SOME PROBLEMS IN QUEUEING THEORY

by

Andrew James Coyle
B.Sc. (Hons.), University of Adelaide, South Australia.

Thesis submitted for the degree
of Doctor of Philosophy
in the
Department of Applied Mathematics
University of Adelaide

awarded 19.2.90

September 1989

CONTENTS

SUMMARY.

This thesis is in two parts.

In PART ONE a method for determining the congestion in telecommunication networks is presented.

Chapter 1 contains a review of some of the various ways that congestions in telecommunications networks have been determined. Problems associated with these methods will be pointed out.

Chapter 2 presents a method to determine the call congestion in a network in which only single link chains are used, some results will be presented.

In Chapter 3 the method discussed in Chapter 2 is extended to the case when multi-link chains are used.

Chapter 4 is the conclusion of this part of the thesis and contains a summary of the work described in this section as well as results obtained using our method.

PART TWO of this thesis presents a study of sensitivity bounds for certain queueing systems. A method to determine certain bounds is given and is used to find bounds on the time congestion in the $GI/M/n/n$ queueing system.

Chapter 5 contains an introduction to the topics of insensitivity and Generalised Semi-Markov Processes. Descriptions of various ways that have been used to determine bounds for some performance measures in some insensitive systems will also be presented.

Chapter 6 describes a method which can be used to determine the bounds on certain performance measures for Generalised Semi-Markov Processes in which

there is only one general lifetime distribution. A few examples of systems which can be analysed using this method will be given and some results presented.

In Chapter 7 the method described in Chapter 6 will be used to determine bounds on the time congestion in the $GI/M/n/n$ queueing system. A proof of this result will be presented for the case when $n = 2$, the full proof of this result is presented in the Appendix. A comparison of the maximum time congestion and the time congestion achieved using some well known arrival distributions will be presented.

Chapter 8 contains some conclusions and points out ways in which further research will extend these studies.

Statement.

The contents of this thesis have not been submitted to any university for the purpose of obtaining any other degree or diploma. Also, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text. I consent to the thesis being made available for photocopying and loan.

Andrew James Coyle.

Acknowledgements.

I would like to thank my three supervisors Les Berry, Peter Taylor and Bill Henderson for all the help and encouragement that they have generously given me whilst working on this thesis. Many thanks also to Diana Pallant for proof reading this thesis and offering many useful comments. The work on the first part of this thesis was supported by Telecom Australia under contract no. 66407. The work on this first part was also supported by an Australian Commonwealth Postgraduate Research Award.

PREFACE.

This thesis is in two parts. Both of these parts are concerned with problems in the field of telecommunications.

The first part presents a method for approximating the end-to-end congestions in a non-hierarchical telecommunications network. This research was undertaken from 1983 to 1985 and was supported in part by a contract from Telecom Australia. The work presented here is an outline of the method and some of the reasoning behind the techniques used. Many of the details of this work have been omitted since they tend to obscure the main points.

The work for the second part of this thesis was undertaken in 1988 and 1989. This part of the thesis is concerned with finding bounds on performance measures in some simple queueing systems with generally distributed inter-arrival or service times. This method can give simple analytic results that could be useful when analysing large networks of queues.

CHAPTER 1

INTRODUCTION TO TELEPHONE NETWORKS.

With the introduction of new digital switching and transmission systems into Australia's telecommunication systems, advantages may be anticipated by replacing some or all of the present hierarchical routeing network with a non-hierarchical routeing system. Among the advantages to be gained by this replacement are the creation of a cheaper, more robust network and the ability to incorporate a certain amount of dynamic routeing by using one set of routeing rules for busy daytime periods and a different set of rules for quieter evening periods. Jessop and McLeod (1983) have discussed the possibility of introducing non-hierarchical networks into the Australian International Digital Network and some of the advantages . New methods for dimensioning these non-hierarchical telecommunication networks must be introduced and one of the possible ways of doing this is to adapt existing models, presently used with hierarchical networks, so that they can be used for the non-hierarchical networks. The adaption of Berry's chain-flow model (1971), in which the network is investigated chain by chain, is the object of the first part of this thesis.

When the exact structure, routeing rules and the traffic offered to a non-hierarchical telecommunication network are known, we shall, by successively looking at each chain, formulate an iterative scheme which will determine the end-to-end congestions for each one of the network's arrival streams.

1.1 Non-Hierarchical Networks.

In the hierarchical networks presently used in Australia (see for example Truitt (1954)), the final choice routes in the network carry a large amount of overflow traffic, which has a high variance to mean ratio, while the direct routes carry traffic with a lower variance to mean ratio. The higher the variance to mean ratio of the traffic offered to a link, the greater the number of circuits per unit of traffic required to carry this traffic. Hence the final choice routes in this network have a lower efficiency than the direct links. However, it is possible that switching and transmission equipment can be introduced into the present system (Schramel (1982)) so that a new type of network is created. More flexible routeing rules can be employed in which every link in a network can carry both direct and final choice traffics. Therefore the total number of circuits required is less than that needed for the present hierarchical network with the same grade of service. The grade of service is specified by a set of lower bounds on the mean proportion of calls that may be lost from any of the origin-destination streams. The introduction of new technology has enabled many different routeing strategies to be employed. The problem of designing routeing strategies to accomodate this flexibility is thus of considerable importance.

The two different types of non-hierarchical networks which may be introduced are dynamic and static networks. A dynamic network is one in which the routeing rules employed in the network are not constant. An example of this is a network in which there are only two sets of routeing rules for the network, one for the busy daytime period and another for the evening period. Another possible system is one in which the routeing rules are constantly changing depending on the present state of the network (see for example Narendra and Mars (1981)). For each of these types of networks there are also many different sets of routeing rules that can be employed. Ash, Cardwell and Murray (1981) have looked at a number of different routeing strategies for a proposed dynamic system. In this system a different routeing pattern is used at different times of the day. They show that there is a possible twenty percent reduction in the cost of the overall network where multi-link path routeing is introduced, in which any call may use any possible multiple link route to get from its origin to its destination. It was discovered, however, that most of the calls only used routes with one or two links per route. Hence a two link path routeing system was proposed since this system was easier to implement and the cost for this system was about the same as the cost for other possible routeing systems.

The second type of network is a static network in which the routeing rules employed do not vary over time. The congestions in a static network may be larger than those of a similar sized dynamic network but the cost of the dynamic network will be greater. In some networks the cost savings due to a dynamic network will not be great enough to justify implementation. This, of course, depends on the network that we are investigating. At this point in time Telecom Australia is interested in investigating non-hierarchical static telecommunications networks. In this thesis we describe a method for determining the congestions in a static non-hierarchical network. This scheme may be applied to the sort of network described by Ash, Cardwell and Murray (1981) by using it twice, once for one set of routeing rules and then for the second set of routeing rules. The congestions at the different times of the day when these different routeing systems are used can be found in this way. However, this scheme can not be used to investigate dynamic networks which have constantly changing routeing patterns.

