in advance, by means of a HOST-to-HOST 'protocol'.

Two links and therefore four sockets are in use for two way communication at the same time. Only one message at a time was allowed to be present in the subnet on a link [3]. An RFNM is sent to the sending 'IMP', immediately after receiving a message, through the same link used for sending the message. The IMP passes the RFNM along to the HOST. Under their control scheme the source HOST cannot send the next message on that particular link, until it receives the RFNM. But the source HOST can create other links, while waiting for one RFNM through a link. In the second phase of ARPA operation for minimum delay and maximum bandwidth, each IMP used to subdivide each message into one or more packets. These packets were independently transmitted, together with their sequence numbers, and reassembled back at the destination IMP before being sent to the HOST. In this way this subdivision was invisible to the HOST. This seemed to work well in the ARPA net under light load. But as the ARPA net grew in size and as there was more competition for resources, i.e., as the load intensity increased, many problems were discovered one by one. Many authors explained these problems and tried to give solutions [19]. One of these problems is the 'reassembly lock-up'. Complete

'deadlock' is possible under this condition. By deadlock we mean formation of a closed loop as explained below within the network for want of network resources and hence complete halt of information transfer. Reassembly lockup occurs due to shortage of reassembly buffers. Reassembly buffers are used to assemble the broken message pieces, i.e. packets. In each node there are two kinds of buffers, i.e., reassembly buffers and 'store-and-forward' buffers. Store-and-forward buffers will be explained in a later section when we will talk of local control.

To explain reassembly lock-up let us take the help of the network of Figure 3.1.1 where IMP A is the multi-packet source and IMP A' is the multipacket destination. If now at IMP A' all the reassembly buffers are either occupied or reserved for expected packets of partially reassembled messages and if neighbors of A' are also filled with store-and-forward packets also headed to A', which IMP A' cannot accept, then this prevents packets at other IMP's from reaching destination A' and completing the partially reassembled messages. This is the reassembly lock-up.

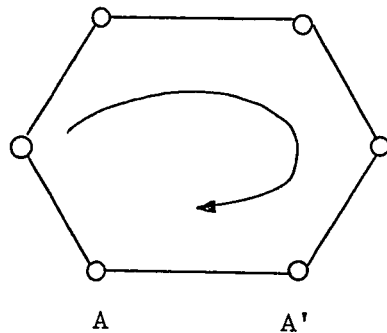


FIGURE 3.1.1: REASSEMBLY LOCK-UP

There are many solutions to this problem. One such solution is to discard the packets of any message that cannot be reassembled for a long time and notify the sending HOST of their disposal. This causes transmission failure and thus it is not desirable. The other solution is to have the source IMP ask the destination IMP for full reassembly space for each message before creating links to send the messages. As in Figure 3.1.1, if IMP A or other IMP's try to send messages to A', they will first ask for reassembly space reservation before sending messages arbitrarily, as was done in the early stage. This introduces occasional delays, because when reassembly space is not available, the space request is queued at the destination IMP until space becomes free. For minimizing delays a single-packet message acts as its own request for space. But a copy is always kept at the source IMP for retransmission, in case the packet is discarded at the destination IMP, because of lack of space. When ultimately space becomes available the source is informed of it and retransmission occurs.

For multi-packet messages, a short space request message is sent first by the source IMP ahead of the regular message. No duplicate of the request message is kept at the source IMP. If space is not available after the first request, a time-out occurs and the source IMP sends another request message. This introduces a set-up delay for the first packet of the multi-packet message, no doubt, but it ensures that multi-packet messages, that could congest the net,

are not permitted to enter. This technique of subnet allows high bandwidth traffic between two HOSTS at the cost of a set-up delay. When the first packet of a multi-packet message encounters a set-up delay, each succeeding multi-packet message to that destination can avoid encountering the delay, if each is sent as explained below upon receipt of the RFNM. For the subnet to achieve this objective, an additional set of reassembly buffers is reserved as soon as possible, after the first packet of a multi-packet message is sent to the destination HOST, after which the RFNM is returned to the source IMP. The next multi-packet message from the source IMP can be sent without first asking the destination IMP if buffer space exists, provided it is received by the source IMP from the source HOST within a short period (e.g., 125 ms) after the HOST received the RFNM. If several requests messages are sent to the same HOST, the corresponding RFNM are sent one at a time in sequence. If a time-out occurs for any multi-packet message, a message is sent from the source IMP to the destination IMP to free the reserved buffers. This shows that it is not possible for a HOST to achieve high bandwidth on a given link unless it is prepared to send its next message before the time-out.

The space allocation scheme must explicitly attend to sequence control and discarding duplicate messages, which it accomplishes by using sequential message numbers. If each IMP uses a single message number for each other IMP

and delivers all messages in the order of insertion, priority messages could occasionally experience unnecessary delays in transit. This priority is due to different types of traffic that the network is supposed to handle. We can distinguish three types of traffic, as mentioned earlier in section 1.1. Therefore, to avoid unnecessary delay, the HOST must specifically mark a priority message as such, if the subnet can deliver it with priority. Of the several methods of implementing a priority handling mechanism, a simple way is for each IMP to keep two or three message numbers (depending on the number of priorities to be handled) for each other IMP. The second and third message numbers are provided to handle priority messages as marked by the HOST. These three message numbers correspond to three "pipes" between IMP's. Within each pipe messages are sent in order. However these three pipes may slide relative to each other, so that a message in highest priority pipe can be delivered before a message of lower priority, even though the highest priority message was transmitted later in time.

Thus we see that in a subnet having limited storage, reserving space at destination is a more powerful method for flow control than the link handling scheme used earlier (blocking and unblocking links), because it does not allow congestion and lock-up due to insufficient reassembly space. In general, a link handling scheme is costly in both program time and space. Again if a message is lost on any link due to subnet failure, that link experiences an

incomplete transmission; no other link is affected, though. Whereas in space reservation scheme an incomplete transmission between two IMPs introduces a short delay only and this seems to be acceptable since the penalty is not severe. Moreover this state of events is very infrequent.

The second lock-up of the end-to-end protocol in the ARPA net was due to shortage of 'reassembly blocks' and it was called 'christmas' lock-up [19]. The third problem was due to 'piggy-back' allocation of RFNM. Details of these problems with their remedies was discussed by Kleinrock and Opderbeck in [19].

The reservation schemes for all the lock-ups discussed so far resolves all reassembly problems, no doubt, but it does not use network bandwidth and reassembly space as efficiently as expected. Generally the limitation of the number of packets, in flight from the source to the destination, to a certain number and the requirement that a message cannot be transmitted unless the previous message has been fully acknowledged, considerably reduces the effective message transfer rate. Furthermore the reassembly buffer space is not utilized efficiently because the protocol requires that the entire message be first reassembled, before delivery to the HOST. If packets received in sequence were delivered to the HOST, some of the buffers could be released for use by other processes. Based on this principle a more flexible and efficient reassembly scheme is proposed in reference [31].

This is accomplished through the concept of "windows" of buffers provided at both transmitting and receiving ends. At the transmitting end, the window is used to store the outstanding, unacknowledged packets. At the receiving end, the window stores the out of sequence packets; packets received in sequence are immediately delivered to destination. The size of the windows is variable and can be optimally adjusted according to various network status variables. For example, the window size at the destination may be determined by the number of processes simultaneously using the reassembly buffer pool, and by the probability of receiving out of sequence packets. The size of the transmitting window may depend on the number of processes simultaneously requesting the buffers, the number of outstanding buffers required to obtain satisfactory message transfer performance, and the maximum input data rate that the network can accept at any given time without becoming congested. This concept of window size, N , will be used while defining congestion control mathematically in later section.

The use of variable size windows probably requires more computational overhead to determine optimal size and more end-to-end traffic overhead to negotiate the appropriate protocol, than the reservation scheme implemented in the ARPA net. This is the trade-off with the advantages offered by the window scheme:

- i) better utilization of reassembly buffers, taking advantage of the out-of-sequence statistics
- ii) fair allocation of reassembly buffers to the users, since window size can be reduced to accommodate more users.
- iii) better utilization of point-to-point bandwidth, obtained by enlarging the transmitting window
- iv) better control on the input rate, obtained by adjusting the transmitting window.

3.2 Local Control As Used In ARPA Net

In ARPA terminology this is also called node to node or IMP to IMP flow control. In the early ARPA net a very simple form of local flow control based on acknowledgement/time-out/retransmission was used between two adjacent IMP's. Each IMP contains N buffers, called store-and-forward (S/F) buffers, each of which may be placed on any output queue. These buffers are different from the reassembly buffers mentioned earlier. Each IMP also keeps a copy of each transmitted packet until either an acknowledgement is returned, in which case the copy is discarded, or a time-out occurs, in which case the packet is retransmitted. When a packet arrives without error and buffer space is available the packet is accepted and an acknowledgement is returned. If the packet is an error or no space is available, the packet is discarded and no acknowledgement is returned.

This extremely simple technique allows maximum autonomy between two IMPs. Unfortunately such a simple strategy can result in two problems:

- a) Direct Store and Forward Lock-up
- b) Indirect Store-and-Forward Lock-up.

The explanation of these lock-ups and proper remedies are given in [3].

In the second phase of the ARPA net operation a new technique has been adopted for IMP-to-IMP flow control. This improved the performance efficiency of the ARPA net by 10-20 percent as compared with the separate acknowledge/timeout/retransmission technique [32]. In this scheme, which is also used for HOST/IMP flow control, each physical network circuit is broken into eight logical channels in each direction. They are called subchannels abbreviated as SCN. Each SCN has a so called "odd-even" (OE) bit associated with it. It also has a source or Primary (P) IMP and a destination or Secondary (S) IMP which keep track of the state of the OE bit for each SCN. Every packet sent carries its SCN number its OE status and an acknowledgement byte. Retransmitted packets use the same SCN number, the same OE status and the same acknowledgement byte. Acknowledgements from S-IMP are returned piggybacked in every packet destined to the P-IMP using the acknowledgement byte (one bit per SCN) of the packet from P. At S, if the OE status bit of the received packet does not match the OE status of the associated SCN,

the packet is accepted and its OE status bit is complemented, otherwise the packet is considered to be a duplicate and is discarded. Any OE bits received at the P-IMP and not matching the status of the corresponding SCN, when this SCN is in use, imply that the packet has not been received correctly. When the S-IMP does receive the packet correctly the next version of the OE bit, which S sends back to P, will match the status of the particular SCN and the P-IMP marks this SCN free for assignment to another packet.

If the packet never arrives at S correctly, P will never see the matching OE bit. In that case P will retransmit the packet after a time out. When no data packets are available in one direction the null packets are used for acknowledgements. Null packets (i.e., sequence of bits with no information in it as identified by S) are not acknowledged back. To minimize the transmission delay one can use the following transmission ordering scheme in any link in one direction. Priority packets which have never been transmitted are sent first; next sent are any regular packets, which have never been transmitted; finally, if there are no new packets to send, previously transmitted packets which are unacknowledged are sent again. Under this scheme unacknowledged packets are periodically retransmitted which might create the so called "Race Problem", as pointed out in [32]. The strategy of continuously retransmitting a packet in the absence of other traffic introduced difficulties, which were not countered in the original system, which retransmitted only

after a long time-out. If an acknowledgement arrives for a packet, which is currently being retransmitted, the output routine must prevent the input routine from freeing the packet. Without these precautions, the header and data in the packet could be changed while the packet was being retransmitted, and unwanted conditions result when this composite packet is received at the other end of the line.

The control of S/F congestion is generally more complex than the reassembly buffer congestion. This is because reassembly congestion is an isolated problem and might occur in only one destination at a time, whereas S/F congestion phenomena can involve more than one node simultaneously. Kleinrock [19] and Gerla [29] tried to come up with other strategies for S/F congestion control different from that used in the ARPA net for better performance. One such method proposed is that each node controls its input packet rates on the basis of the information contained in some flow control tables which are circulated in the network. Such tables contain the information of the number of buffers available in each node, and are continuously updated, while circulating in the network. The information on the number of buffers available at a specific destination allows a source node to control the input rate of packets directed to that destination. In addition, the sum of buffers available over all the nodes provides an indication of the congestion of the entire network. This technique has the following

drawbacks: i) the control of input rates to prevent congestion is very critical, since an error in the computation of the maximum admissible rate could easily lead to a disaster, ii) the congestion control tables reflect a buffer availability situation which was existing in the network some time before.

The second proposal is to use the windowing technique similar to that discussed for control of reassembly buffers. Each source node is allowed a window of outstanding, unacknowledged packets to each destination or group of destinations. This is like a pool of numbers in practical situations. When a packet is sent to a destination it is assigned a free number. When the acknowledgement is returned to the source, the number is returned to the pool. If no numbers are available, the flow to that destination is interrupted. The size of the window is variable and can be adjusted on the basis of the flow control table information, for example, thus allowing some control on the input rate. The advantage of window techniques, with respect to the direct control of the input rate, is the additional protection offered by the acknowledgment mechanism. In the direct control case, if the flow control tables are faulty or slowly circulating, the source might accept a very high, steady stream of packets until the network is flooded. With the window, if the network starts becoming congested, round trip delays increase, acknowledgements are returned less frequently and the input rate is smoothly reduced and eventually stopped.

Anyway these strategies should be tested through simulation or mathematical analysis to verify their effects, before they are put in operation. In the next section, we will review the analytic works with rigorous mathematical definitions and derivations.

3.3 An Analytic Approach to End-to-End Congestion Control

In this section the concept of end-to-end congestion control as applied to a message switched network will be presented. Performance criteria will be defined and evaluated with and without congestion control, using analytic techniques [20].

If a sender tries to send a message he creates a "logical link" i.e., a unidirectional logical connection, from source to destination via the intermediate nodes. Messages entering the network are placed in the first queue in their source node and then pass through the intermediate queues of the "logical link" from source to destination. These messages are called link customers. There are other kinds of messages, called external customers. These customers are due to the fact that other source destination pairs may use the facilities of the logical link in question. External customers may use just one queue and one transmission line of the logical link or they may use several (Fig. 3.3.1). No distinction will be made between these external customers. All will be incorporated independently at each queue. There might be three types of external customers at a stage of the logical link i) The ones coming from adjacent queues which

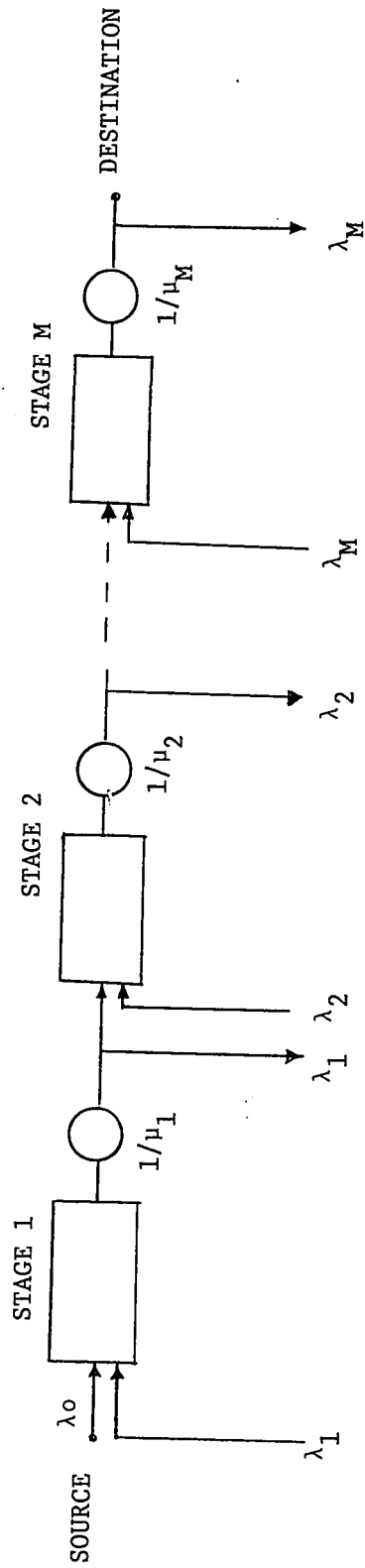


FIGURE 3.3.1: A QUEUEING MODEL OF A LOGICAL LINK [20]

are a part of the logical link in question ii) Others coming from the preceding queue in the logical link and iii) the ones coming in locally. The average arrival rate of these customers at each node i of the link will be taken as λ_i . Since the external customers which proceed to the $(i + 1)$ st queue from the i th queue are already included in the external arrivals at that queue, all external customers in queue i will be assumed to leave after service. Thus we have two interesting sets of customers:-

i) link customers and ii) external customers. Control schemes are applied to the link customers in order to limit the effect they have on external customers. For a link there is one source of link customers, i.e. the first queue of the link. Let their average arrival rate be λ_0 . Each message corresponding to either link or external customers has a length which will be assumed to be selected independently at each node from the same exponential distribution, and with mean the same for all messages arriving at that stage. Therefore the average transmission time for all customers at the i th stage is the same, $1/\mu_i$. After these assumptions the model for a logical link of M stages is shown in Figure 3.3.1. There are M stages in the link and link customers are shown to proceed from one stage to the next. External customers enter at each stage, are served at that stage and exit from the system. Service at each stage is first-come-first-served except when modified by a control scheme. In the next subsections we will see how this queuing network is

changed, when control schemes are applied. Now to evaluate and compare flow control schemes Pennotti in [20] defined the following parameters: i) congestion C ii) percentage improvements I iii) blocking probability P_B . The Mathematical interpretation is given as:

$$C_i = \frac{T_i - T_i | \text{no link customer}}{T_i | \text{no link customer}} \quad (3.3-1)$$

Where T_i is the average waiting time of the external customers at the i th node. C_i is the number of times longer the waiting time of external customers is, because of the presence of link customers. By using Little's result [33] we can write:

$$E(m_i) = \lambda_i T_i \quad (3.3-2)$$

where $E(m_i)$ = expected value of the number of external customers in the i th queue.

Putting (3.3-2) into (3.3-1) we get:

$$C_i = \frac{E(m_i) - E(m_i) | \text{no link customer}}{E(m_i) | \text{no link customer}} \quad (3.3-3)$$

Without the presence of link customers the queue is just a simple M/M/1 queue [34] with arrival rate λ_i and service time $1/\mu_i$ and

$$E(m_i) | \text{no link} = (\lambda_i / \mu_i - \lambda_i) \quad (3.3-4)$$

So,

$$C_i = \frac{(\mu_i - \lambda_i) E(m_i) - \lambda_i}{\lambda_i} \quad (3.3-5)$$

This is the final expression for the stage congestion at stage (node) i . It represents the increase in the waiting

time of external customers at that stage. This stage congestion C_i can be averaged over all the stages, i.e.

$$C = \frac{\sum_{i=1}^M \lambda_i C_i}{\sum_{i=1}^M \lambda_i}$$

$$= \frac{\sum_{i=1}^M [(\mu_i - \lambda_i) E(m_i) - \lambda_i]}{\sum_{i=1}^M \lambda_i} \quad (3.3-6)$$

This is the average number of times longer the waiting time of all external customers due to the presence of link customers.

The average congestion C , while measuring the effect of link customers on the waiting time of external customers, does not answer the effect of different controls. To answer this, I is defined as percent reduction in the waiting time of external customers in the controlled link from what it would be if no controls are used, i.e.

$$I_i = \frac{T_i | \text{no control} - T_i}{T_i | \text{no control}} \times 100\% \quad (3.3-7)$$

Making use of (3.3-2) we can write

$$I_i = \frac{E(m_i) | \text{no control} - E(m_i)}{E(m_i) | \text{no control}} \times 100\% \quad (3.3-8)$$

For two inputs λ_i and λ_o at each stage without any control, $E(m_i)$ can be written as:

$$E(m_i) \Big|_{\text{no control}} = \frac{\lambda_i}{\mu_i - \lambda_i - \lambda_o} \quad (3.3-9)$$

So

$$I_i = \frac{\lambda_i - (\mu_i - \lambda_i - \lambda_o) E(m_i)}{\lambda_i} \times 100\% \quad (3.3-10)$$

Similarly as C, I can be defined as:

$$I = \frac{\sum_{i=1}^M (\lambda_i - (\mu_i - \lambda_i - \lambda_o) E(m_i))}{\sum_{i=1}^M \lambda_i} \times 100\% \quad (3.3-11)$$

The last parameter P_B for link customers is defined as the probability that a link customer arriving at queue 1 finds N link customers in the system and is turned away. This is so defined in order to evaluate different congestion control schemes. For these schemes it is necessary to weigh the limiting effect on congestion against the cost that must be paid by the link customers. Clearly, C is minimized and I maximized by not permitting any link customers to enter the system, but this is not a satisfactory solution. One way to measure this cost of controls to the link user is to consider the system to be a loss system, i.e., to have finite blocking probability P_B . This means that, if the source tries to enter messages at a rate λ_o , it will find that the average rate of entry is reduced by the factor

$(1 - P_B)$. Alternatively, if it has a large number of messages which it wishes to enter, the capacity of the link, or the number of messages per second it can enter, will be reduced by the same factor. This is due to the fact that it must sometimes wait to enter its message because the link is blocked. The effect of the repeated attempts to enter blocked messages is ignored here in order to keep the derivation manageable.

It has been shown by Pennotti in [20] that a queueing model of a logical link employing end-to-end congestion control is equivalent to a closed cyclic queueing model. If we allow N customers only in a logical link, as it is done in end-to-end congestion control of M stages, this is equivalent to a closed system of N customers cycling through $M + 1$ stages, as shown in Figure 3.3.2. Both systems of figure 3.3.1 and figure 3.3.2 are the same, as far as the arrival, service and departure of external customers are concerned. In addition, arrivals of link customers to queue 1 in the closed system occur with Poisson rate λ_0 , whenever queue $M + 1$ is not empty. The condition that queue $M + 1$ is empty is the same as the condition that there are N link customers in queue 1 through M since N link customers cycle through the system. Therefore, the arrival of link customers to queue 1 in the closed system occurs by precisely the same process as in the open system with end-to-end congestion control. There are established results for queues of this type [35],

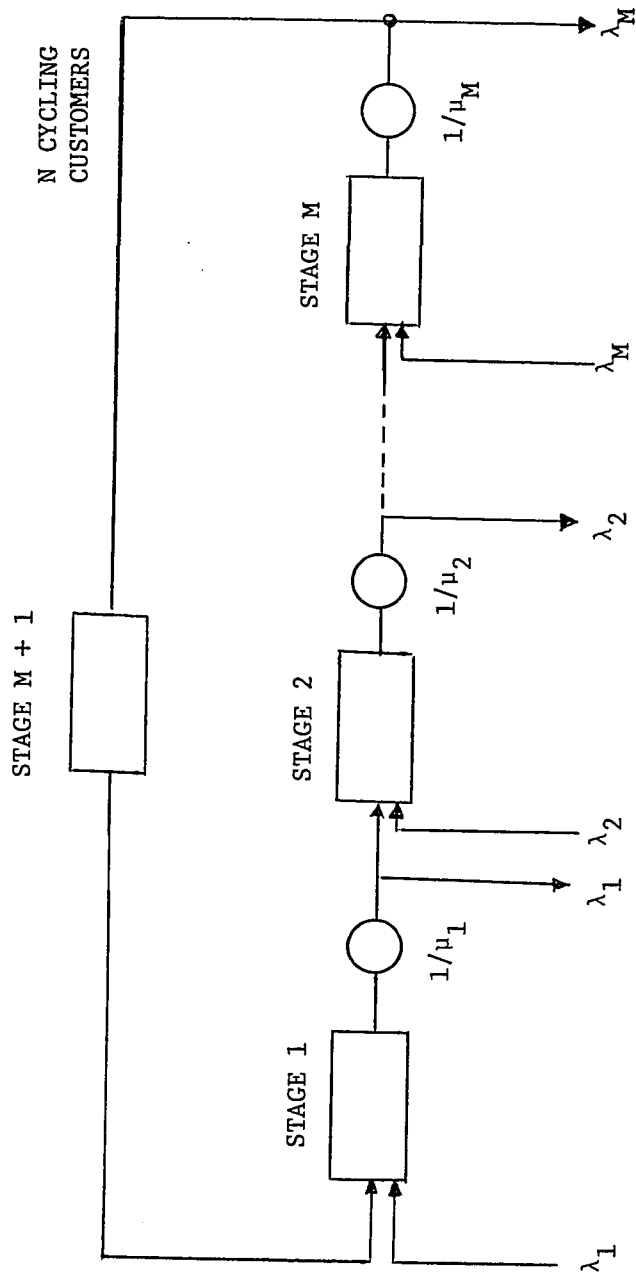


FIGURE 3.3.2: AN EQUIVALENT CLOSED SYSTEM FOR END TO END CONTROL [20]

[36], [46]. But none of them considered the external arrivals. The state of the closed queuing network of Figure 3.3.2 can be described by $2M$ values, the number of link customers on queues 1 through M and the number of external customers on the same M queues. The remaining value, the number of link customers on queue $M + 1$, can be found from these, since the total number of link customers on queues 1 through $M + 1$ is known to be N . Let n_i be the number of link customers at node i and m_i be the number of external customers at node i , where i goes from 1 to M . Then the $2M$ dimensional vector (\bar{n}, \bar{m}) where vector

$$\bar{n} = (n_1, n_2, \dots, n_M) \quad (3.3-12)$$

and vector

$$\bar{m} = (m_1, m_2, \dots, m_M) \quad (3.3-13)$$

completely defines the state of the system. We are now interested in finding the state probability $P(\bar{n}, \bar{m})$. This can be found by noting that under steady-state conditions the probability of entering a state must equal the probability of leaving it.

The solution $P(\bar{n}, \bar{m})$ is given by the following relation [20]:

$$P(\bar{n}, \bar{m}) = P(\bar{0}, \bar{0}) \prod_{i=1}^M \left(\frac{\lambda_0}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!}$$

where n_i and m_i are number of link and external customers respectively at a particular stage i . $P(\bar{0}, \bar{0})$ is the probability that all the queues 1 through M are empty. It can be found from the normalizing condition

$$\sum_{\text{all possible states } (\bar{n}, \bar{m})} P(\bar{n}, \bar{m}) = 1 \quad (3.3-14)$$

all possible
states (\bar{n}, \bar{m})

From (3.3-14), $P(\bar{o}, \bar{o})$ can be derived to be:

$$P(\bar{o}, \bar{o}) = \frac{1}{Z_M^N} \prod_{i=1}^M (1 - \lambda_i / \mu_i) \quad (3.3-15)$$

where,

$$Z_M^N = \sum_{\text{all partitions of } n \leq N \text{ in } M} \prod_{i=1}^M \left(\frac{\lambda_i}{\lambda_i - \mu_i} \right)^{n_i} \quad (3.3-16)$$

So,

$$P(\bar{n}, \bar{m}) = \frac{1}{Z_M^N} \prod_{i=1}^M \left(\frac{\lambda_i}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!} \left(1 - \frac{\lambda_i}{\mu_i} \right) \quad (3.3-17)$$

From (3.3-17) we can think of two special cases; i) the case of a cyclic system with no external arrivals, as solved in [36]. That system is the same as the present one with $\lambda_i = 0$, $i = 1, 2, \dots, M$. From (3.3-17) the only states with non-zero probability would be those with $\bar{m} = \bar{o}$. The result is

$$P(\bar{n}, \bar{o}) = \frac{1}{Z_M^N} \prod_{i=1}^M \left(\frac{\lambda_i}{\mu_i} \right)^{n_i} \quad (3.3-18)$$

where,

$$Z_M^N = \sum_{\text{all partitions of } n \leq N \text{ in } M} \prod_{i=1}^M (\lambda_i / \mu_i)^{n_i} \quad (3.3-19)$$

all partitions
of $n \leq N$ in M

This is really Koenigsberg's result [36]. ii) The other special case is the one in which there are no cycling cust-

omers, or $\lambda_0 = 0$. This corresponds to a series of M independent $M/M/1$ queues. From (3.3-17), the only states with non-zero probability are those with $\bar{n} = \bar{o}$. The result is

$$P(\bar{o}, \bar{m}) = \prod_{i=1}^M \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \left(1 - \frac{\lambda_i}{\mu_i} \right) \quad (3.3-20)$$

since $Z_M^N = 1$ in this case.

From (3.3-17) the following quantities can be found:

$$1) \quad P_B = \sum_{\substack{\text{all states} \\ (\bar{n}, \bar{m}) \text{ with} \\ n = N}} P(\bar{n}, \bar{m}) = 1 - \frac{Z_M^{N-1}}{Z_M^N} \quad (3.3-21)$$

$$2) \quad E(n_k) = \text{average number of link customers in queue, } k$$

$$= \sum_{\substack{\text{all possible} \\ \text{states } (\bar{n}, \bar{m})}} n_k P(\bar{n}, \bar{m})$$

$$= \frac{\lambda_k}{Z_M^N} \frac{d}{d x_k} Z_M^N \quad (3.3-22)$$

where,

$$x_k = \frac{\lambda_0}{\mu_k - \lambda_k} \quad (3.3-23)$$

$$3) \quad E(m_k) = \text{average number of external customers in queue, } k.$$

$$= \sum_{\substack{\text{all possible} \\ \text{states } (\bar{n}, \bar{m})}} m_k P(\bar{n}, \bar{m})$$

$$= \frac{\lambda_k}{\mu_k - \lambda_k} (1 + E(n_k)) \quad (3.3-24)$$

If we are now interested in finding the probability distribution of link customers only, i.e. $P(\bar{n})$, this can be written as

$$\begin{aligned}
 P(\bar{n}) &= \sum_{\text{all states } \bar{m}} P(\bar{n}, \bar{m}) \\
 &= \frac{1}{Z_M^N} \prod_{i=1}^M x_i^{n_i} \quad (3.3-25)
 \end{aligned}$$

where,

$$x_i = \left(\frac{\lambda_0}{\mu_i - \lambda_i} \right) \quad (3.3-26)$$

This is the identical result found by Koenigsberg for cyclic systems with no external arrivals. He took λ_i to be λ_0 / μ_i .

It is therefore observed that, as far as the performance of the system for link customers is concerned, the effect of external customers is to reduce the average service rate of each queue by the average external arrival rate. Also, considering the value of $E(m_k)$ from (3.3-24), the term $(\lambda_k / \mu_k - \lambda_k)$ is just the average value of the number of external customers, which would be found in the k th queue, if there were no link customers. Therefore, the effect of the link customers on the average number of external customers in the k th queue is to increase it by the factor $(1 + E(n_k))$. From the above observation one would solve the problem of Figure 3.3.2 more easily - just by replacing each μ_i by $(\mu_i - \lambda_i)$ and solving the resulting system. This result will, in effect, be used in the next sub-section where a closed form solution with the original consideration is very difficult.

3.4 Analysis of Local Congestion Control [20]

In this subsection we are considering the local congestion control scheme in the same queueing model of a logical link as shown in Figure 3.3.1. The restriction placed on this model by local congestion control is that no more than N_i link customers may reside on the i th queue at one time. If there are N_i link customers present, the $(i - 1)$ st stage may not begin servicing another link customer until one departs from the i th queue. External customers may be served in the meantime and customers which arrive at the first queue, when there are already N_i present, are assumed turned away, never to return [20].

An approximate solution was attempted first and then a simple example, whose exact solution can be found, was given to see the difference between the approximate solution and the exact solution [20]. The exact solution is not, in general, possible because no equivalent model, which incorporates the blocking mechanism as a basic feature, such as cyclic model of the previous section, has yet been constructed [20], even if it is assumed that there are no external customers in the link. There were two assumptions for the approximate solution. The first one was that as far as service to link customers is concerned, the external customers can be omitted, if the exponential service rate at each stage is reduced by the corresponding average arrival rate - a useful result

suggested by Pennotti [20]. Under this assumption the local congestion control reduces to a series of finite queues, as shown in Figure 3.4.1. Under steady state conditions, the equation for the probability distribution of a single stage of Figure 3.4.1 can be written as:

$$\text{Pr (leaving state } J) = \text{Pr (entering state } J) \quad (3.4-1)$$

where J goes from 0 to N_i . This can further be written as:

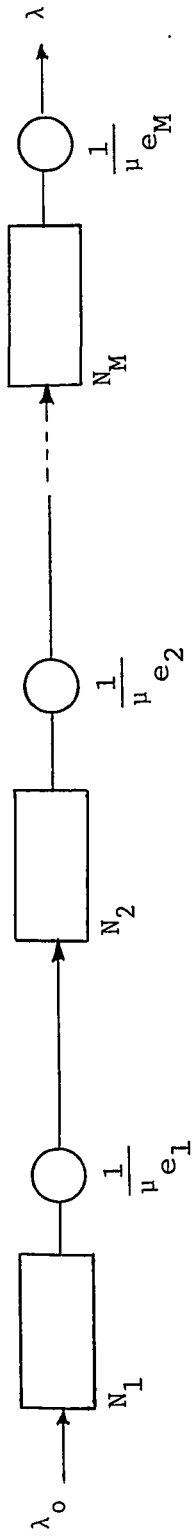
$$\begin{aligned} & [\text{Pr (departure} | J + \text{Pr (arrival} | J)] P(J) \\ & = \text{Pr (departure} | J + 1) P(J + 1) + \text{Pr (arrival} | J - 1) \\ & \quad P(J - 1) \end{aligned} \quad (3.4-2)$$

where $P(J)$ is the probability of state J and that is to be found.

But if $J < 0$ or $J > N_i$, $P(J) = 0$.

The second assumption made was regarding the conditional probability $\text{Pr (next stage not full} | J)$, which was approximated by the unconditioned probability that the next stage is not full.