Telecommunications networks, both of the static and dynamic type, may or may not use crankback, that is a call's signal, having proceeded along a route to some node and finding the remaining paths to its destination all blocked may be "pulled back" to an earlier node on its route to try an alternative path to its destination. Crankback in which the call's signal may be pulled back only one link or in which it may be pulled back a number of links are all possibilities which may be implemented. Note that if a two link path routeing system is employed then only one link crankback is needed since any two link route must be from the origin to the destination. The method outlined in this thesis can be used to investigate a system with or without crankback.

As well as possibly being cheaper, non-hierarchical networks have other advantages over the present hierarchical systems. One of these advantages is the increased robustness of the network. If, in a hierarchical network, one of the links is blocked due to a breakdown, the increase in traffic lost from this network is usually greater than the increase in traffic lost from a comparable non-hierarchical system. This robustness is particularly evident in the networks which have constantly changing routeing rules but it is still evident in static non-hierarchical systems. Cameron (1981) has demonstrated this robustness for a dynamic routeing system using simulation techniques to investigate the performance of hierarchical and dynamic non-hierarchical networks before and after a link breakdown. The greater flexibility offered by non-hierarchical routeing systems is one of the reasons why this greater robustness of the network is possible, since a traffic stream may have a number of possible routes, each having no link in common with any other route to which this traffic stream may be offered.

Once the type of network and the routeing system that is to be employed has been decided upon it is then necessary to investigate the performance of these networks for any circuit allocation vector, that is a vector consisting of the number of circuits on each link in the network. First it is necessary to be able to determine the end-to-end congestions of the network when a certain circuit allocation vector and set of arrival streams are known. Using these results it may then be possible to find a process to optimise the number of circuits in the network. Garbin and

Knepley (1981), for example, have presented an iterative approach for designing a static non-hierarchical network. In this approach a network which is fully connected, that is every node is connected to every other node, has, at each successive iteration, a number of circuits removed from the network in order to lower the cost of the network. In some cases an entire link may be removed from the system. The final network has a much lower cost than the initial fully connected system. The procedure outlined in this thesis may be used with this type of scheme in which circuits are successively removed until the grade of service requirements are just met. In this case a lower cost, although not necessarily a minimum cost, network may be designed.

We shall investigate a method to determine the end-to-end congestions in a static non-hierarchical telecommunication network. It will be shown that it is possible to use an iterative scheme to approximate the end-to-end congestions using a chain-flow approach. A chain is a series of links forming a route between an origin-destination pair. Each origin-destination pair may have a number of possible chains or paths that a call may take. This iterative scheme calculates the traffic lost from each arrival stream offered to the network by calculating the traffic lost from each chain that a call from this arrival stream may take. A number of people have investigated the problem of determining end-to-end congestions in non-hierarchical networks using iterative schemes. These schemes all consist of an iterative procedure in which the approximations to the congestion values on the network converge after a number of iterations to some specific values. How close these values are to the real values for the congestion on the network, usually determined by simulation, is the major factor in assessing how good these schemes are. Also of major importance is the running time for these schemes. If we want to optimise a network using one of these iterative procedures, then this procedure will be called many times during one run of an optimisation program. An overly long running program which gives more accurate results is not necessarily a better method than one which has a shorter running time.

In many procedures assumptions about the network are made which are not necessarily true and these can lead to major errors. For example Lin, Leon and Stewart (1978) have presented a single-moment method for determining end-to-end congestions. Two assumptions made, which are admitted by the authors to be open to skepticism, are "Link blocking probabilities are statistically independent" and "Call arrivals (node originated plus overflows) on any link is a Poisson process". The second assumption means that the model assumes that the sum of the overflow and direct traffics offered to a single link are Poisson. This is obviously not the case. However, in comparing the results for a network's grade of service using this method with those obtained by using a two-moment method, in which the two parameters mean and variance are used to describe the offered traffics to links, both sets of results were close to the results obtained using simulation. This comparison was for an AUTOVON network, part of the worldwide Defense Communication System, in which special routeing strategies were used. For networks in which a large proportion of the traffic is carried on first choice routes, the assumption of Poisson offered traffics to every link may be approximately true. For other networks in which large amounts of traffic are not carried on the first choice routes, two moment methods would be expected to be more accurate. Of course even two moment methods are still approximations and the description of traffic using more than two moments may be more accurate still. However, the difficulties involved would be even greater than those that exist for the two moment method and the benefits are unlikely to be major. Whitt (1984) has investigated how much variation may exist in a queue performance measure when the arrival stream is approximated by using only the stream's first two moments. This work is discussed in more detail in the second part of this thesis.

The assumption of independent link blocking probabilities is an assumption which is made by all other methods known to the author. When a number of independent arrival streams are offered to various routes in a network, the state of the entire network is dependent on each individual arrival stream and every possible route that this stream may take. Hence there is some amount of dependence between any two overflow streams and between any overflow and direct stream in that network. This dependence may range from being insignificant to being fairly important and is one of the major problems which must be investigated when looking at these networks. Very few actual results from the methods that assume independence seem to have been published and the accuracy of some of them is in doubt. This also means it is hard to compare the different methods. Gaudreau (1980) has demonstrated a method in which the probability that a call is carried on a path is the product of the probabilities that this call is carried on each of the links in this path. For example for the serial path in figure 1.1, with blocking probability p_1 on the first link and blocking probability p_2 on the second link, the congestion on the path is calculated to be equal to $B = p_1 + (1 - p_1)p_2$ using the independence of blocking probabilities.

If only a single stream is carried along this path, however, the probabilities of blocking are not independent and in fact $B = p_1 = p_2$, in other words the total blocking on the path is the same as the blocking on each one of these links. This leads to a significant error in the proposed approach for this simple network. For large networks in which overflow streams from any single direct route link are then routed into other multi-link chains, this effect will not be very serious if the amount of overflow traffic is small. Therefore, for certain networks the above

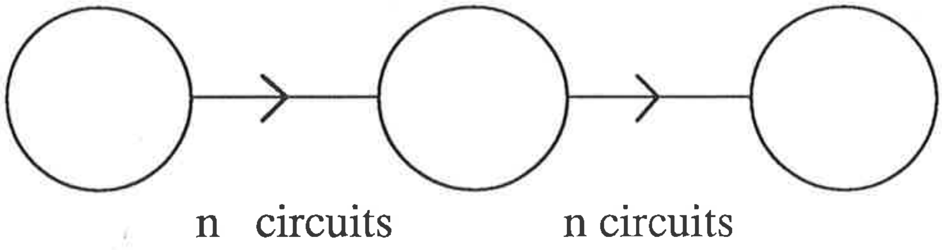

Figure 1.1 *A simple network in which individual link blocking probabilities are not independent.*

assumptions may be approximately true; in general, however, serious difficulties may occur when they are used.