The arrival probability is also a complicated quantity to be found because the probability of an arrival at one stage equals the probability of a departure from the previous stage and once again these two quantities are not independent. Therefore, the simplest possible arrival mechanism was assumed, that of Poisson arrivals, with the condition that arrivals which occur when the queue is full are turned away. The rate, designated λ_e , will be found from the throughput by the relation [20]:



$$\mu e_i = \mu_i - \lambda_i$$

FIGURE 3.4.1: MODEL FOR LINK CUSTOMERS WITH LOCAL CONTROL [20]

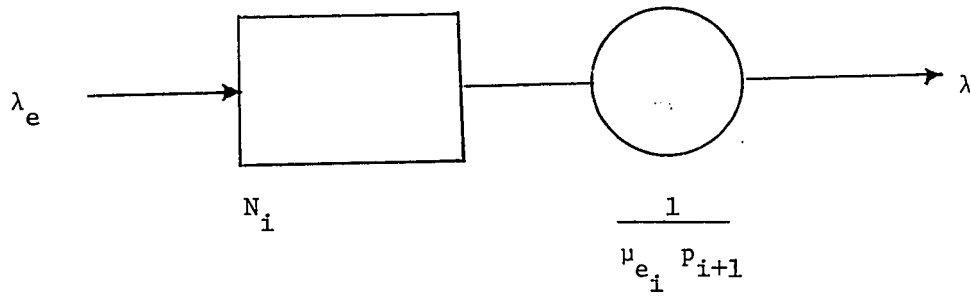


FIGURE 3.4.2: MODEL FOR SINGLE NODE WITH LOCAL CONTROL [20]

$$\lambda = \lambda_e (1 - P(N_i)) \quad (3.4-3)$$

Because $(1 - P(N_i)) = p_i =$ probability that the stage i is not full. If we now assume the probability that the stage $i+1$ is not full to be p_{i+1} , we can write the probability balance equation for any node i as follows [20]:

$$P(J) (\lambda_e + \mu_{e_i} p_{i+1}) = P(J+1) \mu_{e_i} p_{i+1} + P(J-1) \lambda_e \quad (3.4-4)$$

Equation (3.4-4) is the same as that which would be obtained if a single finite queue with Poisson arrivals at rate λ_e and exponential service with rate $\mu_{e_i} p_{i+1}$, as shown in Figure 3.4.2, were analyzed. The solution is known [34] to be

$$P(j) = \frac{(\lambda_e / \mu_{e_i} p_{i+1})^j}{\sum_{k=0}^{N_i} (\lambda_e / \mu_{e_i} p_{i+1})^k} \quad (3.4-5)$$

$$= \frac{(\lambda / \mu_{e_i} p_i p_{i+1})^j}{\sum_{k=0}^{N_i} (\lambda / \mu_{e_i} p_i p_{i+1})^k} \quad (3.4-6)$$

$$\text{as } \lambda_e = \frac{\lambda}{(1 - P(N_i))} = \frac{\lambda}{p_i}$$

p_i can be solved from the following polynomial, whose coefficients are known if λ , μ_{e_i} and p_{i+1} are known:

$$p_i^{N_i} - \sum_{k=0}^{N_i-1} p_i^k \left(\frac{\lambda}{\mu_{e_i} p_{i+1}} \right)^{N_i-1-k} \left(1 - \frac{\lambda}{\mu_{e_i} p_{i+1}} \right) = 0 \quad (3.4-7)$$

After p_i is found, $P(J)$ is completely determined, as also the blocking probability and the average stage queue length.

Pennotti describes a method in [20] for finding p_i from p_{i+1} starting from the end of the queue of Figure 3.4.1. There is a simple relation between λ , λ_0 and p_1 at the 1st queue which is written as

$$\lambda - \lambda_0 p_1(\lambda) = 0 \quad (3.4-8)$$

At the end of the queue of M stages $p_{M+1} = 1$ because there is no blocking in the next stage. By using equation (3.4-7), p_M can be found with some arbitrary throughput λ . By this way we can go up to stage 1 to check relation (3.4-8). If (3.4-8) is not satisfied for some λ_0 the procedure is repeated. There are other simple procedures to solve (3.4-7) as mentioned in [20].

Two examples were considered by Pennotti [20], whose exact solutions could be found in order to compare the results of these approximation for local flow control. He justified his assumptions for these two systems and then verified his results by simulation for more complicated systems. We will base on his assumptions and verifications to compare our results with those of Schwartz [37] in later sections.

3.5 Random Routing

In this section we will define random routing and show the rationale for considering this in our analysis. Random routing rules are defined as those procedures in which the choice

as to the next node to visit is made according to some probability rule. For example, one such procedure might be to choose the next nodes out of idle channels with equal probability. Kleinrock in [10] did some analytic as well as simulation work on random routing to compare it with other routing procedures, such as fixed routing and alternate routing. He also considered these routing procedures constrained by topology. There were two main reasons for him to consider the random routing. The first of them was that for a particular class of random routing procedures, it is easy to solve for the expected number of steps that a message must take before arriving at its destination. The second was that it is easy to derive an expression for the expected time that a message spends in the net. These results he used to prove some of his main theories of communication nets under some restrictive assumptions.

Kleinrock in [10] considered message delay to be the measure of network performance and investigated the effects on performance by variations in channel capacity assignment, routing procedure and topological network structure. In his simulation experiments he has been able to demonstrate some interesting behaviour in regard to the design parameters, while holding fixed the total channel capacity assigned in the net. It is natural to consider message delay as a measure of network performance and Kleinrock [10] has shown the effects of capacity assignment, routing procedure and

topological network structure on such performance measure. Out of his many findings, the followings are worth mentioning.

- i) The square root channel capacity assignment gives better performance, as compared with a number of other channel capacity assignments
- ii) The throughput of a straight forward fixed routing procedure with a square root capacity assignment surpasses that of a simple alternate routing procedure.
- iii) The alternate routing procedure adapts the internal traffic flow to suit the capacity assignment (i.e., the bulk of the message traffic is routed to the high-capacity channels). This effect is especially noticeable and important in the case of a poor capacity assignment, which may come about, owing to uncertainty or variation in the applied message traffic.

Most of these findings were based on simulation and so one should try to establish an analytic basis for these results. This is what we are aiming for in our next sections. Because of unpredictable behaviour of alternate routing, it becomes intractable to handle it mathematically and so our main interest will be random routing. We can look at random routing to be very simple, both in conception and in realization in practical system. Systems operating under random routing become relatively insensitive to changes in the structure of the network. This does not prevent the random routing proce-

ture to function, even if some of the channels disappear. Directory informations need not be circulated in the case of hostile environment, in which changes in the network take place continuously. If it were necessary to circulate changing information around the network for each node, the network would easily become congested with directory data alone, thus leaving no transmission capability for message traffic.

There are, however, some in-born disadvantages to random routing procedures. The major drawback is that it does not take into account certain available information. According to the topological structure of the network it is natural that for certain messages destined for a particular node there are certain paths preferred over others. Random routing procedures never utilize this information. As a result, messages are sometimes looped around and may be fortunate enough to reach the destination. This effect increases the internal traffic that the network is to handle; consequently, the external traffic that may be applied to the net is greatly reduced. This leads to the fact that the time that a message spends in the network is increased, which reduces the grade of service to the user of the system.

We will prove some of the stated results by mathematical analysis. Our comparison will be mostly with respect to fixed routing as analyzed by Schwartz [37]. In the absence of mathematical analysis for any other involved routing

scheme, one can feel that these results will serve as a measure of the quality of performance for design purposes.

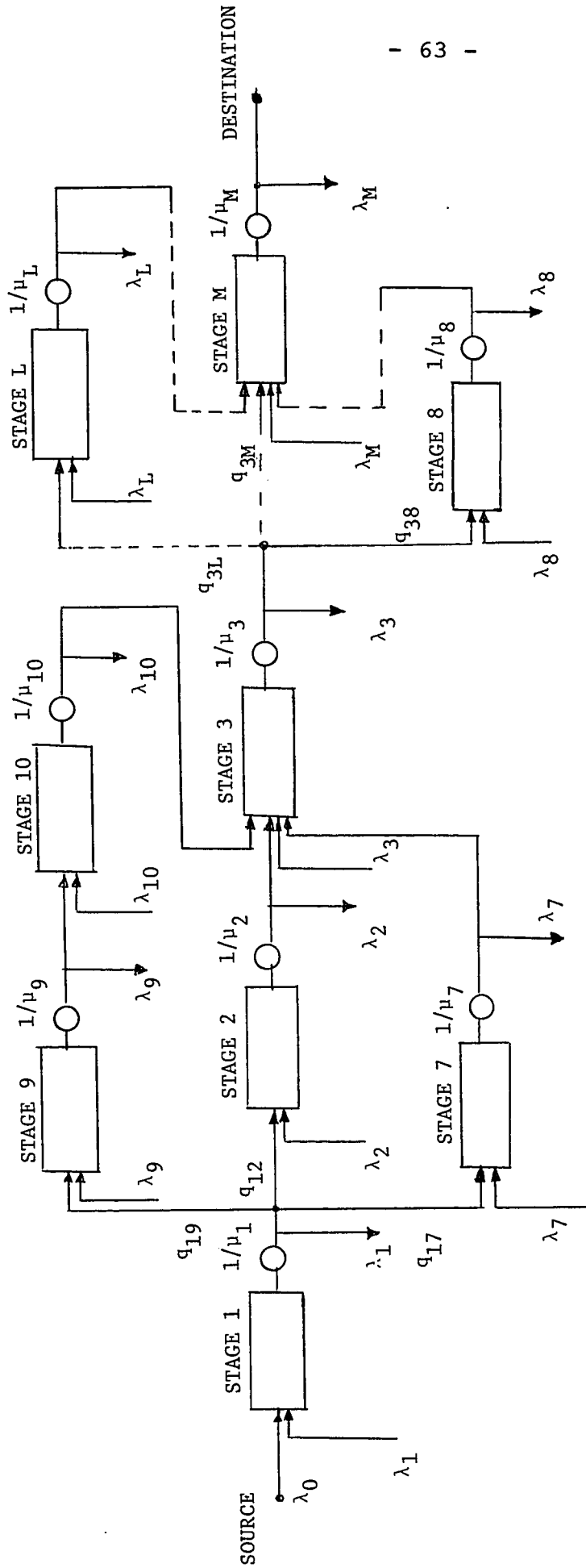
CHAPTER - 4

NETWORK WITH END-TO-END CONGESTION CONTROL AND RANDOM ROUTING

4.1 Introduction

Several packet-switched data communication networks are commercially operational or about to be operational at the present time. As it is well-known, with packet-switching a greater utilization of line can be achieved for a given delay objective per packet. This ensures acceptable response time, particularly for short messages, along with the capability to deal with messages of arbitrary length

Information-bearing packets in a packet-switched network are routed from a source node to a destination node via intermediate nodes. The choice of intermediate nodes is dynamically variable and can be represented by a probability matrix at each node. Transfer of information is effected through a logical link. We define a logical link as the unidirectional logical connection between the source and destination nodes. One end communicates with the other end by sending a sequence of signals over a given path. These two ends agree upon this path for use by means of protocols. A link for a network with no routing capability looks like a system of queues connected in series from the source to the destination as shown in Figure 3.3.1, - whereas one with random routing capability can be considered as a network of queues connected together, as shown in Figure 4.1.1.



TOTAL STAGES = M
 $\sum_j q_{ij} = 1$ at each branching
 λ_0 = Link arrival rate
 λ_i = external arrival rate
 q_{ij} = probability of branching.

FIGURE 4.1.1: LOGICAL LINK FOR PACKET-SWITCHING WITH RANDOM ROUTING

In this illustration, there are M stages between node-pair 1 and M which constitutes the source-destination pair of a link. In general, a logical link as defined will consist of upto M stages for a M node network. The important difference between the two cases is the existence of probabilistic routing in the case of the latter, against no alternate routing in the case of the former.

Any resource shared network can have congestion when there is uncontrolled demand. This congestion may be defined as the decrease in the quality of service to one customer because of the presence of another, following Pennotti and Schwartz [37]. The purpose of the end-to-end congestion control technique is to ensure that the equivalent utilization factor ρ of the queuing network model does not go beyond certain critical value [34].

This is done by specifying a maximum allowable number of packets in each logical link.

Pennotti modelled a message-switched network with end-to-end congestion control (i.e. by specifying a maximum number of messages in a logical link) as a closed queueing network for one source-destination pair [20], [37]. He then defined appropriate performance measures and analyzed the system using both end-to-end and local congestion control. These analytical results were then compared with the ones obtained by simulation. There are not many other analytical work done in this area and Pennotti's work gives considerable insight into the behaviour

of a communication network with congestion control.

In communication networks, alternate or adaptive routing is provided to prevent the network from partitioning under certain node or link failure conditions, increase throughput and, in general, enhance the network performance. With fixed routing, network partitioning is possible with a single link failure even in fully connected networks.

Alternate routing or adaptive routing is, therefore, generally employed. The random routing or probabilistic routing approach has been adopted for analysis in this work in order to shed some light on the behaviour of alternate or adaptive routing schemes. The alternate routing and adaptive routing schemes are not easily amendable to mathematical modelling for analysis. It is believed that inclusion of random routing in the mathematical model will enhance its usefulness in providing an analysis indicative of alternate routing performance.

In this chapter we extend Pennotti's work to the more general case of a communication network with end-to-end congestion control and random routing. A queueing model is developed for this purpose. The analysis of this type of queueing network model is unfortunately computationally awkward. The results obtained will be useful for the design of a message or a packet-switched network using an end-to-end congestion control and random routing [38].

4.2 A Queueing Model For Congestion Control

Any user (or customer) of a logical link has messages and hence packets to be transferred from one end of the link to the other. These packets can be routed through different sections of the logical link. In this section a single logical link with different possible routes, as observed from the sending end, will be modelled by a queueing network. Consider the sending end, where packets are being generated and admitted to the network for onward transmission to the receiving end. In the model indicated here, different routes of a logical link are possible for the transmission of packets between the same source - destination pair. The logical link with all its routes, however, is supposed to be fixed for the time of interest for which we are analyzing the system. As indicated earlier a similar system with no routing capability has been analyzed by Pennotti [20]. The only difference between his model and our model is that in the former successive messages sent over a logical link are transmitted through the same system of tandem queues, whereas in our model packets could be transmitted from one end of the link to the other using different routes, depending on the probability of a route being available, as shown in Figure 4.1.1. When the packet enters the network, it joins the first queue of the logical link and waits its turn to be served. After being served at the first stage it joins the next stage according to the assigned routing probability. This process is repeated

at all successive nodes until it finally arrives at the destination node. There it is finally processed and sent to the sink. Therefore, as seen by the sending end, the link looks like a system of series - parallel queues connected by transmission lines extending from the source node to the destination node.

The packets transmitted by a particular source node, in transition to a particular destination, sharing different routes of a logical link will be called link packets or link customers after Pennotti and Schwartz [20], [37]. Besides these link packets, packets belonging to the other source destination pairs will also be using the communication lines of the logical link being examined. These packets are called external customers. Both link and external customers are processed at each node on a first-come-first-served basis. No distinction will be made between different kinds of external customers [20].

The arrivals of the packets at all the queues are assumed to be Poisson with an average rate of λ_0 for link customers at the source node and with an average rate of λ_i , $i = 1, 2, \dots, M$, for external customers. The service time for different packets in any queue will be assumed to be an exponentially distributed random variable. For the i th queue, the average of this random variable will be denoted by $1/\mu_i$. A model for a logical link resulting from the above assumptions is shown as a network of M queues in Figure 4.1.1. Link customers are routed from the source to the destination according to the

routing probabilities. External customers enter at each stage, are served at that stage and then exit from the system. Service discipline is first-come-first-served at each stage. The solution of a specific form of this network will be obtained in a later section so as to compare with previously published results. As discussed earlier the end-to-end congestion control places a limit on the number of link customers which can be present on the logical link at one time. If a link arrival should occur when there are already N link customers present, it is assumed to be lost. The link is thus modelled as a loss-delay system in which the throughput is less than the arrival rate λ_0 by a factor which is the blocking probability.

4.3 Performance Measure

Following Pennotti and Schwartz [20], [37] we will define the following three criteria for evaluating the end-to-end congestion control technique with random routing i) Congestion C_i , ii) Percentage Improvement I and iii) Blocking Probability, P_B . These criteria have been found useful in the evaluation of performance of congestion control techniques. The mathematical interpretation is given as:

$$C_i = \frac{T_i - T_i \mid \text{no link customer}}{T_i \mid \text{no link customer}} \quad (4.3-1)$$

where T_i is the average waiting time of external customers at the i th node. C_i is the fraction of time an external customer has to wait, because of the presence of link customers. By using Little's result and averaging over all the stages we get [20].

$$C = \frac{\sum_{i=1}^M \lambda_i C_i}{\sum_{i=1}^M \lambda_i} \quad (4.3-2)$$

$$= \frac{\sum_{i=1}^M [(\mu_i - \lambda_i) E(m_i) - \lambda_i]}{\sum_{i=1}^M \lambda_i} \quad (4.3-3)$$

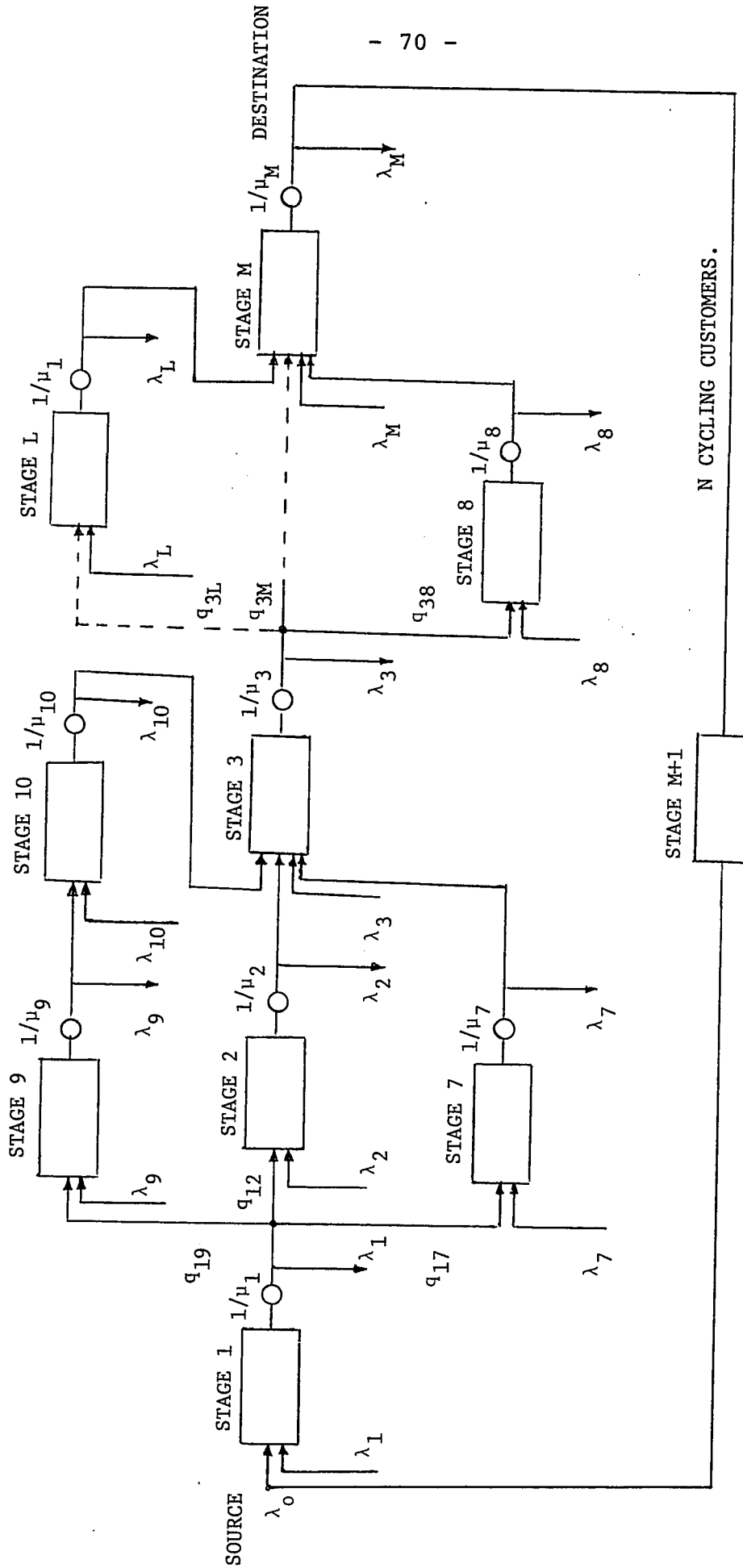
where $E(m_i)$ = expected queue length of external customers at stage i .

The average congestion C , while measuring the effect of link customers on the waiting time of external customers, does not show the effect of different controls. To answer this, I_i is defined as the percent reduction in the waiting time of external customers in the controlled link from what it would be if no control were used, i.e.

$$I_i = \frac{T_i | \text{no control} - T_i}{T_i | \text{no control}} \times 100\% \quad (4.3-4)$$

Averaging I_i similar to C over all stages we can define I as:

$$I = \frac{\sum_{i=1}^M [\lambda_i - (\mu_i - \lambda_i - \lambda_o) E(m_i)]}{\sum_{i=1}^M \lambda_i} \quad (4.3-5)$$



TOTAL STAGES = M+1
 $\sum_j q_{ij} = 1$ at each branching
 λ_0 = External arrival rate
 λ_i = Link arrival rate
 q_{ij} = Probability of branching

FIGURE 4.4.1: AN EQUIVALENT CLOSED SYSTEM FOR THE END-TO-END CONTROL

The last parameter P_B for link customers is defined as the probability that a link customer arriving at queue 1 finds N link customers in the system and is turned away. The significance of P_B can be analyzed as follows. Clearly, C is minimized and I is maximized by not permitting any link customers to enter the system, but this is not a satisfactory solution. One way of measuring this cost of controls to the link customer is to consider the system to be a loss system, i.e. to have a finite blocking probability P_B . This means that if the source tries to enter packets at a rate λ_0 , it will find that the average rate of entry is reduced by the factor $(1 - P_B)$. This is the price to be paid for using link congestion control. Mathematical derivation for P_B is given later in equation (4.4-10).

4.4 Equivalent Closed Queueing System and Its Analysis

Following the discussion of Gordon and Newell [35] the queueing network of Figure 4.1.1 with M stages, after employing the end-to-end congestion control, can be considered as the closed queueing network of Figure 4.4.1 with $M+1$ stages. Arrivals to queue 1 in the closed system occur with an average rate λ_0 , Poisson distributed. Queue $M+1$ is empty is the same as the condition that there are N link customers in queue 1 through M , since the N link customers cycle through the system. Therefore, the arrival of the link customers to queue 1 in the closed system occurs (when $(M+1)$ is not empty) by precisely the same process as in the open system with the end-to-end congestion control.

Thus the two systems are equivalent as far as link customers are concerned. External customers arrive at each node, Poisson distributed, and leave the system after being served at that node. Link customers and external customers at each node constitute different classes of customers (one being in closed network and the other in open network) as analyzed by Baskett, Muntz et al [39], [46].

Queueing systems similar to Figure 4.4.1 have been studied in [35], [40], and [41]. But these systems did not consider external customers. Pennotti [20] analyzed such a closed queueing system with external customers, but did not account for branching or probabilistic routing.

We will follow a similar approach. The application of this technique is illustrated by applying it to the arbitrary network of Figure 4.4.1. Here stage 1 is the source and stage M is the destination. There are total M stages in this network of the logical link and customers after being served in the first stage are routed to other stages according to some random probabilities before they reach the destination, M.

The state of the closed queueing network of Figure 4.4.1 is composed of 2M variables namely the number of link customers on queues 1 through M and the numbers of external customers in those M queues. If we denote by $P(\bar{n}, \bar{m})$, the probability of any state we can obtain directly the steady state equilibrium equations as in [20]. Here, $\bar{n} = (n_1, n_2, \dots, n_M)$ and

$\bar{m} = (m_1, m_2, \dots, m_M)$ and $\sum n_i \leq N$. The steady state equilibrium equations are obtained by equating the probability of leaving state (\bar{n}, \bar{m}) and the probability of entering state (\bar{n}, \bar{m}) as follows:

$$\begin{aligned}
 & [\lambda_0 + \sum_{i=1}^M \lambda_i + \sum_{i=1}^M \mu_i] P(\bar{n}, \bar{m}) \\
 &= \lambda_0 P(n_1-1, n_2, \dots, n_M, \bar{m}) + \sum_{i=1}^M \lambda_i P(\bar{n}, m_1, m_2, m_i-1, \dots, m_M) + \\
 & \quad \frac{n_{M+1}}{n_M+1+m_M} \mu_M P(n_1, n_2, \dots, n_{M+1}, \bar{m}) + \sum_{i=1}^{M-1} \sum_{\substack{j \neq i \\ j=1}}^M q_{ij} \frac{n_i+1}{n_i+1+m_i} \\
 & \quad \mu_i P(n_1, n_2, n_i+1, \dots, n_j-1, \dots, n_M, m) + \sum_{i=1}^M \frac{m_{i+1}}{n_i+1+m_i} \mu_i \\
 & \quad P(\bar{n}, m_1, m_2, \dots, m_i+1, \dots, m_M) \tag{4.4-1}
 \end{aligned}$$

Solving the equation (4.4-1), we can get the solution for $P(\bar{n}, \bar{m})$ as:

$$P(\bar{n}, \bar{m}) = P(\bar{0}, \bar{0}) \prod_{i=1}^M \left(\frac{y_i \lambda_0}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!} \tag{4.4-2}$$

where,

λ_0 = arrival rate

y_i = is the solution of the following matrix equation

$$A^T y = y \tag{4.4-3}$$

where $y = [y_1, y_2, \dots, y_M]^T$

The elements of matrix A are q_{ij} , i.e., the probabilities of branching from node to node, $\sum_j q_{ij} = 1$ for all $i = 1, 2, \dots, M-1$. This can be verified by direct substitution into the balance equation. The term $P(\bar{0}, \bar{0})$ is the probability that all queues,

1 through M are empty. It can be calculated from the normalizing condition that the summation of all state probabilities equals to unity. The formation of probabilistic loops depend on the choice of the elements for the matrix A. As we are analyzing a logical link from the source to the destination, it is natural to choose the elements of A such that the routes from the source to the destination do not form loops. But the analysis does not lose its basis even if we allow loop formation when choosing matrix A. Equation (4.4-2) can be further simplified as:

$$P(\bar{n}, \bar{m}) = \frac{1}{Z_M^N} \prod_{i=1}^M \left(\frac{y_i \lambda_o}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!} \left(1 - \frac{\lambda_i}{\mu_i} \right) \quad (4.4-3)$$

where,

$$Z_M^N = \sum_{\substack{\text{all partitions} \\ \text{of } n \leq N \text{ in } M}} \prod_{i=1}^M \left(\frac{y_i \lambda_o}{\mu_i - \lambda_i} \right)^{n_i} \quad (4.4-4)$$

The computation of Z_M^N is somewhat complex, even for relatively simple networks.

The complexity grows depending on whether $x_i = \left(\frac{y_i \lambda_o}{\mu_i - \lambda_i} \right)$ is uniform (i.e. all x_i 's are equal), non-uniform (i.e. all x_i 's are different) or arbitrary (i.e. some are uniform and some non-uniform), [42], [43]. Now the following quantities can easily be calculated from $P(\bar{n}, \bar{m})$ [20].

$$\begin{aligned}
 1) \quad E(n_k) &= \text{expected link-queue length} \\
 &= \sum_{\substack{\text{all possible} \\ \text{states } (\bar{n}, \bar{m})}} n_k P(\bar{n}, \bar{m}) \quad (4.4-5)
 \end{aligned}$$

$$= \frac{x_k}{Z_M^N} \frac{d}{dx_k} Z_M^N \quad (4.4-6)$$

$$\begin{aligned}
 2) \quad E(m_k) &= \text{expected queue length of external customers} \\
 &= \sum_{\substack{\text{all possible} \\ \text{states } (\bar{n}, \bar{m})}} m_k P(\bar{n}, \bar{m}) \quad (4.4-7)
 \end{aligned}$$

$$= \frac{\lambda_k}{\mu_k - \lambda_k} [1 + E(n_k)] \quad (4.4-8)$$

$$3) \quad P_B = \text{blocking probability}$$

$$= \sum_{\substack{\text{all states } (\bar{n}, \bar{m}) \\ \text{with } n = N}} P(\bar{n}, \bar{m}) \quad (4.4-9)$$

$$= 1 - \frac{Z_M^{N-1}}{Z_M^N} \quad (4.4-10)$$

Proofs of these derivations are given in Appendix A

4.5 Numerical Example

To illustrate how the above technique of analysis can be applied in a specific case, consider a three node network having the topology of Figure 4.5.1.

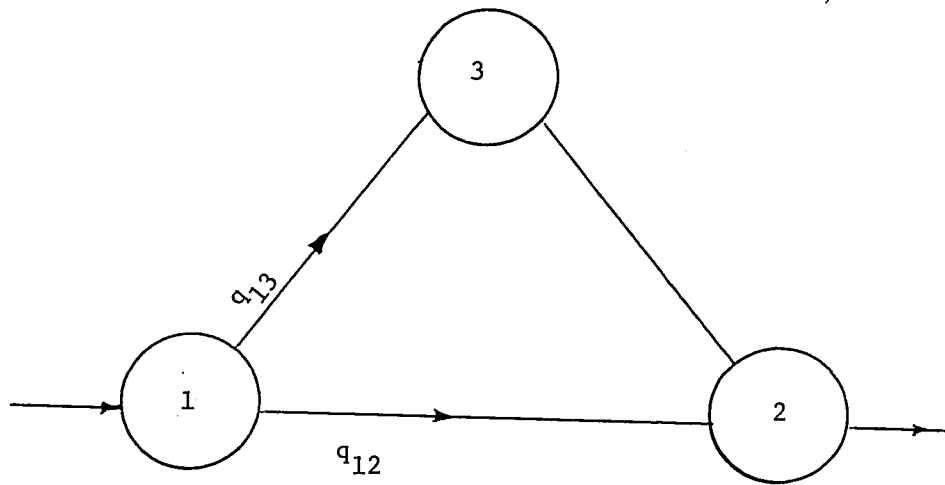


FIGURE 4.5.1: A SIMPLE NETWORK TOPOLOGY.

Packets or messages are arriving at node i , $i=1,2,3$ destined for other nodes. Depending on the destination of these arrivals there could be many source-destination pairs, but for this three-node topology, as the source destination pairs 1-2, 2-3, 1-3 are the same as the source-destination pairs 2-1, 3-2, 3-1 respectively, we can think of three distinct source-destination pairs (1-2, 2-3, 1-3). Let us take nodes 1 and 2 as the source-destination pair for our analysis. Then our logical link will consist of two routes: a) one direct from node 1 to node 2 with probability q_{12} and b) the other from node 1 to node 2 via node 3 with probability q_{13} . In other words our probability matrix A is given by

$$A = \begin{bmatrix} 0 & q_{12} & q_{13} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

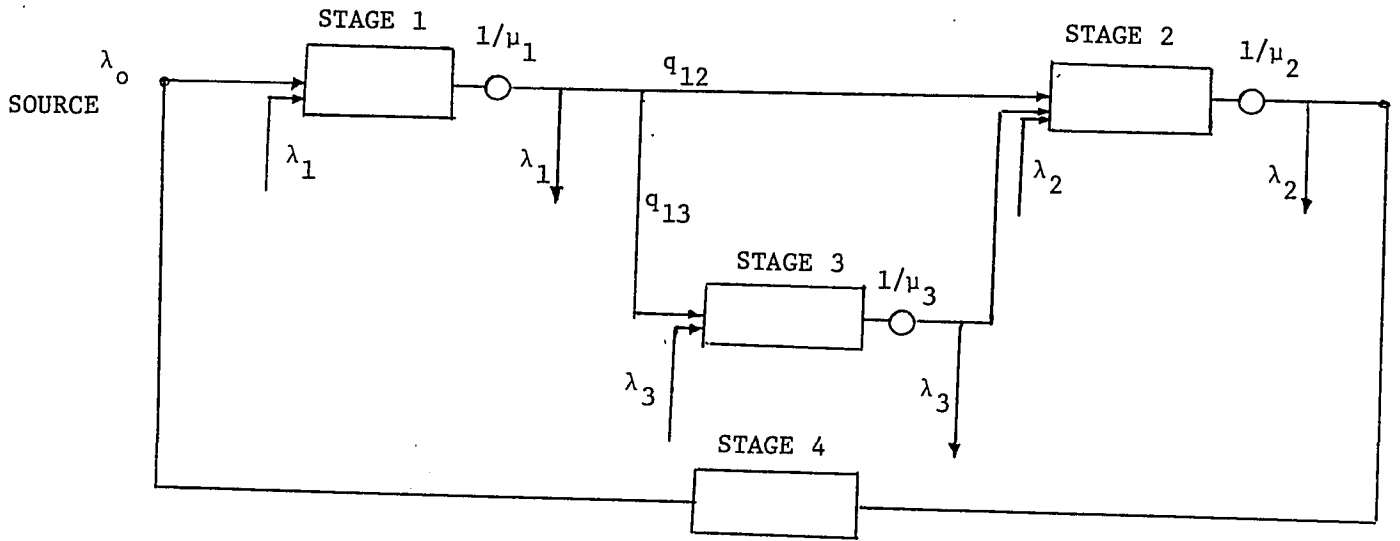


FIGURE 4.5.2: CLOSED QUEUEING SYSTEM WITH ROUTING FOR THE TOPOLOGY OF FIGURE 4.5.1.

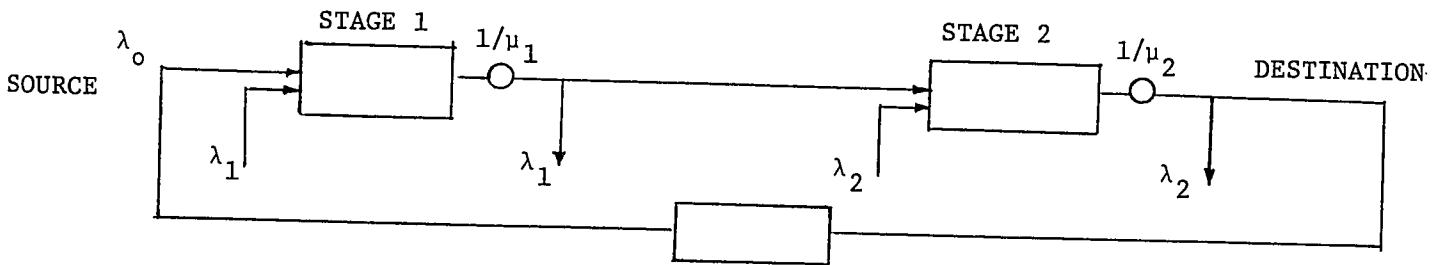


FIGURE 4.5.3: CLOSED QUEUEING SYSTEM FOR 2 STAGE TANDEM LINK FOR THE TOPOLOGY OF FIGURE 4.5.1.

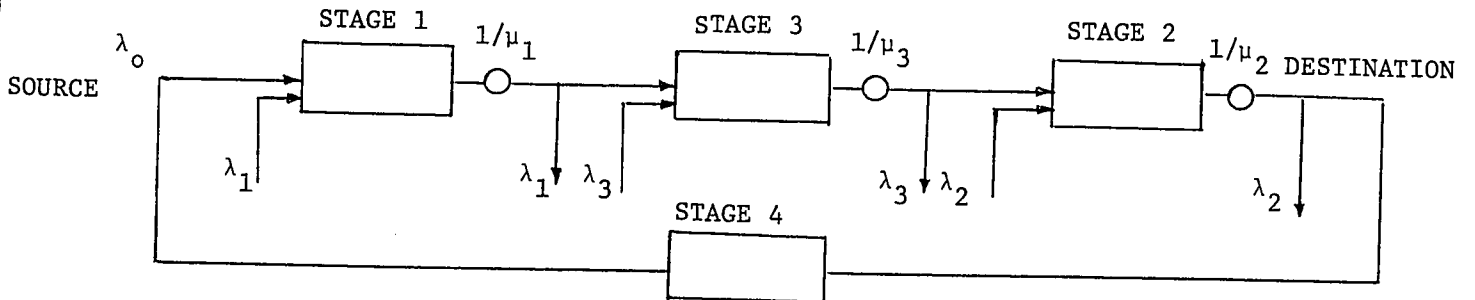


FIGURE 4.5.4: CLOSED QUEUEING SYSTEM FOR 3 STAGE TANDEM LINK FOR THE TOPOLOGY OF FIGURE 4.5.1.