Further examples of methods to determine congestions include a three moment method by Kuczura and Bajaj (1977), a polynomial algorithm for a certain class of networks by Girard and Ouimet (1983) and the Fixed Point Method by Kelly (1986).

1.2 General Assumptions.

The aim of this thesis is to formulate a method for determining the end-to-end congestions in a general network. The assumptions which are made about the network under investigation are:

i) Independent Poisson arrival streams.

ii) Negative exponential service times on all links.

iii) All blocking occurs on the links.

iv) All blocked calls are lost and do not return.

v) Call set up time is negligible.

vi) All traffic is routed under full availability conditions.

vii) All traffic is described by only its first two moments.

viii) The system has reached equilibrium conditions.

As all the traffic is described by only its first two moments, we will use the mean and variance, usually denoted M and V, to describe any traffic stream on the network.

The assumption of independence of overflow and direct streams offered to a single link is not made, however, if it is found that this assumption is valid it may be incorporated into the procedure at a later stage.

1.3 Crankback and Routeing.

There are many different routeing systems that can be employed for non-hierarchical networks. The system that is currently favoured by Australian Telecom (see for example Berry and Coyle (1984)) has been used on simple networks to analyse the method discussed in this thesis for determining the end-to-end congestions in a network. The routeing schemes used have at most two links in any chain of a network, see figure 1.2.

Normally an arrival stream is first offered to the direct single link chain from the stream's origin, node 1, to its destination, node 2. The stream of calls offered to this link but which cannot be carried on this link form the overflow traffic stream. This overflow traffic is now offered to the second choice route: a chain composed of two links, the link from node 1 to node 3 and the link from node

3 to node 2. A spare circuit must exist on both of these links for a call to be carried on this chain. If a spare link does not exist on both of these links then the call overflows from this chain. If a third choice route is used, the overflow from the second choice route is also offered to a third choice two link chain and so on. The traffic overflowing from the final two link chain is lost from the network. The percentage of the calls from this arrival stream that are lost from the network is this arrival stream's end-to-end congestion. Using this scheme no two chains carrying traffic from the same arrival stream have a common link. The problem of the dependence of separate streams of traffic offered to a link is therefore not as significant for this routeing scheme as it is for some other routeing schemes.

Some routeing schemes use crankback. A call that is offered to a multi-link chain initially tries to get on the first link in this chain. If this first link is busy the call cannot get on this chain and is offered to the next choice chain. If the first link of this chain is not busy the call will now try to get onto the second link in this chain. If this second link is blocked the call cannot be carried on this chain. For systems using crankback this call can be offered to the next choice route, we say the call has been "cranked back" to the start of this chain. For systems without crankback this call is lost whether or not another choice route exists. This process is the same for the other links in this chain.

For example, in figure 1.2 a call trying to get from node 1 to node 2 will first be offered to the direct route 1-2. If this is blocked then the route 1-3-2 will be attempted. If link 1-3 has a spare circuit the call's signal will go to node 3 and try to get on link 3-2. Without crankback if this link is blocked then the call is lost, whether or not a path exists on the route 1-4-2. With crankback the call's signal is pulled back to node 1 and the third choice route 1-4-2 can now be tried.

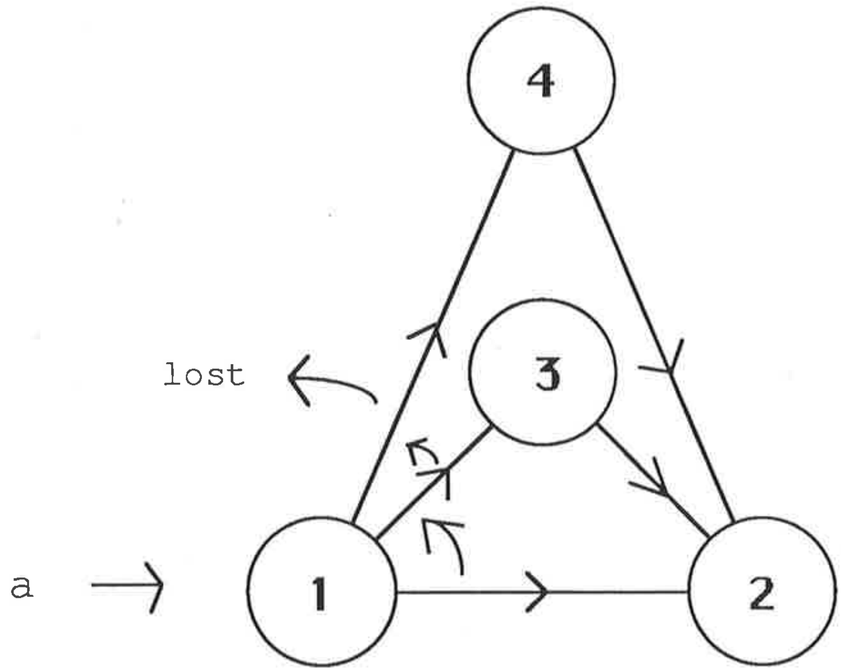

Figure 1.2 *An example of the routeing scheme that is favoured by Australian Telecom.*

We would expect the amount of traffic lost from a network using crankback to be less than that lost from the same network not using crankback. This is shown to be true in most cases when we compare some results from networks for which crankback is implemented with those where no crankback is implemented. For some cases, however, this is not true and using crankback actually increases the congestion on a network. These results are presented in chapter 4.

CHAPTER 2

AN ITERATIVE SCHEME FOR SIMPLE NETWORKS.

A scheme for determining the call congestion for some simple telecommunications networks when only single link chains are used in that network is introduced in this chapter. This method will be extended in the next chapter so that it can also be used on networks in which the chains are composed of more than one link. The first three sections of this chapter introduce some well known preliminary results which shall be used in the iterative scheme described in the fourth section of this chapter.

2.1 Poisson Traffic Offered to a Link.

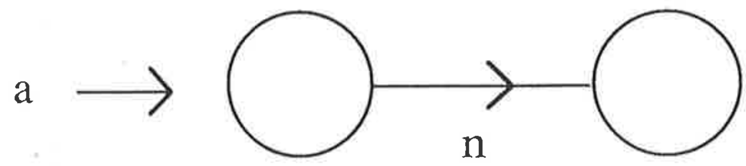

Figure 2.1 *The Erlang loss system.*

One of the simplest and best known queueing systems is the Erlang loss system. This system is represented in figure 2.1. A Poisson arrival stream with intensity a is offered to a single link composed of n circuits. When a call arrives at this link that call will be carried on the link if any of the n circuits are unoccupied. If all of the circuits are occupied when a call is offered to this link the call cannot be carried and is lost from this link. A number of formulae exist, some exact, others approximate, to calculate how much of the traffic offered to this link will be lost, the variance of this traffic, the amount of traffic carried on this link and the variance of this carried traffic. The amount of traffic lost from this link, M, is given by

$$M = aE(n,a)$$

where

$$E(n,a) = \frac{a^n/n!}{\sum_{i=0}^{n} a^i/i!} \tag{2.1}$$

is the well known Erlang loss formula (see for example Brockmeyer, Halström and Jensen (1948)) which gives the probability that all n servers of the link are busy. The variance of the traffic lost from this link, V, can be found exactly by using Riordan's formula given by

$$V = M\left(1 - M + \frac{a}{n+1+M-a}\right), \tag{2.2}$$

(see Wilkinson (1956)). Although it is not obvious from the above, it can be shown that $V > M$, this is a characteristic of traffic overflowing from a link offered Poisson traffic. Traffic for which the variance to mean ratio is greater than one is called *rough* traffic. It is always true that a traffic stream overflowing from a link in a network with Poisson offered traffics will have a variance to mean ratio greater than one.