It should be noted that each node processes link customers as well as external customers. External customers are those belonging to other source-destination pairs. As both link and external customers queue up at each node before processing is done, link customers can not be routed before they are processed in the originating node or subsequent nodes. With the above stated reasoning, we obtain the closed queueing network of Figure 4.5.2, for the logical link of source-destination pair 1 and 2. With no routing the closed queueing system for the same source-destination pair would look like Figure 4.5.3 or Figure 4.5.4 depending on whether we consider two stage link or three stage link for message transfer between node 1 and node 2. Note that the available routes between node 1 and node 2 cannot be taken into account in the model considered by Pennotti. Our system of Figure 4.5.2 looks like the combination of Figure 4.5.3 and Figure 4.5.4 with different probability q_{13} and in the limit, when $q_{13} = 1$, it coincides with the system of Figure 4.5.4. This is in fact the case for the system we are considering and it can be verified from the results. For $\rho_0 = \lambda_0 / \mu_1 = 0.3$, $q_{13} = 0.3$, $\lambda_1 / \mu_1 = \lambda_2 / \mu_2 = \lambda_3 / \mu_3 = 0.3$, $N = 3$ and for the two stage configuration of Figure 4.5.3, $E(n_1) = E(n_2) = 0.5332$: whereas for the three stage configuration of Figure 4.5.4, $E(n_1) = E(n_2) = E(n_3) = 0.4672$. For our system of Figure 4.5.2, $E(n_1) = E(n_2) = 0.5170$, $E(n_3) = 0.1231$ for the branching probability $q_{13} = 0.3$. As q_{13} is increased from 0.3 to 0.9, the result of our system of Figure 4.5.2 is $E(n_1) = E(n_2) = 0.4751$, $E(n_3)$

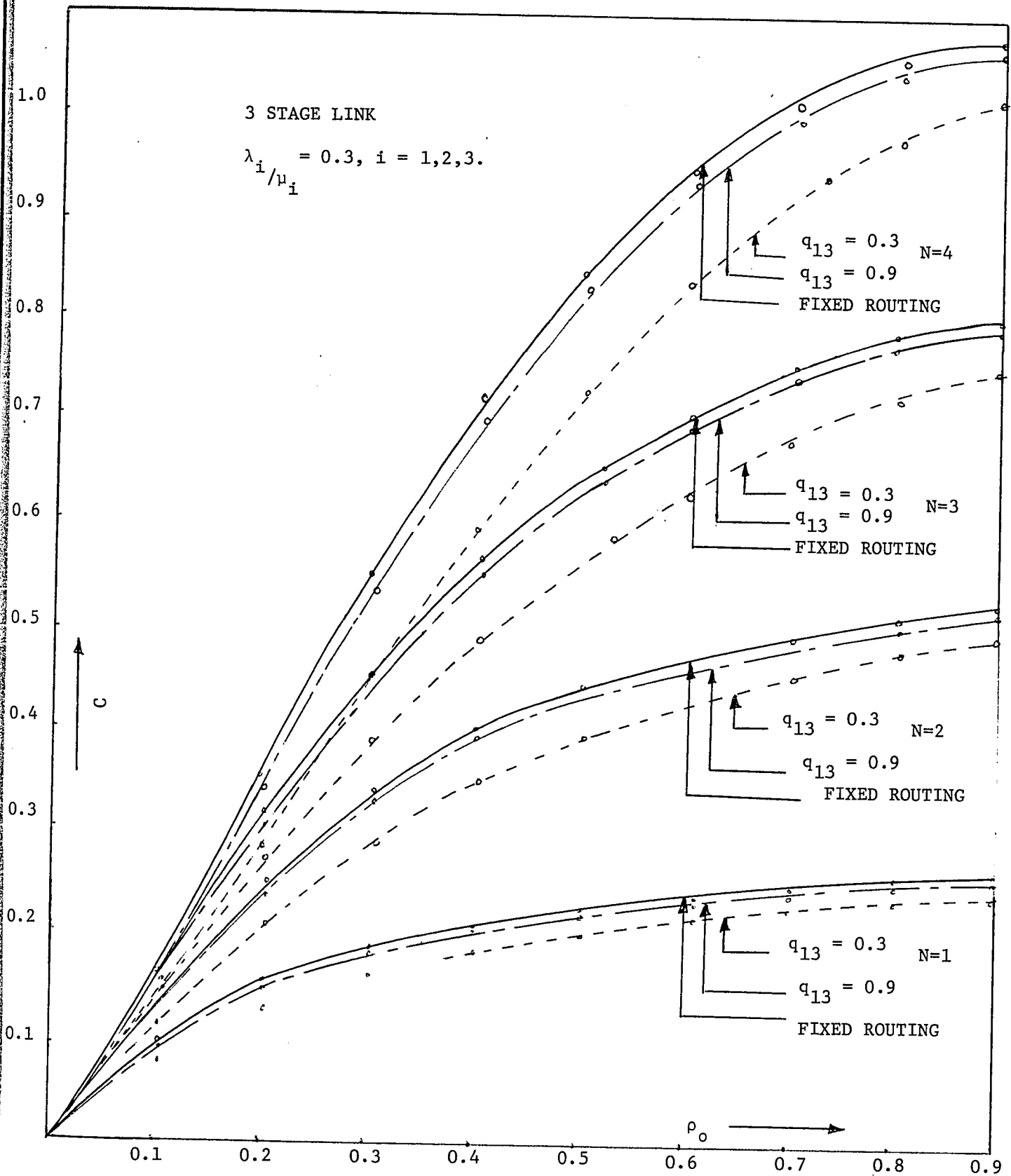


FIGURE 4.5.5: END-TO-END CONTROL PERFORMANCE WITH RANDOM ROUTING AND FIXED ROUTING

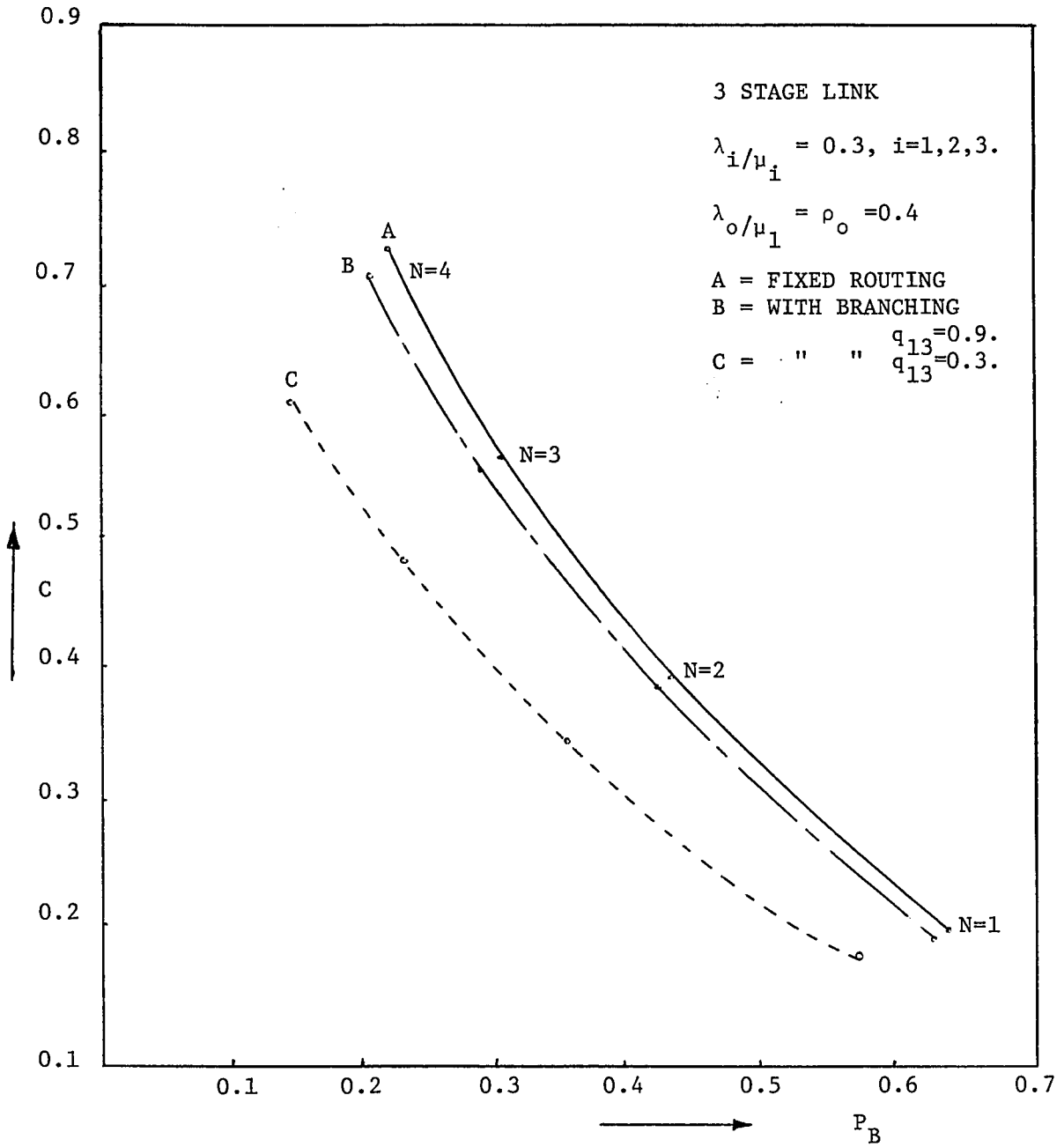


FIGURE 4.5.6: END-TO-END CONTROL CONGESTION vs. BLOCKING PROBABILITY WITH RANDOM ROUTING AND FIXED ROUTING.

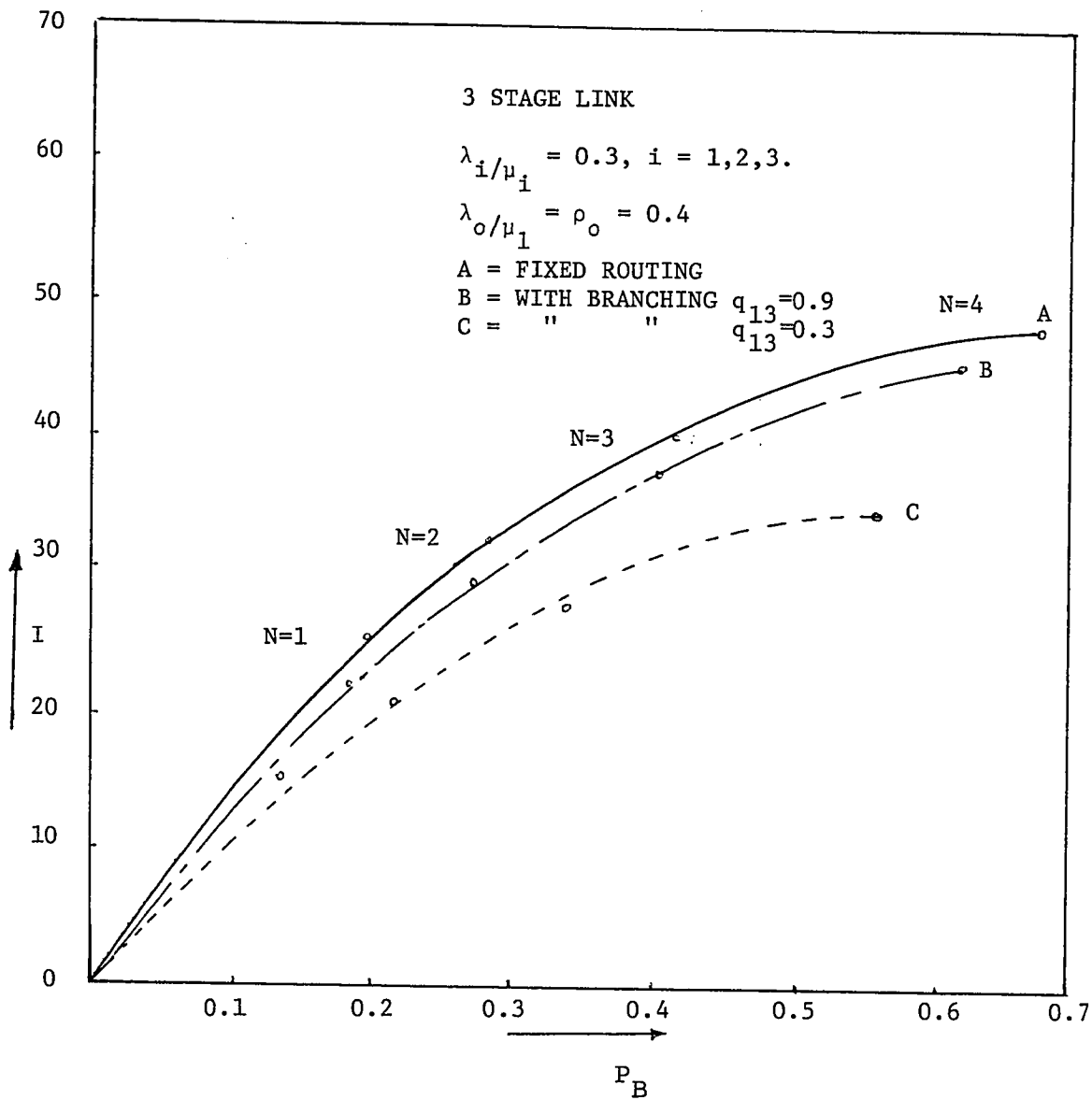


FIGURE 4.5.7: END-TO-END CONTROL: PERCENTAGE IMPROVEMENT vs. BLOCKING PROBABILITY WITH RANDOM ROUTING AND FIXED ROUTING.

= 0.4149 which is close to the one of the 3-stage system of Figure 4.5.4. q_{13} increasing more and more means that all the customers go from node 1 to node 2 via node 3 instead of going directly giving rise to the system considered by Pennotti. This is true for other parameters $E(m_k)$, C , P_B . The relative difference between the two models (i.e. with random routing and fixed routing) is shown in Figure 4.5.5, 4.5.6 and 4.5.7. In Figure 4.5.5 relevant to the example considered, we see that average congestion in our model with different routing is less for the same ρ_0 compared to Pennotti's model for the same control.

Congestion as defined here is related to the queue length at each node. If less and less traffic is routed to a particular node, its queue length will naturally be low compared to the queue length of the other node via which more traffic is routed. More queue length means more buffer storage requirement, higher congestion and overflow probability. So by decreasing average congestion, we decrease the average buffer requirement of the network, though the buffer requirement of a particular node may be higher.

Considering the three stage link of Pennotti [20], we see that the average congestion in our model (also three-stage one) is less and so we can expect more throughput through it. This is evident from Figure 4.5.6. Here, for the same congestion, we see that our model has a low value of P_B , i.e. less traffic is blocked and more throughput obtained. This is not true if we take the Percentage Improvement, I , as the performance

criterion. We see from Figure 4.5.7 that for the same I , the blocking probability P_B of our model is higher compared to the P_B of Pennotti's model. On the other hand, if we take P_B as the performance criterion, we see that I in our model is less for the same P_B , Figure 4.5.7. I is related to the delay encountered in the system; a lower value of I implies a higher average delay for the customer. For the specific case of Figure 4.5.1, we see that more throughput is associated with higher delay. For the example considered, random routing seems to give less buffer requirement at the cost of greater delay. This might not be the case, in general, for any link of arbitrary topology.

CHAPTER - 5

NETWORKS WITH LOCAL CONGESTION CONTROL AND RANDOM ROUTING

5.1 Introduction

In the previous chapter we considered end-to-end congestion control. There we derived a mathematical model of a logical link with end-to-end congestion control and random routing. After defining performance measures, we compared our results with those of fixed routing. In general, end-to-end and local congestion controls fall in the same class of control schemes. End-to-end congestion control places restrictions directly at the message source by monitoring the number of messages on the logical connection between a source node and a destination node - whereas local congestion control places restrictions on the number of messages or packets at each node of the logical connection between a source and a destination. In this chapter we will analyze local congestion control. Analytic results for networks using this type of controls and deterministic routing were given in [20], [37]. In this chapter we will extend the results of Pennotti and Schwartz [37] for networks with local congestion control and probabilistic routing. First, we will analyze the results for the previous chapter, regarding end-to-end congestion control to show why additional local congestion control is essential and then analyze the local control scheme. This is important, because it has been shown by simulation and measurements that neither end-to-end congestion control nor local congestion control in isolation provide a complete solution to the congestion problem. But

end-to-end congestion control in conjunction with local congestion control can potentially solve the congestion problem. For the example to be considered later, probabilistic routing turns out to be better than deterministic routing, in terms of network congestion, for locally controlled networks, as also was the case for the example considered earlier for networks with end-to-end congestion control [38]. It is hoped that inclusion of random routing in our model will enhance its usefulness in providing an analysis indicative of alternating routing performance in the absence of any other such mathematical model.

5.2 Need for Local Congestion Control

In chapter 4, we have analyzed the network with end-to-end congestion control in conjunction with random routing. In this section we will go more in details with the results of end-to-end congestion control in isolation. As the results obtained with fixed and random routing are similar in nature, any intuitive and mathematical argument or reasoning given for one will hold for the other too. For mathematical simplicity we will take the results of section 3.3. For a uniform system, where all λ 's and μ 's are equal and given by λ and μ respectively, we can write:

$E_l = E(n_i)$ = Expected queue length of link customers at each stage.

$E_e = E(m_i)$ = Expected queue length of external customers at each stage.

$T = E_\ell + E_e$ = Sum of the expected queue lengths of the link and external customers at each stage. This can be taken as a measure of the total buffer requirement at any node.