The traffic carried on this link, M_{ca}, is obviously just given by

$$M_{ca} = a - M = a(1 - E(n,a)) \tag{2.3}$$

and the variance of the traffic carried on this link can be approximated by V_{ca} given by

$$V_{ca} = M_{ca} - (a - M_{ca})(n - M_{ca}) \tag{2.4}$$

(see for example Beneš (1961)). When a Poisson stream is offered to a single link the traffic carried on that link has $V_{ca} < M_{ca}$. Traffic for which the variance to mean ratio is less than one is called *smooth* traffic. It does not follow, however, that any traffic stream carried on a network has a variance to mean ratio less than one, since if rough overflow traffic is offered to a link the traffic carried on this link may have a variance to mean ratio greater than one.

2.2 Wilkinson's Equivalent Random Method

In the above section it has been shown how the lost and carried traffics on a single link can be calculated when a Poisson arrival stream is offered to this link. To determine these same quantities when the offered traffic is no longer Poisson a new technique must be used. Wilkinson's Equivalent Random Method (ERM) can be used in this case (see Wilkinson (1956)). The situation for which the ERM is used is shown in figure 2.2.

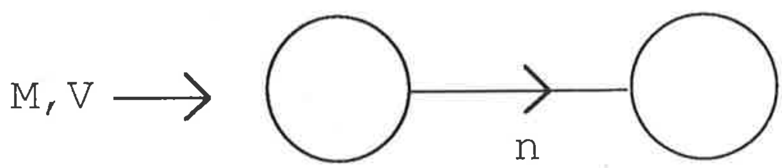

Figure 2.2 *Overflow traffic offered to a link.*

A traffic stream with mean M and variance V is offered to a single link with n circuits. This offered traffic stream is the overflow traffic from a previous link or links and so it will be a rough traffic stream. Wilkinson's ERM represents this rough traffic stream by the overflow traffic which would result when a Poisson arrival stream with arrival rate a_{eq} is offered to a link with n_{eq} circuits. In other words we must find a_{eq} and n_{eq} such that

$$M = a_{eq}E(n_{eq}, a_{eq}) \tag{2.5}$$

and

$$V = M\left(1 - M + \frac{a_{eq}}{n_{eq} + 1 + M - a_{eq}}\right). \tag{2.6}$$

So the overflow traffic stream with mean M and variance V is equivalent to a Poisson arrival stream with intensity a_{eq} offered to a link with n_{eq} circuits. This situation is shown in figure 2.3.

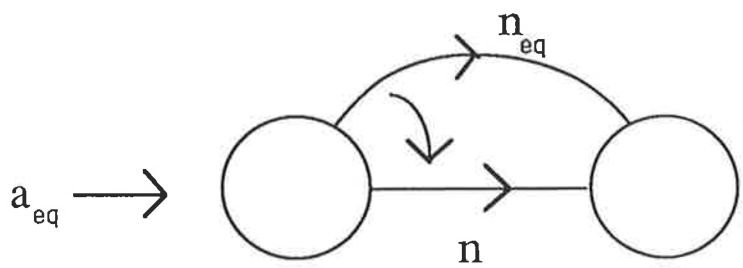

Figure 2.3 *A situation equivalent to that shown in figure 2.2.*

The traffic that overflows from a link with n circuits when it is offered an arrival stream with mean M and variance V can now be calculated by

$$M_{ov} = a_{eq}E(n + n_{eq}, a_{eq}) \tag{2.7}$$

and

$$V_{ov} = M_{ov}\left(1 - M_{ov} + \frac{a_{eq}}{n + n_{eq} + 1 + M_{ov} - a_{eq}}\right). \tag{2.8}$$

The traffic carried on the link is given by

$$M_{ca} = M - M_{ov} \tag{2.9}$$

and the variance of the traffic carried on this link can be approximated by

$$V_{ca} = M_{ca} - (M_{ov} - M_{ca})(n - M_{ca}). \tag{2.10}$$

The ERM can be extended to the case when the traffic offered to a link is not rough but smooth. This technique is called the Extended Equivalent Random Method, EERM (see Bretshneider (1973)). This technique will be necessary when multi-link chains are investigated but is not necessary when calculating the congestions on single link chains.

The situation that occurs when more than one traffic stream is offered to a circuit is discussed in the next section.

2.3 Combinations of Overflow Traffics.

The problem of determining lost traffics when two overflow traffic streams are offered to a single link has been studied by many people. The results of some of this work is shown here as these techniques will be used in the iterative scheme described later in this chapter and some of them will be used in the more general scheme described in Chapter 3. For mixing rough traffic streams the simple network shown in figure 2.4 will be investigated.

Here traffic with intensity a_1 is first offered to link 2 with n_2 circuits and any overflow from this link will be offered to link 1 with n_1 circuits. Similarly a traffic stream with intensity a_2 is offered to link 3 with n_3 circuits, the overflow from this stream is also offered to link 1. Hence we have two overflow traffic streams being

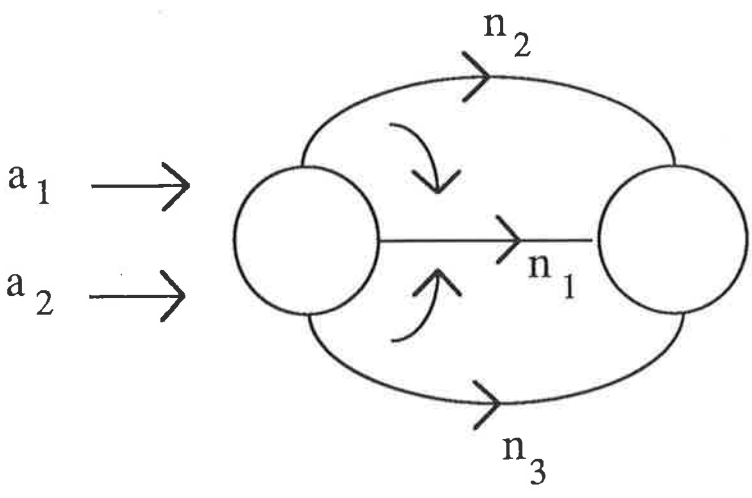

Figure 2.4 *A simple network in which combinations of rough traffic are offered to a single link.*

offered to link 1. The mean and variance of these overflow traffic streams can be calculated using Erlang's loss formula and Riordan's formula respectively. Let M_1 and V_1 be the mean and variance respectively of the overflow traffic stream from link 2 when Poisson traffic of intensity a_1 is offered to n_2 circuits and M_2 and V_2 be the mean and variance of the overflow traffic stream when traffic of intensity a_2 is offered to n_3 circuits. Then the system depicted in figure 2.4 can be represented by the situation shown in figure 2.5.

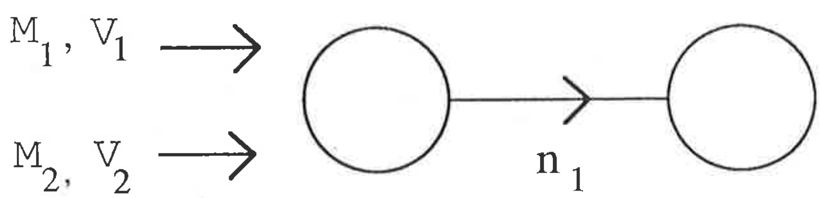

Figure 2.5 *The network shown in figure 2.4 with the overflow means and variances calculated.*

Since these two overflow streams are independent the total mean and variance of the traffic offered to the n_3 circuits is just the sum of the individual means and variances of the two offered traffic streams. Wilkinson's ERM shows how we can determine the total lost traffic from the above system. The ERM can be used to determine the lost traffic from a link when any combination of independent overflow traffic streams is offered to that link. In this case let the mean of the traffic which overflows from link 1 be m and the variance of this traffic be v. We now only have to find a method to determine how much of this traffic is due to the first arrival stream and how much of it is due to the second arrival stream. Olsson (see Berry (1971)) has shown that an approximate method for splitting this overflow traffic m into its two individual traffic components, that we shall call m_1 and m_2, is to split the traffic so that the proportion of the lost traffics is the same as the proportion of the values $V_j + M_j^2/V_j$ for each traffic stream j. We can compare the results we obtain using these methods with exact results obtained by solving the steady state equations numerically for small values of n_1, n_2 and n_3. That is, if the vector of state probabilities at time t is $\mathbf{p}(t)$ then

$$\mathbf{p}(t + \delta t) = \mathbf{p}(t) + \mathbf{p}(t)A\delta t + O(\delta t^2) \tag{2.11}$$

where A is the matrix of transition probabilities and so

$$\frac{d\mathrm{p}(t)}{dt} = \mathrm{p}(t)A. \tag{2.12}$$

Since equilibrium conditions are assumed the derivative of $\mathbf{p}(t)$ with respect to t must be zero. So $\mathbf{p}(t)$ is the solution to the matrix equation $\mathbf{p}(t)A = \mathbf{0}$ such that the sum of the probabilities is 1.0. The application of this well known technique to systems of the form shown in figure 2.4 is examined in more detail in Berry and Coyle (1984). Approximate results obtained by simulation can be obtained for larger values of n_1, n_2 and n_3. Some of these results are shown in table (2.1).

The variances of the individual lost traffic streams must also be calculated, so that if one or both of the lost traffic streams were now to be offered to a third choice route we can, using the ERM, repeat the above procedure and hence determine the lost traffic from this route and so on. For determining the variances of the lost traffic streams Harris and Helm (1976) have proposed the approximate formula,

$$v_j = p_j \left\{ (p_j + (1 - p_j)e^{-p_j n})(v - m) + m \right\} \tag{2.13}$$

where v is the variance of the total overflow traffic from the link and $p_j = V_j/V$ where V is the sum of the offered variances.

We shall now employ all of these techniques in an iterative procedure in order to calculate the traffic congestions on a small one-link chain network.

	Mean	Var.	Exact Lost	Olsson's Lost	% Error
$n_1 = 5$					
S_1	3.383	6.133	1.258	1.309	3.0
S_2	3.383	6.133	1.258	1.309	3.0
$n_1 = 6$					
S_1	6.600	13.811	2.546	2.519	1.1
S_2	2.732	5.268	1.003	0.978	2.5
$n_1 = 2$					
S_1	0.796	1.301	0.190	0.194	2.1
S_2	0.618	0.945	0.143	0.139	2.8
$n_1 = 18$					
S_1	10.759	17.313	0.592	0.604	2.0
S_2	2.705	6.340	0.185	0.205	11.0

Table (2.1) *A comparison of the exact results obtained from the system shown in figure 2.5 with the results obtained using Olsson's splitting formula. The two separate streams of traffic are entitled S_1 and S_2.*

2.4 An Iterative Scheme.

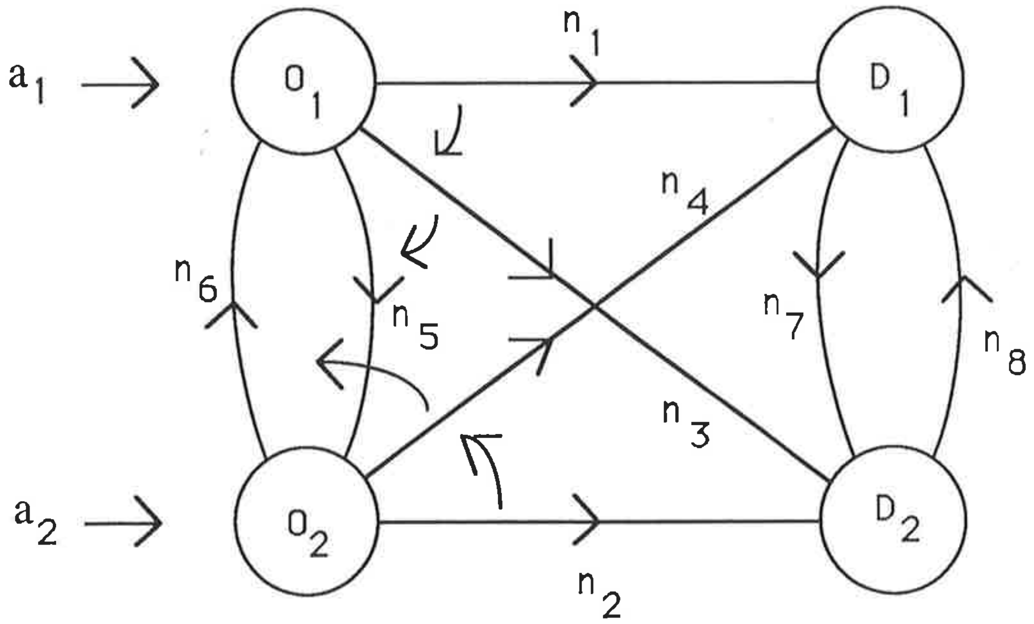

Figure 2.6 *A simple network used to investigate the proposed iterative method for determining congestions in non-hierarchical networks.*