Mathematically, using the expressions in section 3.3,

$$T = \frac{x}{Z_M^N} \cdot \frac{dZ_M^N}{dx} + \frac{\lambda}{\mu - \lambda} \left(1 + \frac{x}{Z_M^N} \cdot \frac{dZ_M^N}{dx} \right) \quad (5.2-1)$$

$$= E_\ell + \frac{\lambda}{\mu - \lambda} (1 + E_\ell) \quad (5.2-2)$$

where

$$x = \frac{\lambda_0}{\mu - \lambda} \quad (5.2-3)$$

Now we would like to see the influence of λ_0 and λ on T .

Differentiating (5.2-2) with respect to λ_0 , we get:

$$\frac{\partial T}{\partial \lambda_0} = \frac{\mu}{(\mu - \lambda)^2} \frac{dE_\ell}{dx} \quad (5.2-4)$$

Further,

$$\frac{\partial T}{\partial \lambda} = \frac{\mu \lambda_0}{(\mu - \lambda)^2 (\mu - \lambda)} \cdot \frac{dE_\ell}{dx} + \frac{\mu (E_{\ell+1})}{(\mu - \lambda)^2} \quad (5.2-5)$$

E_ℓ , the expected queue length of link customers, saturates as x increases, because $\frac{dE_\ell}{dx} = 0$ for large x . At high values of x , $\partial T / \partial \lambda > \partial T / \partial \lambda_0$ which is an indication of higher buffer requirement with respect to external customers compared to the link customers, when there is no local congestion control, assuming there is common buffer space for both. The second term in equation (5.2-5) gives the relation of how this buffer

requirement rises with respect to x . At this phase of link situation local control of external customers as well as link customers is very important at each node of the link for equitable utilization of buffer space. In the next section we will create a model of local control with random routing. Because of the mathematical complexity involved, we will be talking of putting local control on link customer only in that model. Before we do that, we can analyze the results of the chapter 3 further by taking a numerical example to show the relative difference of buffer requirements, as we increase λ_0 or λ , for the model of end-to-end congestion control only.

As $\partial T / \partial \lambda$ is always increasing with x , contrary to $\partial T / \partial \lambda_0$ which is monotonically decreasing to zero, we can define the limiting value of x for which both $\partial T / \partial \lambda$ and $\partial T / \partial \lambda_0$ are equal. Let $\mu / (\mu - \lambda)^2 = K$, then from (5.2-4) and (5.2-5) we get:

$$K \frac{dE_\ell}{dx} \left(\frac{\lambda_0}{\mu - \lambda} - 1 \right) + K (1 + E_\ell) = 0 \quad (5.2-6)$$

where,

$$E_\ell = \sum_{n=1}^N x^n \frac{(n+M-1)!}{(n-1)! M!} \quad (5.2-7)$$

Solution of the equation (5.2-6) will give equal buffer requirement for certain x . Equation (5.2-6) can be further simplified as:

$$\frac{dE_\ell}{dx} (x-1) + 1 + E_\ell = 0 ; \text{ with } E_\ell(0) = 0 \quad (5.2-8)$$

Equation (5.2-8) can be solved for E_ℓ , i.e.

$$(1+E_\ell) (1-x) = 1$$

or,

$$E_\ell = \frac{x}{1-x} \tag{5.2-9}$$

From (5.2-7) and (5.2-9) we obtain an algebraic equation for x . Solving it we then compute T . As an example let us take $N = 3$, $M = 3$. Where $N =$ window size and $M =$ number of stages in the link. With these values equation (5.2-9) becomes:

$$20x^3 - 6x^2 - 3x - 1 = 0$$

There are three solutions for x , the acceptable one being $x = 0.6499$. Table 5.2.1 below shows the relationship of E_ℓ , E_e and T for the same x , but different combination of λ_0 and λ .

Table 5.2.1 Influence of λ and λ_0 on T (Constant x)

x	μ	λ_0	λ	E_ℓ	E_e	T
0.6499	2.0	0.3	1.53	1.853	9.28	11.14
		0.5	1.23		4.56	6.4
		0.7	.923		2.44	4.29
		0.9	.615		1.26	3.12
		1.105	0.3		0.503	2.356
		0.975	0.5		0.951	2.804
		0.845	0.7		1.536	3.389
		0.715	0.9		2.334	4.187

In Table 5.2.1, λ_0 and λ have been varied between 0.3 and 0.9 in the upper and lower halves respectively keeping x constant. It can be seen that, for the same ratio of variation of λ/λ_0 ,

higher values of λ necessitate much higher requirements for buffers. In the example we find that variations in λ are more demanding on the network resources and thus the external customers require a greater control compared to the link customers at the time of congestion. This indicates for the example considered that End-to-End congestion control by itself is not sufficient and so end-to-end congestion control in combination of local congestion control may be more effective for the solution of the congestion problem.

To qualify our statements further, we plot the following curves in figures 5.2.1 and 5.2.2.

- 1) Throughput and delay vs. λ_0 or λ
- 2) Expected queue length vs. λ_0 or λ

Our aim is to maximize throughput through the link under consideration. From curve (a) of figure 5.2.2 we see the requirement of total expected buffer size goes up at a much faster rate with λ than with λ_0 . If we now look on the combined expected delays (including both link as well as external customers), we see that delay also increases at a faster rate with external customers compared to the link customers.

So at the start of congestion, defined by a rectangular block of figures 5.2.1 and 5.2.2, we should start accepting link customers only and blocking external customers, as far as possible, because of rapid consumption of buffer-space by external customers. This can be done first by accepting

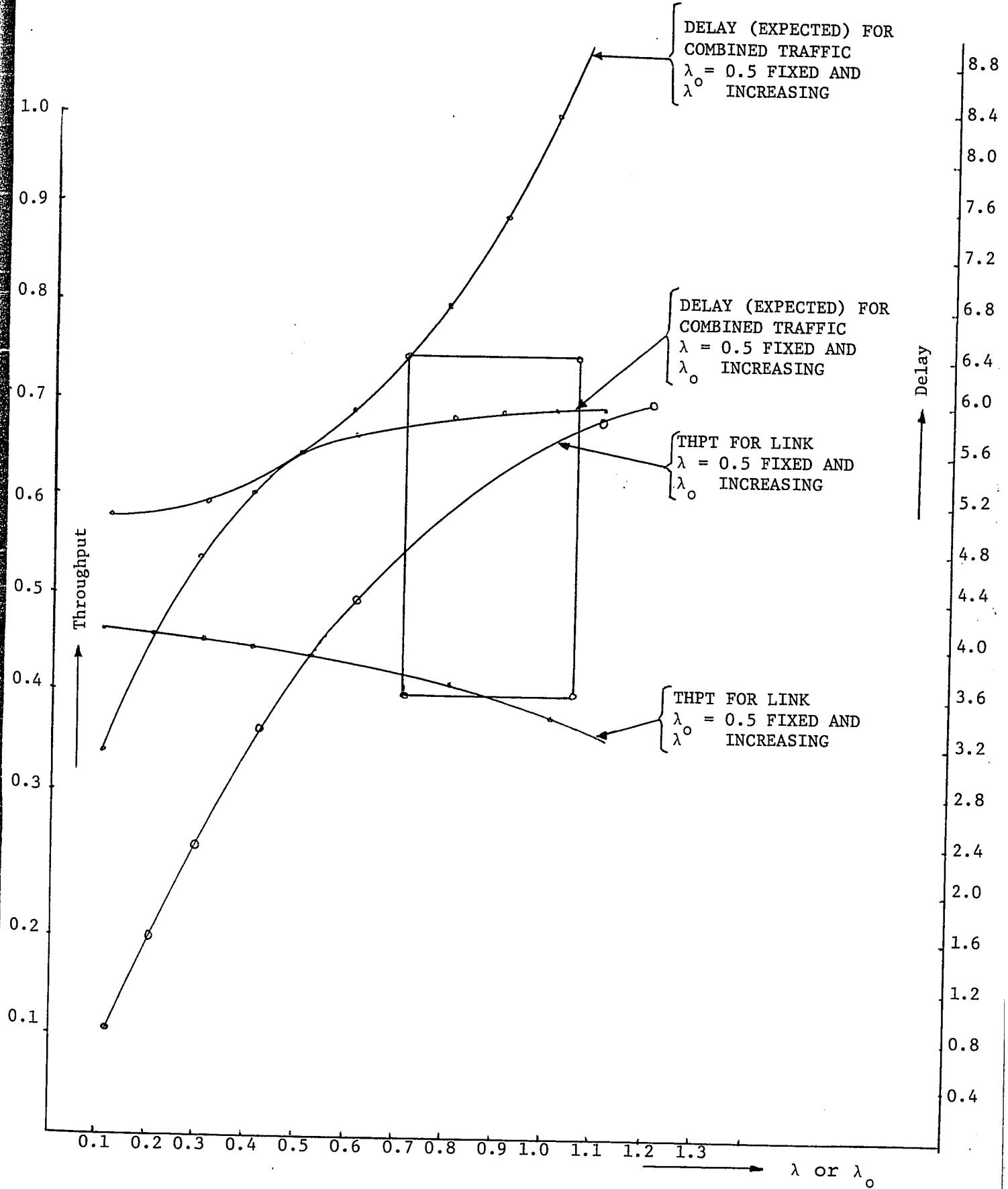


FIGURE 5.2.1: DELAY AND THROUGHPUT vs. λ OR λ_0

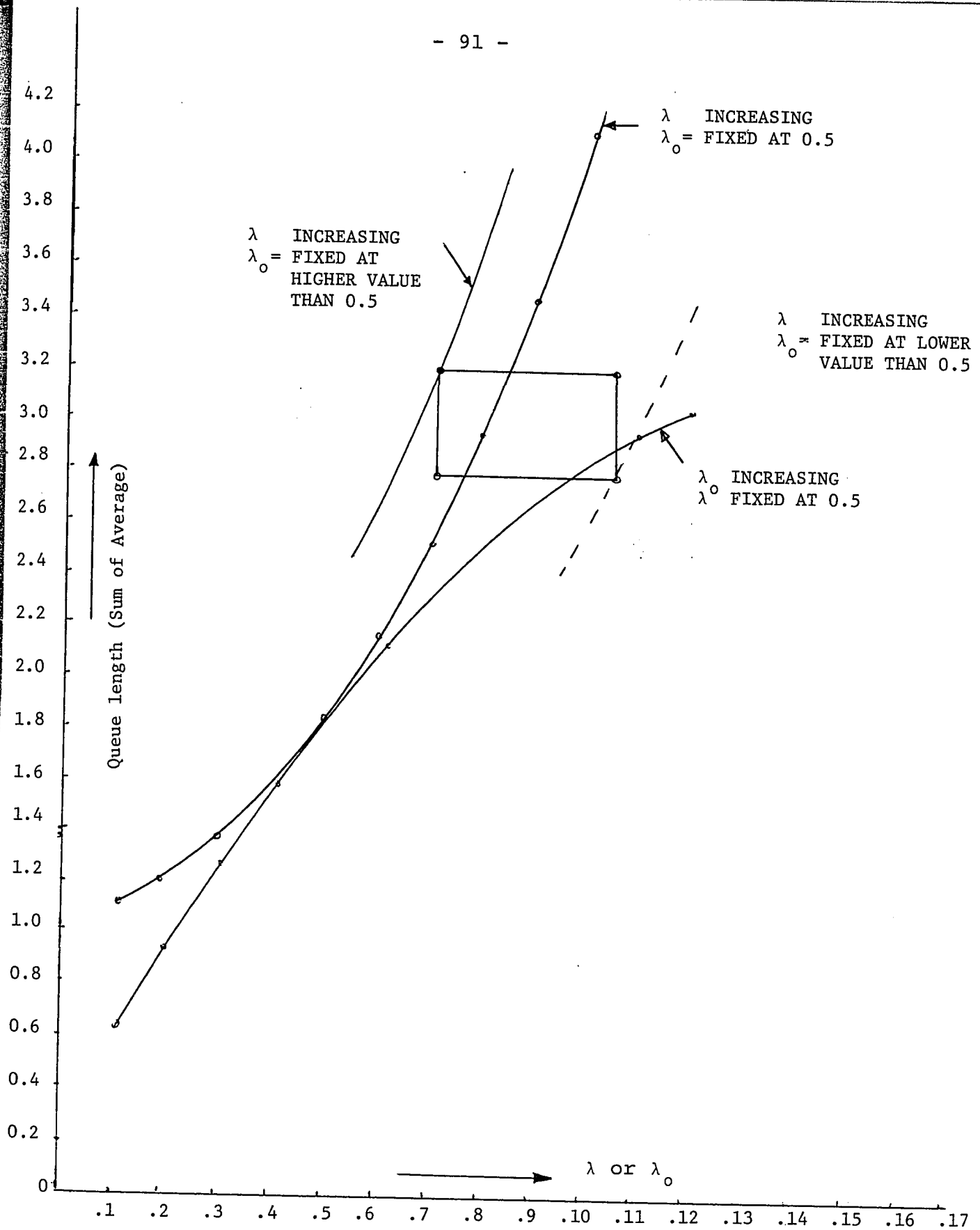


FIGURE 5.2.2: QUEUE LENGTH vs. λ OR λ_0

customers from the node of the link and thus serving those belonging to the link we are concerned with. After a proper time out, if the buffer occupancy does not go down, we should start blocking link customers, i.e., modify the window size, the value of maximum number of outstanding customers N .

Thus we can summarize from all the discussion so far that five techniques can be employed for controlling congestion in the link, after it has been observed in any node. They are

- (1) by accepting customers destined to that node and close to that node
- (2) by blocking λ
- (3) by reducing N , i.e., the window size
- (4) any combination of the above
- (5) by purging packets.

The last technique can be used, if congestion has not been improved by all the other four techniques. A criterion to purge certain types of packets can also be established depending on the durations of such packets in the network.

5.3 The Queueing Model

For each source-destination pair of nodes in the network we define a "logical link" as the unidirectional logical connection between the source and the destination, as defined earlier. In the network considered here a logical link is a subnetwork of M queues interconnected according to the

routing probabilities (q_{ij}) as shown in Figure 4.1.1. The messages generated by the user of this logical link, called link customers, will enter the system at the first queue and join the waiting line there. When their turns for service arrive, they will be served and then will proceed to another stage according to the routing probabilities, and so on until they reach the destination node and finally exit from the system.

In addition to the link customer, messages belonging to other source-destination pairs also join the queues of the logical link considered and obtain service. These messages are called external customers. We make the same assumption as in [37], namely that the arrival process of external customers at each node is independent of the one of the link customers and also independent from node to node. With this assumption, we can treat each logical link of the network separately.

Arrivals of both types of customers are assumed to be Poisson with rates λ_0 , for the link customers, and $\lambda_i = 1, 2, \dots, M$, for external customers at node i of the link. All message lengths are assumed to be taken from the same exponential distribution independently for each node. The average service time at each stage i is assumed to be $1/\mu_i$, $i = 1, 2, \dots, M$.

The local congestion control scheme that we consider here places restrictions on the total number of link customers allowed at each node. Namely, no more than N_i link customers are allowed at node i , $i = 1, 2, \dots, M$. If node i has N_i link customers, all further link customer arrivals to i are blocked and these customers block the service of all nodes connected to node i according to the routing probabilities. Eventually, this blocking could reach all the way back to the source. In the meantime, the i th stage is permitted to serve external customers, even though they may be after the blocked link customers in the queue. We assume, however, that when the blocking condition is removed a blocked link customer will be given priority and that this priority is preemptive resume.

Following Pennotti and Schwartz [37], we consider the following three criteria for evaluating the local congestion control scheme:

- (i) Congestion C
- (ii) Percentage Improvement l and
- (iii) Blocking Probability P_B .

C is the fractional increase in queueing time suffered by external customers due to the presence of link customers, averaged over all external customers in the logical link [20].

$$C = \frac{\sum_{i=1}^M [(\mu_i - \lambda_i) E(m_i) - \lambda_i]}{\sum_{j=1}^M \lambda_j} \quad (5.3-1)$$

where $E(m_i)$ = Expected queue length of external customers at stage i .

I is the % reduction in the waiting time of external customers in the controlled link from what it would be if no control were used:

$$I = \frac{T_i | \text{no control} - T_i}{T_i | \text{no control}} \times 100\%$$
$$= \frac{\sum_{i=1}^M [\lambda_i - (\mu_i - \lambda_i - \lambda_0) E(m_i)]}{\sum_{j=1}^M \lambda_j} \quad (5.3-2)$$

The blocking probability P_B gives the proportion of link arrivals that are rejected by the system.

5.4 Analysis of the Controlled Link

The queueing model of the controlled link presented above is not amenable to exact analytic treatment, even for the case with no external customers. Exact equations for the case of a tandem link and no external arrivals have been given in [35], [41] and [44], but they have only been solved for very restricted cases. Pennotti and Schwartz [37] presented an approximate analysis for local control of a tandem link and their simulations showed good agreement with the results of this approximate analysis. We are going to extend this approximate analysis for the local control of a logical link with probabilistic routing. As in [37] we eliminate the external arrivals from the model of local congestion control by reducing each service rate by the corresponding external

arrival rate:

$$\mu_{e_i} = \mu_i - \lambda_i, \quad i = 1, 2, \dots, M \quad (5.4-1)$$

This approximation, found in [20], is quite acceptable. Then, the resulting system of M finite queues is solved by treating each queue separately as a single, finite M/M/1 queue with an independent probability of the server being not blocked, equal to the sum of the probabilities that the succeeding stages are not full multiplied by the corresponding routing probabilities, q_{ik} .

Assuming that there exists a statistical steady-state we can write the state probability equations for each of the above mentioned finite M/M/1 queues. Figure 5.4.1 gives the model of the i th queue ($i \neq M$), where λ_{e_i} is the, yet to be specified, arrival rate.