We demonstrate the iterative procedure for determining the end-to-end congestions in a teletraffic network with only single link chains by examining the simple network shown in figure 2.6. The traffic from the ith stream offered to its jth choice route shall have mean M_j^i and variance V_j^i. Therefore, since the ith Poisson offered stream has intensity a_i, $M_1^i = V_1^i = a_i$. A Poisson stream of intensity a_1 is first offered to the route $O_1 - D_1$, the overflow traffic from this route with mean M_2^1 and variance V_2^1 is then offered to the second choice route $O_1 - D_2 - D_1$ and finally the overflow traffic stream from this route, with mean and variance M_3^1, V_3^1, is offered to the final choice route $O_1 - O_2 - D_2 - D_1$. The overflow traffic from the final choice route, with mean m_1, is lost. Similarly, a traffic stream with intensity a_2 is offered to the route $O_2 - D_2$, the overflow from this route with mean and variance M_2^2, V_2^2 is offered to the route $O_2 - D_1 - D_2$ and the overflow traffic from this route with mean and variance M_3^2, V_3^2 is offered to the final choice route $O_2 - O_1 - D_1 - D_2$, any overflow from this final route, with mean m_2, is lost. By choosing $n_5 \geq n_2$, $n_6 \geq n_1$, $n_7 \geq n_1 + n_4$ and $n_8 \geq n_2 + n_3$ we ensure that any traffic overflowing from any one of these routes is due only to the number of circuits on the links n_1, n_2, n_3, n_4 and the intensity of the arrival streams a_1, a_2. Hence for practical purposes we can look at this network as having chains which are only composed of single links and calculate the lost traffics from this network by only looking at these single links. We will try to find some approximate values for m_1 and m_2, the mean traffics lost from this network, using the following iterative scheme.

When we use the ERM to calculate the overflow traffic with mean m and variance v when traffic with mean M and variance V is offered to a link with N circuits, we will denote this procedure by $(m, v) = ERM(M, V, N)$. Our method calculates approximate results for all the traffic streams on the network during each iteration. Since the traffic from one of the arrival streams is dependent on the results obtained from the other traffic stream, as one set of results becomes more accurate this in turn affects the other set of results. So it is expected that the approximations for the lost traffic on the network will rapidly approach the correct values.

2.5 An Overview of the Method.

Step 0: Only look at arrival stream 1.

Calculate M_2^1, V_2^1 by

$$M_2^1 = a_1 E(n_1, a_1),$$

$$V_2^1 = M_2^1(1 - M_2^1 + a_1/(n_1 + M_2^1 - a_1 + 1)).$$

Calculate M_3^1, V_3^1 by

$$(M_3^1, V_3^1) = ERM(M_2^1, V_2^1, n_3).$$

Step 1a: Let $M = a_2 + M_3^1, V = a_2 + V_3^1$. Then $(M_{ov}, V_{ov}) = ERM(M, V, n_2)$ is the mean and variance of the total traffic overflowing from link n_2. Use Olsson's splitting formula to obtain M_2^2 and m_1. Use Harris and Helm's variance formula to obtain V_2^2.

Step 1b: Calculate

$$(M_3^2, V_3^2) = ERM(M_2^2, V_2^2, n_4).$$

Step 2a: Let $M = a_1 + M_3^2, V = a_1 + V_3^2$. Then $(M_{ov}, V_{ov}) = ERM(M, V, n_1)$ is the mean and variance of the total traffic overflowing from link n_1. Use Olsson's splitting formula to obtain M_2^1 and m_2. Use Harris and Helm's variance formula to obtain V_2^1.

Step 2b: Calculate

$$(M_3^1, V_3^1) = ERM(M_2^1, V_2^1, n_3).$$

Step 3: If the results from the present iteration for m_1 and m_2 are sufficiently close to those obtained from the previous iteration stop the algorithm, if the results have not converged sufficiently repeat the above from step 1a.

2.6 Some Results.

If all the formulae used in this procedure were exact then the exact and calculated results would be the same. However, since this is not the case some error will exist in the results. The size of these errors determines how useful this method can be, and as the accuracy of these approximate formulae increases, the accuracy of this method also increases. The traffics lost from the network in figure 2.6 can be found exactly by solving the relevent steady state equations. Table (2.2) compares the exact and approximate results for some sets of network parameters. The results show reasonably small errors involved in calculating the lost traffics from this network. For a large number of cases the errors involved using this iterative scheme were calculated and the average percentage error was less than four percent. It usually took about three iterations before the results converged to this accuracy. So we conclude that this iterative scheme works well for cases when only single link chains are used. In the next chapter this method will be extended so that it can be used on networks with multi-link chains.

n_1	n_2	n_3	n_4	a_1	a_2		m_1	m_2
3	1	2	1	5.5	2.2	Exact	1.707	0.731
						Approx.	1.697	0.742
						% err	0.58	1.6
4	1	3	1	7.0	1.1	Exact	1.485	0.213
						Approx.	1.474	0.231
						% err	0.80	8.4
4	4	2	3	6.6	7.7	Exact	1.814	1.877
						Approx.	1.799	1.882
						% err	0.90	1.3
8	8	3	3	10.0	10.0	Exact	1.197	1.197
						Approx.	1.193	1.200
						% err	0.39	0.29
15	9	6	2	20.0	10.0	Exact	1.898	1.216
						Approx.	1.985	1.181
						% err	0.650	3.3

Table (2.2) *A comparison of exact results for the simple non-hierarchical network and the results obtained using the iterative scheme described above. For each set of parameters shown in the table the first row in the final column is the exact results, the second row are the results obtained using our method and the third row is the percentage error of the results.*

CHAPTER 3

THE EXTENDED METHOD.

In chapter 2 an iterative scheme was described which calculates the congestions in telecommunication networks with only single link chains. The results from this method are quite good since the various traffic formulae used are fairly accurate. In this chapter the algorithm will be extended to the more general case when chains with more than one link can be used. The problems associated with analysing networks with multiple link chains will be discussed and then a solution to these problems proposed. This technique of analysing a telecommunications network differs from other techniques known to the author in that it focuses on the traffics carried on the chains in the network.

3.1 An Analysis of Multi-link Chains.

In the discussion that follows we will assume that crankback is implemented. We will later extend these results to include the case when crankback is not used.

One approach which has been used to solve the problems discussed in this thesis is to offer the traffics from the various chains in a network to each single link in these chains. A lot of research has gone into looking at results for this approach, which includes finding methods of determining the total lost traffics and creating approximate splitting formulae for the separate streams of lost traffic. As was shown in chapter 2, for systems with only single link chains this approach works very well. For systems in which multi-link chains are used, however, this approach does not always give satisfactory results.

Figure 3.1 shows part of a large network. An overflow stream, stream 1, with mean M_1 and variance V_1, is offered to the chain composed of links 1,3 and 4.

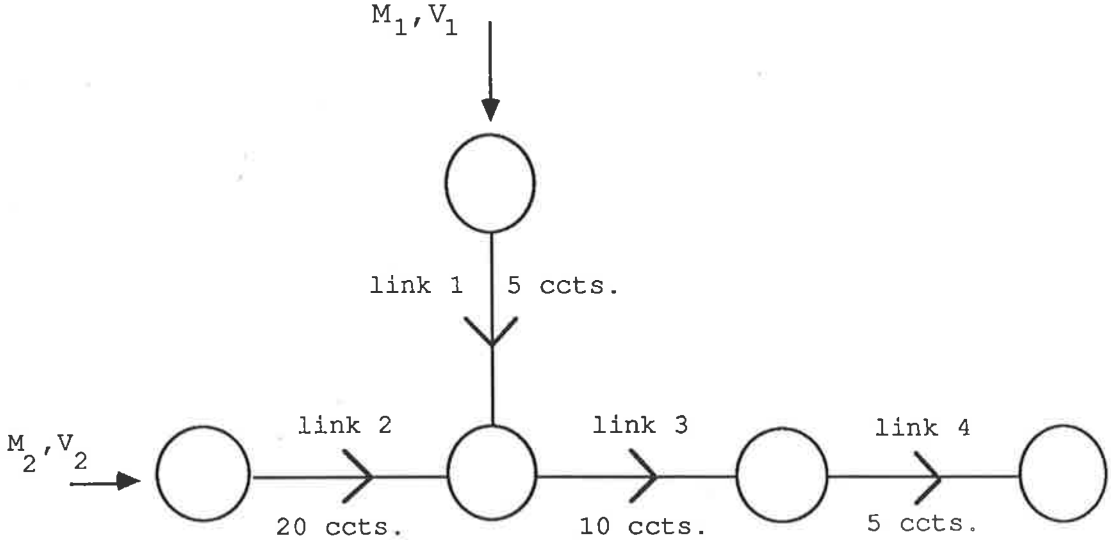

Figure 3.1 *Part of a large telecommunications network.*

Another overflow stream, stream 2, with mean M_2 and variance V_2 is offered to the chain composed of links 2 and 3. Apart from link 3 all of these links also have other traffic streams offered to them from various chains in the network. If we want to determine the amount of traffic lost due to link 3 from the streams M_1, V_1 and M_2, V_2 respectively we must first of all determine the traffic which is offered to this link from these two arrival streams. If this is known we can then, using Wilkinson's Equivalent Random Method , approximate the total traffic lost from the network due to this link.

A naive approach to the problem would be to let the streams of traffic, M_1, V_1 and M_2, V_2, be offered directly to link 3 and to calculate the traffic lost due to this link. However, this could lead to more traffic from stream 1 being carried on link 3 than is possible, since any traffic carried on link 3 from stream 1 must firstly have been carried on link 1. Link 1, having only five circuits, will permit only a certain amount of traffic from stream 1 to flow on this chain. So, this simple technique will not yield very accurate results.

Another possible approach would be to offer M_1, V_1 to link 1, determine how much traffic is carried on this link and then offer this traffic to link 3. Similarly, the traffic M_2, V_2 will be offered to link 2 and the traffic carried on this link will also be offered to link 3. The previous problem has been solved since the traffic carried on link 1 which is now offered to link 3 cannot be greater than the amount of traffic which can be carried on link 1. These two traffic streams are offered to link 3, the total lost traffic is calculated and then the traffic lost from each of the two separate arrival streams is calculated. We must now offer the traffic carried on link 3 from the first stream to link 4 which has only five circuits, as well as having a number of other traffic streams offered to it. We would expect that a lot of the traffic offered to link 4 could not be carried and so will be lost. This means that the traffic that was calculated to have been carried on link 3 in fact cannot be carried on this link, which also means that more traffic from the second stream could have been carried on link 3. Thus the traffic streams offered to link 4 have affected the result we obtained for the traffic overflowing from the chain of the second offered stream which does not even include this link.

For the simple system in figure 3.1 we could have offered the traffic M_1, V_1 to link 4 initially and then continued in the manner described in the previous paragraph, since the order in which we put the links in a chain should make no difference to the final result as we are assuming crankback is used. However, if we also consider another stream offered to link 2 the problem becomes even more complicated. We will now have to decide what the traffic offered to link 2 due to this new stream will be, which may involve looking at the links in the chain associated with this stream and the chains of the other streams using these links and so on. Soon the problem of investigating the traffic overflowing from one link will involve the whole network. These problems occur because we cannot talk about traffic streams being offered to links independently of the rest of the chain with which these arrival or overflow streams are associated. Since no one part of the network is completely independent from any other part even the simplest problem may soon involve looking at a large number of overflow streams and links.

We have attempted to solve the problems presented above by considering the traffics carried on the chains of the network. The traffic carried on a chain is the same at each single link in this chain and so the difficulty in having to look at the other links in these chains no longer exists. The traffic offered to these chains and the position of links in the chains are no longer important. If we want to determine how much traffic is lost from the offered stream M_1, V_1 when it is offered to the chain with links 1, 3 and 4, we only need to know how much traffic is carried on each of these links from the other chains in the network. It is now a much simpler matter of offering M_1, V_1 to these three links in turn to determine how much of the traffic from this stream is lost.

3.2 The Traffics Carried on a Chain.

Figure 3.2 depicts a traffic stream with mean M and variance V offered to a chain of links in a network. We will call this traffic stream the chain offered traffic. This chain offered traffic is either an arrival stream or a stream of overflow traffic from a previous chain. Each link in this chain carries some traffic which is the sum of all the other traffic streams that are carried on that link. This carried traffic does not include the traffic which is carried from the chain offered traffic. The problem we must now solve is to calculate how much of the chain offered traffic is carried on this chain and how much of it overflows from this chain. Given these

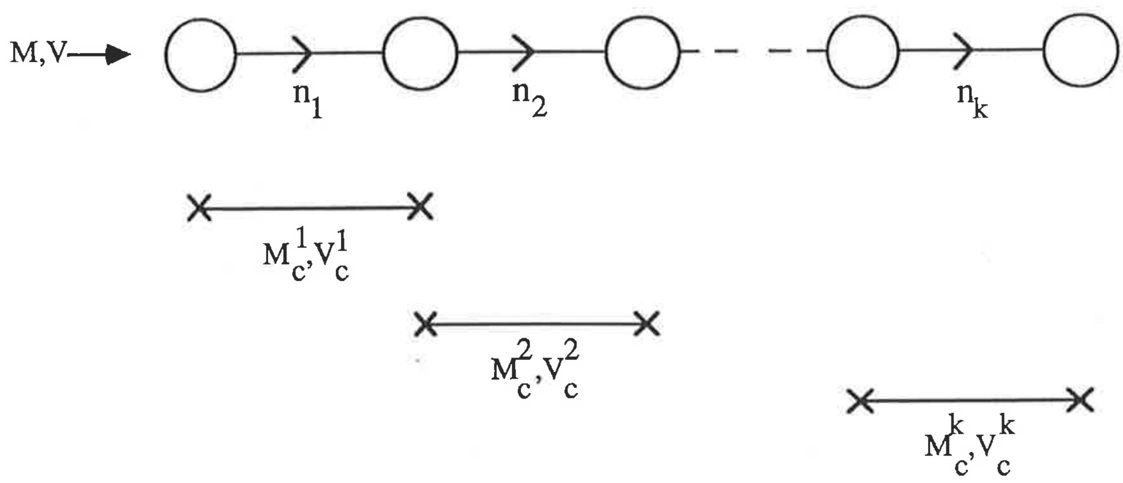

Figure 3.2 *A chain in a network. The traffic offered to this chain has mean M and variance V. The traffics carried on the links in this chain, with means M_c^i and variances V_c^i, are represented by the lines below the chain.*

results we could then look at the next chain in this network and find the same results for this chain and its associated offered stream and so on.