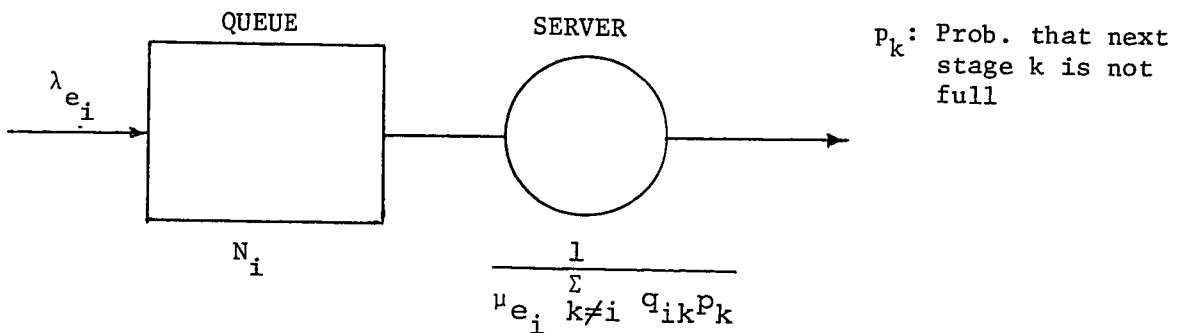


FIGURE 5.4.1: M/M/1 MODEL FOR STAGE i WITH LOCAL CONTROL ($i \neq M$)

The model of the M th and last stage of the logical link is the same with the only difference that the server is never blocked, i.e., the mean service time of the M th stage is $\frac{1}{\mu_{e_i}}$.

The solution of the steady-state equations for M/M/1 queues is well known [34]. The state probabilities $P_i(j)$ of having j link customers in the i th stage given by:

$$P_i(j) = \frac{[\lambda_{e_i} / (\mu_{e_i} \sum_{k \neq i} q_{ik} P_k)]^j}{\sum_{\ell=0}^{N_i} [\lambda_{e_i} / (\mu_{e_i} \sum_{k \neq i} q_{ik} P_k)]^\ell} \quad ; \text{ for } i \neq M \quad (5.4-2)$$

$$j = 0, 1, 2, \dots, N_i$$

$$P_i(j) = \frac{[\lambda_{e_i} / \mu_{e_i}]^j}{\sum_{\ell=0}^{N_i} [\lambda_{e_i} / \mu_{e_i}]^\ell} \quad ; \text{ for } i = M \quad (5.4-3)$$

The probability that the i th queue is not blocked is now:

$$p_i = 1 - P_i(N_i) \quad (5.4-4)$$

Let λ_i^* be the portion of the total steady-state link throughput λ^* that corresponds to node i . We can then write:

$$\lambda_i^* = \lambda_{e_i} P_i \quad (5.4-5)$$

From (5.4-2), (5.4-3), (5.4-4) and (5.4-5) we finally obtain the following system of nonlinear algebraic equations with unknowns p_i and λ_i^* , $i = 1, 2, \dots, M$.

$$P_i = \frac{\sum_{\ell=0}^{N_i - 1} [\lambda_i^* / (\mu_{e_i} P_i \sum_{k \neq i} q_{ik} P_k)]^\ell}{\sum_{\ell=0}^{N_i} [\lambda_i^* / (\mu_{e_i} P_i \sum_{k \neq i} q_{ik} P_k)]^\ell} ; i=1,2,\dots,M-1 \quad (5.4-6)$$

$$P_M = \frac{\sum_{\ell=0}^{N_M - 1} [\lambda^* / (\mu_{e_M} P_M)]^\ell}{\sum_{\ell=0}^{N_M} [\lambda^* / (\mu_{e_M} P_M)]^\ell} \quad (5.4-7)$$

The node throughputs λ_i^* can be found from the link throughput λ^* from the following signal-flow graph, which is identical to the graph of the logical link, by using Mason's rule [45].

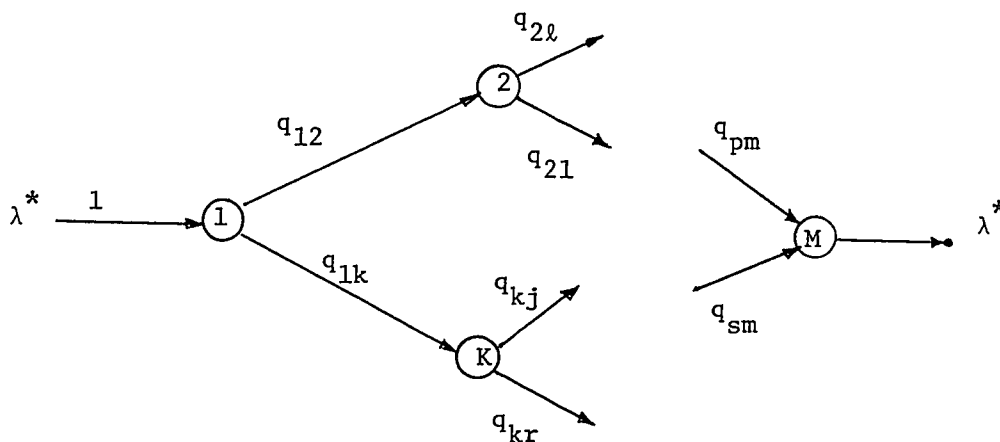


Fig 5.4.2: SIGNAL FLOW GRAPH FOR THROUGHPUT CALCULATIONS

From the flow-graph, using Mason's rule, we obtain:

$$\lambda_i^* = \lambda^* T_{1,i} ; i = 1, 2, \dots, M \quad (5.4-8)$$

where $T_{1,i}$ = Transmittance from source node to node i

The system of equations (5.4-6), (5.4-7), (5.4-8) has now as unknowns λ^* and p_i , $i = 1, 2, \dots, M$. An additional equation is provided by the given link customers arrival rate λ_0 :

$$\lambda^* = \lambda_0 p_1 \quad (5.4-9)$$

Solving those equations we obtain the probabilities p_i , $i = 1, 2, \dots, M$ and from them the average queue lengths $E(n_i)$ at each node and the system blocking probability:

$$P_B = 1 - p_1 \quad (5.4-10)$$

The above results extend the ones obtained in [37] for tandem queues.

An iterative numerical procedure for the evaluation of the throughput λ^* which was introduced in [37] can still be used in our case, but evidently we are dealing with a much more complicated system of nonlinear algebraic equations.

Having the average queue lengths $E(n_i)$, of link customers, we can calculate the average queue lengths $E(m_i)$ of external customers by the following extension of an approximate expression derived in [37].

$$E(m_i) = \frac{\lambda_i}{\mu_i - \lambda_i} [1 + E(n_i) \sum_k q_{ik} p_k] \quad (5.4-11)$$

Then the system measures of performance, C and I and P_B are obtained from (5.3-1) and (5.3-2).

We demonstrate the results by an example. We also make comparisons with the case of deterministic routing.

5.5 Numerical Example

We consider the three node network of Fig. 5.5.1. Node 1 is the source and node 2 the destination of link customers of the logical link (1,2).

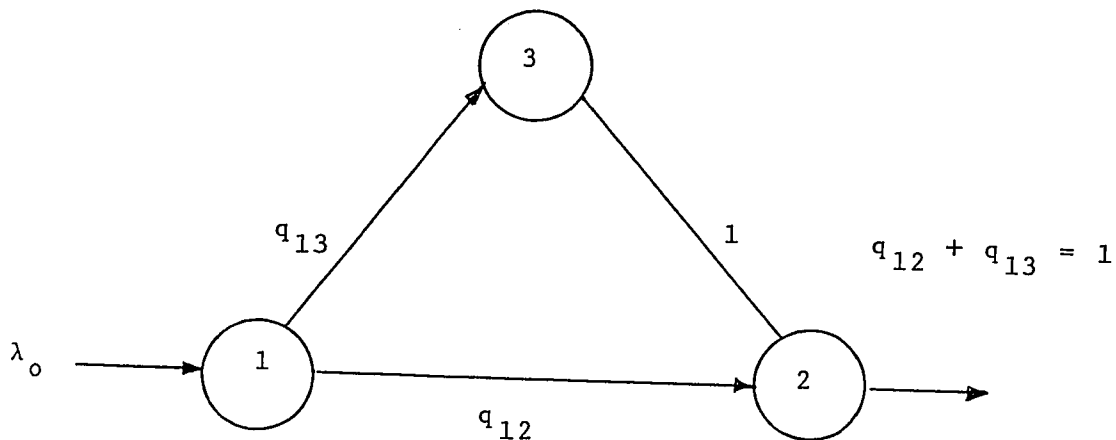


FIGURE 5.5.1: A SIMPLE NETWORK TOPOLOGY

Let λ_0 be the arrival rate of link customers and $\lambda_1 = \lambda_2 = \lambda_3 = 0.3$ be the arrival rates of external customers at the nodes 1, 2, and 3 respectively. Let also $\mu_1 = \mu_2 = \mu_3 = 1$ be the corresponding service rates. We assume that all nodes have the same storage for link customers, i.e., $N_1 = N_2 = N_3 = N$. The loading factor of the link is ρ_0 / μ_1 . The case $q_{13} = 1, q_{12} = 0$ reduces the above

logical link to the tandem one considered by Pennotti and Schwartz [37] and we can make some comparisons.

Equation (5.4-8), becomes here:

$$\lambda_1 = \lambda^*, \quad \lambda_3^* = \lambda^* q_{13}, \quad \lambda_2^* = \lambda^* \quad (5.5-1)$$

and equations (5.4-6) and (5.4-8), give:

$$p_1 = \frac{\sum_{\ell=0}^{N-1} [\lambda^* / (0.7 p_1 (q_{13} p_3 + q_{12} p_2))]^\ell}{\sum_{\ell=0}^{N-1} [\lambda^* / (0.7 p_1 (q_{13} p_3 + q_{12} p_2))]^\ell} \quad (5.5-2)$$

$$p_3 = \frac{\sum_{\ell=0}^{N-1} [\lambda^* q_{13} / (0.7 p_3 p_2)]^\ell}{\sum_{\ell=0}^{N-1} [\lambda^* q_{13} / (0.7 p_3 p_2)]^\ell} \quad (5.5-3)$$

$$p_2 = \frac{\sum_{\ell=0}^{N-1} [\lambda^* / (0.7 p_2)]^\ell}{\sum_{\ell=0}^{N-1} [\lambda^* / (0.7 p_2)]^\ell} \quad (5.5-4)$$

For given λ^* , N and routing probabilities q_{ij} , we first solved (5.5-4), then (5.5-3), and finally (5.5-2). Since λ^* is not known we find it by the same iterative procedure used in [37] and the use of (5.4-9)

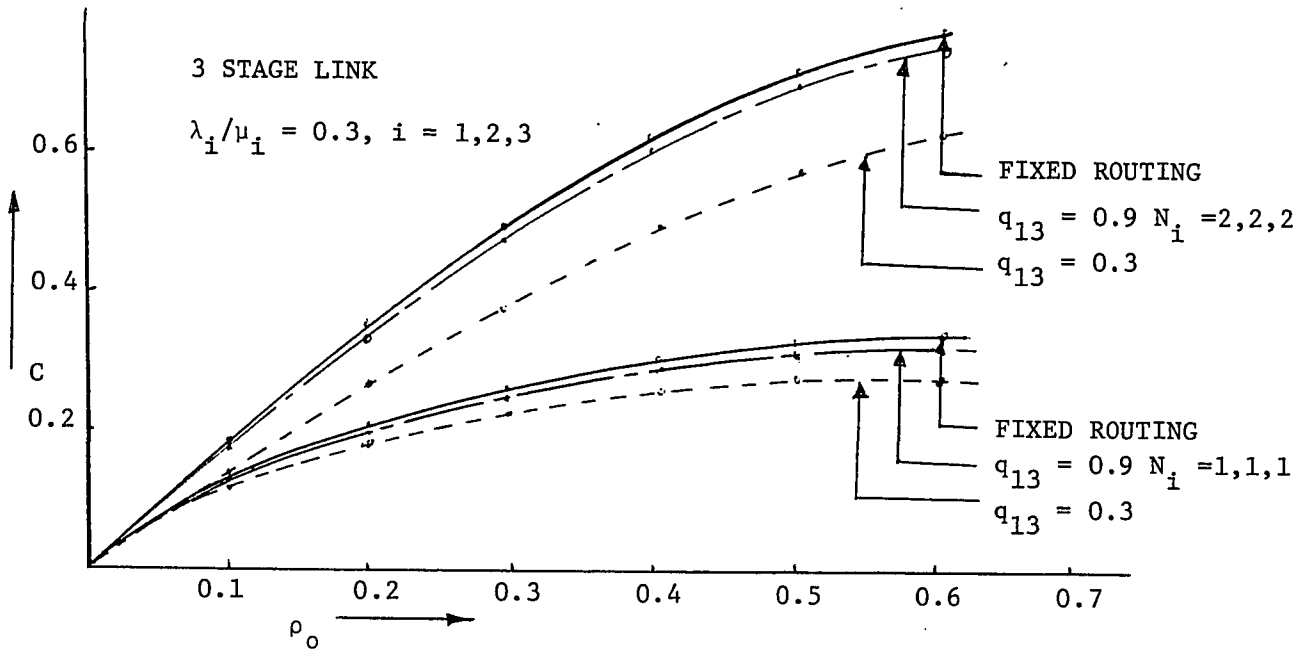


FIGURE 5.5.2: CONGESTION PERFORMANCE WITH PROBABILISTIC AND DETERMINISTIC ROUTING.

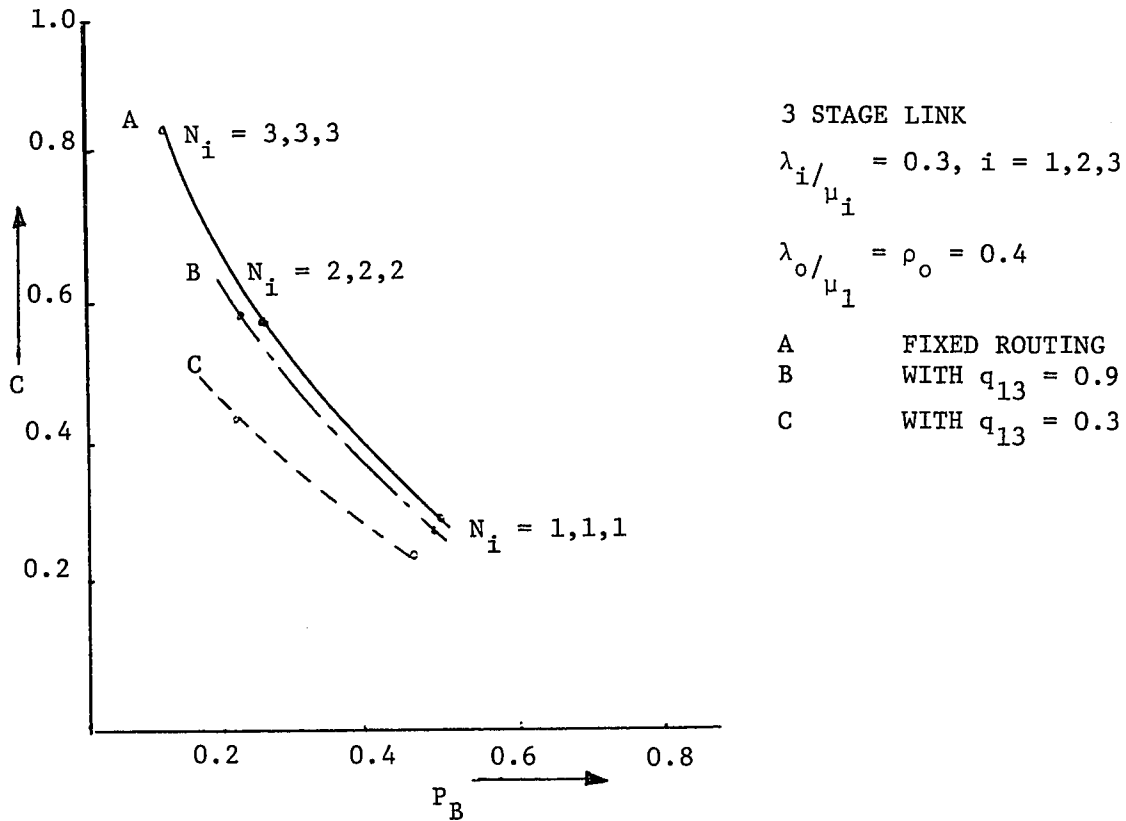


FIGURE 5.5.3: CONGESTION vs. BLOCKING PROBABILITY WITH PROBABILISTIC AND DETERMINISTIC ROUTING.

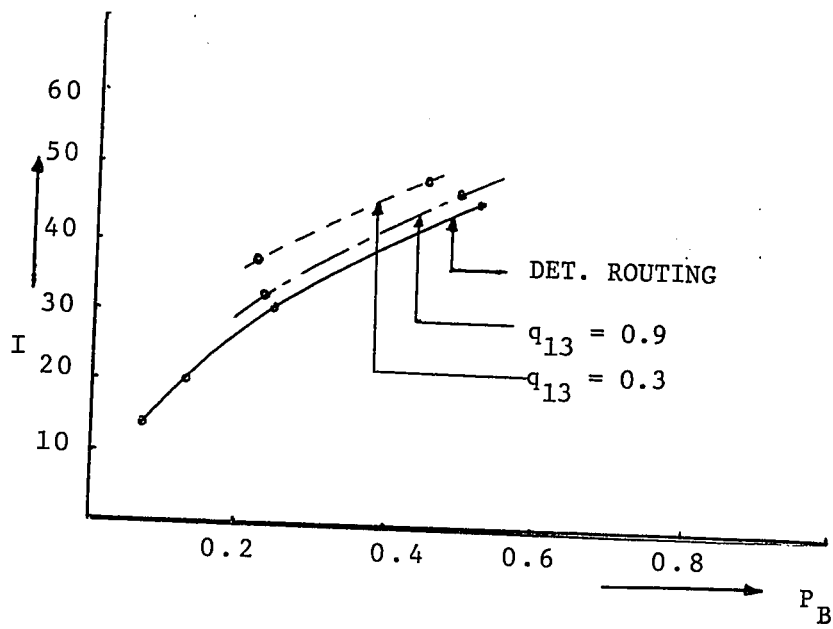


FIGURE 5.5.4: IMPROVEMENT vs. BLOCKING PROBABILITY WITH PROBABILISTIC AND DETERMINISTIC ROUTING

We then computed the basic system measures P_B , C and I . Figure 5.5.2 shows the effect of loading ρ_0 on the congestion C with local control varied from (1,1,1) to (2,2,2). Fig. 5.5.3 shows congestion C versus blocking probability P_B . We observe that with $q_{13}=1$ the results reduce to the one obtained by Pennotti and Schwartz [37]. For the example considered, we also observe that random routing ($q_{13}<1$) reduced the congestion C as compared to the case of deterministic routing of [37]. Figure 5.5.4 shows I vs. P_B . We observe that here I improves with routing.

We did not perform any simulations to verify our approximate results. We mostly based our results on the justification given by Pennotti in [20] by comparing his results with those of ours.

CHAPTER - 6

SUMMARY AND CONCLUSIONS

Flow and congestion controls are two very important factors for proper functioning of the communication networks. In this thesis we first gave a unified definition of flow and congestion control in computer communication networks. In any such network, where there are resources to be shared, flow control must be implemented. Institution of flow control is not related to the level of demand imposed upon the network. If there is a likelihood of the instantaneous demand on the network exceeding its resources, proper congestion control techniques must be implemented to ensure maximum permissible throughput. In a telephone network or in general in any circuit switched environment, it is somewhat easier to implement congestion control techniques compared to that in a store and forward switching environment.

The influence of routing on the throughput was illustrated. It was also shown that optimal routing is a function of traffic level. An example of routing was quoted from the circuit switched telephone network to illustrate this point of view.

As stated in the introduction, analysis is given for two cases; end-to-end congestion control with random routing and the local congestion control with random routing. The results obtained from these specific models were compared with published results, using specific examples. For the examples chosen, the superiority of random

routing compared to the fixed routing from the network congestion point of view was illustrated, under normal load condition.

Further study is required to analyze a generalized network incorporating congestion controls at different levels, possibly with isarithmic, end-to-end and local congestion controls within the same model. In the absence of any other results, the results obtained in this thesis will prove useful in the design of a network, from the congestion point of view.

In the literature, the analytic study of congestion with any routing scheme other than fixed is not available. The generalization introduced in this thesis by formally incorporating a random routing scheme is expected to yield valuable insight in assessing the likely performance of networks with adaptive routing schemes.

With the results of our work, one can ask the simple question whether institution of both end-to-end and local congestion controls in conjunction with a routing scheme will be a complete and infallible solution to the congestion problem. The answer to this question is, obviously, no. The unpredictability of link and node failures and also the likely wide variations of demands on the network resources are the key factors for a negative answer.

So, further extensions of the work reported in this thesis appear to be in order. One potential area is the determination of an algorithm to dynamically change the control parameters to suit

the current network status (as typified by the failed nodes or links) and meet the congestion control requirements.

Further, there is sufficient scope for additional work in modelling the protocols used to effect the different kinds of congestion controls.

In summary, we can say that the whole area of modelling of packet-switched networks with congestion control and protocols is relatively new and much work is yet to be done to come up with models that are mathematically tractable and are also representative of a real network environment.

REFERENCES

- [1] Tobias, M.J., and Booth, G.M., "The Future of Remote Information Processing Systems", Proc. AFIPS Spring Joint Computer Conference, vol. 41, Part II, AFIPS Press; Anaheim, California, 1972.

- [2] Roberts, L.G., and Wessler, B.D., "Computer Networks Development to Achieve Resource Sharing", Proc. AFIPS Spring Joint Computer Conference, vol. 36, AFIPS Press, Montvale, N.J., 1970.

- [3] Kahn, R.E., and Crowther, W.R., "Flow Control in Resource Sharing Computer Network", IEEE Transaction on Communications, Com 20, no. 3, June, 1972.

- [4] Kimbleton, S.R., and Schneider, G.M., "Computer Communication Networks: Approaches, Objectives and Performance Considerations", Communication, ACM, vol. 19, no. 1, Jan., 1976.

- [5] Schwartz, M., Boorstyn, R.R. and Pickholtz, R.L., "Terminal Oriented Computer Communications Networks", Proceedings of IEEE, Special Issue on Computer Design, June 1, 1972.

- [6] Kleinrock, L., "Queueing Systems, vol. II: Application", John Wiley and Sons, New York, 1976.

- [7] Flato, L., "Know Your Common Carriers", Computer Decisions Dec. 1976.

- [8] Millar, J.Z., "DCS AUTODIN Trunking Transmission Workshop on Networks of Computers", NDC-68: 221-224, National Security Agency, Fort Meade, MD, Oct., 1968.

- [9] Paotelli, L.M., "AUTODIN, Computer Communication Networks", Nordhoff International Publ. Leydon, The Netherlands, 1975.

- [10] Kleinrock, L. "Communication Nets", Dover Publications Inc., New York, 1964.

- [11] McQuillan, J., "Throughput in the ARPA Network - Analysis and Measurement", Report no. 2491, Bolt, Beranek and Newman Inc., Cambridge, Mass., January, 1973.

- [12] Miyaraha, H., Hassejawa, T. and Techigauara, Y., "A comparative Evaluation of Switching Methods in Computer Communication Networks", ICC, San Francisco, June, 1975.

- [13] Wood, D., "Economics of Packet Switching Networks", 2nd USA - Japan Computer Conference Proceedings, Tokyo, Japan, August, 1975.

- [14] Clipsham, W.W., Glave, F.E., and Narraway, M.L., "Datapac Network Overview", International Conference on Comp. Communications, Toronto, Canada, 1976.

- [15] Fultz, G.L., and Kleinrock, L., "Adaptive Routing Techniques for Store and Forward Computer Communication Networks", July, 1972, UCLA - ENG - 7252 (ARPA).

- [16] Gerla, M., "Deterministic and Adaptive Routing Policies in Packet Switching Networks", based on UCLA Report no. UCLA - END - 7319, January, 1976.

- [17] Rudin, H., "On Routing and Delta Routing: A comparison of Techniques for Packet Switched Networks", In Proc. 1975 Int. Conference on Communications, paper 416, June, 1975, San-Francisco.

- [18] Rudin, H., "A Performance Comparison of Routing Technique for Packet Switched Networks", IEEE Transactions on communications, vol. COM 24, January, 1976.

- [19] Opderbeck, H., and Kleinrock, L., "The Influence of Control Procedures on the Performance of Packet-Switched Networks", NTC, December, 1974.

- [20] Pennotti, M.C., "The Control of Congestion in Message Switched Networks", Ph. D. Dissertation, June, 1974, Polytechnical Institute of New York.

- [21] Davet, A., Despres, R., Le Rest, A., and Pichon, G., "The French Public Packet-Switching Network", International Conference on Computer Communications, Toronto, 1976.

- [22] Burke, P.J., "Automatic Overload Controls in a Circuit Switched Communication Network", Proceedings of the National Electronics Conference, vol. XXIV, 1968.

- [23] Rudin, H., "Chairman's Remarks, Session 16", ICC '76, Toronto.

- [24] Lande, J.L., "Local Network Management Overload Administration of Metropolitan Switched Networks", Proceedings of the National Electronics Conference, vol. XXIV, 1968.

- [25] Frank, R.L., and Rishel, R.W., "Optimum Network Call Carrying Capacity", The Bell System Technical Journal, vol. 52, no. 7, 1973.

- [26] Weber, J.H., "A Simulation Study of Routing and Control in Communication Networks", The Bell System Technical Journal vol. XLIII, no. 6, 1964.

- [27] Davies, D.W., "The Control in Packet Switching Networks", IEEE Transaction on Communications, COM 20, no. 3, June, 1971.
- [28] Price, W.L., "Simulation of Packet Switching Networks Controlled on Isarithmic Principles", Data Network Analysis and Design, 3rd Data Communication Symposium, Nov., 1973.
- [29] Gerla, M., and Chow, W., "Flow Control in Packet Switched Computer Networks", NTC, December, 1974.
- [30] Sencer, M.A., "Evaluation of Packet Switching Networks Controlled on Isarithmic Principles", NTC, San Diego, 1976.
- [31] Cerf, V.G., and Kahn, R., "A Protocol for Packet Network Intercommunications", IEEE Transactions on Communications, May, 1976.
- [32] McQuillan, J.M., Crowther, W.R., Cosell, B.P., Wlader, D.C., and Heart, F.E., "Improvements in the Design and Performance of the ARPA Network", Fall Joint Computer Conference '72.
- [33] Little, J.D.C., "A Proof of the Queueing Formula, $L = \lambda W$ ", Operation Research, 1961.

- [34] Kleinrock, L., "Queueing Systems, vol. 1: Theory" John Wiley and Sons, New York, 1974.
- [35] Gordon, P.J., and Newell, G.F., "Closed Queueing Systems with Exponential Servers", Operations Research, April, 1967.
- [36] Koenigsberg, E., "Cyclic Queues", Operational Research Quarterly, 1958.
- [37] Pennotti, M., and Schwartz, M., "Congestion Control in Store and Forward Tandem Links", IEEE Trans. on Comm., vol. Com-23, no. 12, 1975.
- [38] Chatterjee, A., Georganas, N.D., and Verma, P.K., "Analysis of a Packet Switched Network with End-to-End Congestion Control and Random Routing", International Conference on Computer Communications, Toronto, Canada, 1976. Also IEEE Transactions on Communications (concise paper) to appear.
- [39] Baskett, F., and Muntz, R.R., "Queueing Network Models with Different classes of Customers", COMPCON '73 and also Technical Report no. 33, August, 1972, Stanford University, Stanford, California.
- [40] Jackson, J.R., "Networks of Waiting Lines", Operations Research vol. 5, 1957.

- [41] Finch, P.D., "Cyclic Queues with Feedback", Journal of Royal St. Soc., Series B-21.
- [42] Muntz, R., and Wong, J., "Efficient Computational Procedures for Closed Queueing Network Models", Proceedings of the 7th HICSS, January, 1974.
- [43] Buzen, J. "Computational Algorithm for Closed Queueing Networks with Exponential Servers", Communications of the ACM, vol. 16, 1973.
- [44] Hunt, G.C., "Sequential Arrays of Waiting lines", Operation Research, December, 1956.
- [45] Mason, S.J., "Feedback Theory - Further Properties of Signal Flow Graphs", Proc. IRE, vol. 44, 1956.
- [46] Baskett, F., Chandy, K.M., Muntz, R.R. and Palacios, F.G., "Open, Closed and Mixed Networks of Queues with Different Classes of Customers", Journal of A.C.M., vol. 22, 1975.
- [47] Tymes, L.R., "TYMNET - A Terminal oriented Communication Network", Spring Joint Computer Conference, 1971.

APPENDICES

APPENDIX A

CALCULATION OF Z_M^N FOR THE CASE OF RANDOM ROUTING

The normalizing condition is:

$$\sum_{\text{all possible states } (\bar{n}, \bar{m})} P(\bar{n}, \bar{m}) = 1 \quad (A1)$$

or

$$P(\bar{0}, \bar{0}) \sum_{\text{all possible states } (\bar{n}, \bar{m})} \prod_{i=1}^M \left(\frac{y_i \lambda_i}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!} = 1 \quad (A2)$$

or

$$P(\bar{0}, \bar{0}) \sum_{\text{all partitions of } n \leq N \text{ into } M} \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} \dots \prod_{i=1}^M \left(\frac{y_i \lambda_i}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!} = 1 \quad (A3)$$

\sum is the sum over all states \bar{n} with all partitions of $n \leq N$ into M

$$n = \sum_{i=1}^M n_i \leq N$$

So,

(A3) can be written:

$$P(\bar{0}, \bar{0}) \sum_{\text{all partitions of } n \leq N \text{ in } M} \prod_{i=1}^M \sum_{m_i=0}^{\infty} \left(\frac{y_i \lambda_i}{\mu_i} \right)^{n_i} \left(\frac{\lambda_i}{\mu_i} \right)^{m_i} \frac{(n_i + m_i)!}{n_i! m_i!} \quad (A4)$$

or

$$P(\bar{0}, \bar{0}) \sum_{\text{all partitions of } n \leq N \text{ in } M} \prod_{i=1}^M \left(\frac{y_i \lambda_i}{\mu_i} \right)^{n_i} \left(\frac{1}{n_i!} \right) \sum_{m_i=0}^{\infty} \frac{(n_i + m_i)!}{m_i!} \dots \quad (A5)$$

$$\left(\frac{\lambda_i}{\mu_i} \right)^{m_i} = 1$$

The technique of evaluating the sum over m_i in (A5) has been shown in [20] to give:

$$\sum_{m_i=0}^{\infty} \frac{(n_i + m_i)!}{m_i!} \left(\frac{\lambda_i}{\mu_i}\right)^{m_i} = \frac{n_i!}{(1-\rho_i)^{n_i+1}} \quad (\text{A6})$$

where,

$$\rho_i = (\lambda_i/\mu_i) \quad (\text{A7})$$

Putting (A6) into (A5), we get:

$$P(\bar{0}, \bar{0}) \sum_{\substack{\text{all partitions} \\ \text{of } n \leq N \text{ in } M}} \prod_{i=1}^M \left(\frac{y_i \lambda_{0i}}{\mu_i}\right)^{n_i} \frac{1}{(1-\frac{\lambda_i}{\mu_i})^{n_i+1}} = 1 \quad (\text{A8})$$

or,

$$P(\bar{0}, \bar{0}) \sum_{\substack{\text{all partitions} \\ \text{of } n \leq N \text{ in } M}} \prod_{i=1}^M \left(\frac{y_i \lambda_{0i}}{\mu_i - \lambda_i}\right)^{n_i} \frac{1}{(1 - \frac{\lambda_i}{\mu_i})} = 1 \quad (\text{A9})$$

If the sum is defined as Z_M^N , the result is

$$P(\bar{0}, \bar{0}) = \frac{1}{Z_M^N} \prod_{i=1}^M \left(1 - \frac{\lambda_i}{\mu_i}\right) \quad (\text{A10})$$

where,

$$Z_M^N = \sum_{\substack{\text{all partitions} \\ \text{of } n \leq N \text{ in } M}} \prod_{i=1}^M \left(\frac{y_i \lambda_{0i}}{\mu_i - \lambda_i}\right)^{n_i} \quad (\text{A11})$$

With the above expression of Z_M^N , the other parameters can be calculated as in [20].

APPENDIX B

CALCULATIONS OF C, I, AND P_B FOR LOCAL CONGESTION CONTROL

The general expression for C, shown in [20] is:

$$C = \frac{\sum_{i=1}^M [(\mu_i^{-\lambda_i}) E(m_i) - \lambda_i]}{\sum_{i=1}^M \lambda_i} \quad (B1)$$

It has been shown in [20] that the $E(m_i)$, the queue length of external customers for local congestion control with no routing can be approximated as:

$$E(m_i) = \frac{\lambda_i}{\mu_i^{-\lambda_i}} [1 + E(n_i) p_{i+1}] \quad (B2)$$

where,

p_{i+1} = the probability that the (i+1) th stage is not full.

For the case of random routing, p_{i+1} will be defined to account for the random routing probability, q_{ik} , as follows:

$$p_{i+1} = \sum_k q_{ik} p_k \quad (B3)$$

where,

q_{ik} = the probability of being routed from stage i to the next stage k. (B4)

p_k = probability that the next stage is not full. (B5)

From (B3), (B2) can be written as:

$$E(m_i) = \frac{\lambda_i}{\mu_i - \lambda_i} [1 + E(n_i) \sum_k q_{ik} p_k] \quad (B6)$$

Using (B6) and (B1), we get an expression for local congestion, C_L as:

$$C_L = \frac{\sum_{i=1}^M [\lambda_i E(n_i) \sum_k q_{ik} p_k]}{\sum_{i=1}^M \lambda_i} \quad (B7)$$

Similarly I for local congestion control can be approximated as:

$$I_L = \frac{\sum_{i=1}^M \lambda_i [x_i - (1-x_i) E(n_i) \sum_k q_{ik} p_k]}{\sum_{i=1}^M \lambda_i} \quad (B8)$$

where,

$$x_i = \frac{\lambda_o}{\mu_i - \lambda_i} \quad (B9)$$

and finally,

$$P_{B_L} = (1 - p_1) \quad (B10)$$

The relevant equation for C_L , I_L and P_B are given in equations, (B7), (B8) and (B10) above.

VITA

NAME : Amalendu CHATTERJEE

DATE OF BIRTH : December 29, 1944

PLACE OF BIRTH : SANDVIP, BANGLADESH

EDUCATION :

1. High School : S.S. High School
Sandvip, Bangladesh
2. College : Comilla Victoria College
Bangladesh
3. Universities : Bangladesh University of
Engineering and Technology,
Bangladesh.
: University of Ottawa, Canada.

DEGREES : B. Sc. Engg. (Elect) in 1967
from Bangladesh.

M.A. Sc. (Elect.) in 1974 from
University of Ottawa.