3.3 Finding the Traffic Lost from a Chain.

In chapter 2 we investigated ways of determining the traffic lost from a link when the traffics offered to that link are known. If we can restate our problem in terms of offered traffics then some of these techniques can be used.

For each link in figure 3.2 there is an associated stream of carried traffic, the sum of all the traffic carried on this link not including that carried from the chain offered traffic. For each of these carried streams there is an equivalent traffic stream which when offered to the link will result in the correct amount of traffic being carried on this link. This equivalent offered stream may be calculated solely from the traffic carried on the link, which is known, and the chain offered traffic. This is not the same as the procedure mentioned in section 3.1 in which offered traffic streams were combined, offered to a link and the traffic carried on the link calculated. In our algorithm the offered traffics are calculated from the carried traffics, not the other way round. Also the carried stream here is the sum of all the carried traffics on this link, the procedures described earlier looked at each individual offered stream one at a time. Once the carried traffic streams have been replaced by equivalent offered streams then the system in figure 3.2 can be represented by the system in figure 3.3.

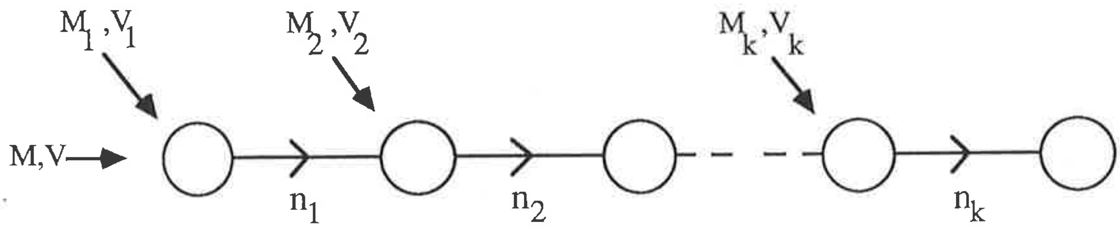

Figure 3.3 *The chain in figure 3.2 but with the traffics carried on the links replaced by traffics offered to the links. If we choose the offered traffics correctly the traffic lost from this system would be the same as the traffic lost from the chain shown in figure 3.2.*

In figure 3.3 the traffics carried on the links have been replaced by streams of traffics offered to the links. It should be noted that these streams of offered traffic are also dependent on the offered traffic stream M, V. If this stream changes then the other offered streams will also change. What is important is that the amount of traffic carried on this link due to this calculated offered stream is the same as the amount of traffic we know is carried on the link.

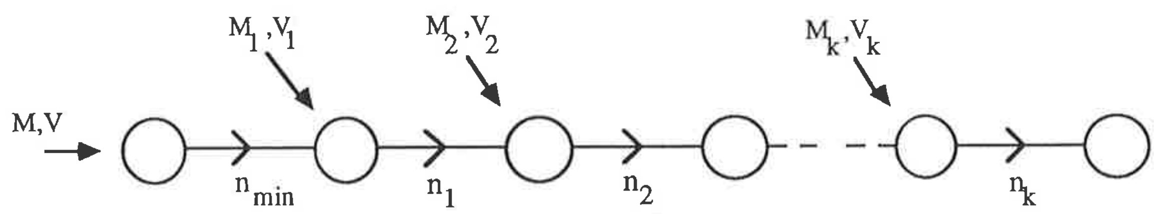

Figure 3.4 *The chain in figure 3.3 with an extra link added. The traffic lost from this chain is equivalent to that lost from the chain in figure 3.3.*

For the chain depicted in figure 3.3 let

$$n_{min} = \min_{1 \leq i \leq k}(n_i),$$

that is, n_{min} is the minimum number of circuits that are in any link of this chain. Now we introduce an extra link to the chain in figure 3.3 which has n_{min} circuits. No other traffic will be offered to this link apart from the chain offered traffic M, V. The amount of traffic carried on this new chain is the same as the amount of traffic carried on the chain depicted in figure 3.3. This is so since a call will be blocked by this new link if and only if all the circuits on this link are being used by calls from the chain offered traffic stream. However, when this occurs the call would have been blocked anyway since there is at least one link in the chain with the same number of circuits. We will find this simple result useful in trying to find the amount of traffic lost from this chain.

From the above we know that the amount of traffic lost from the chain shown in figure 3.3 is the same as the amount which is lost from the chain shown in figure 3.4, with n_{min} defined as above. The reason we introduce this extra link is because the maximum amount of traffic that could be carried on the chain is the maximum amount of traffic that could be carried on the link with n_{min} circuits. If no other streams of traffic were carried on the links in this chain, i.e. $M_i = 0$ for all i, then the amount of traffic lost from this chain would be just the amount of traffic lost when M, V is offered to the single link containing n_{min} circuits. For the case when the other traffics are not zero we will use an approximate technique to find the traffic blocked on this link due to the other traffic streams carried.

3.4 A Summary of our Algorithm.

For any given network in which a set of routeing rules, the intensity of the arrival streams and the number of circuits on each link are known, approximations to the values of the end-to-end congestions of this network are found in the following way.

Let $S(k)$ be the set of all the chains (i,j) that includes the link k, where (i,j) is the chain from the jth choice route of the ith arrival stream. Also let $m_{ca}^{(i,j)}, v_{ca}^{(i,j)}$ be the mean and variance respectively of the traffic which has been calculated so far to be carried on the chain (i,j). Initially $m_{ca}^{(i,j)} = v_{ca}^{(i,j)} = 0$ for all the chains (i,j). The following must be repeated for every arrival stream in order to complete one iteration of this algorithm. At the end of each iteration a test is performed to see if the results have converged.

For the ith arrival stream with intensity a_i :

For the jth chain in the network that this stream is offered to :

Let M_{of}, V_{of} be the mean and variance respectively of the traffic offered to chain (i,j). Either this is the traffic which has overflowed from a previous chain, or if this is the first chain that this stream is offered to, $M_{of} = V_{of} = a_i$.

Let n_{min} be the minimum number of circuits contained in any link of chain (i,j) and append a link k' with n_{min}circuits which carries no other traffic.

Using Wilkinson's Method find the amount of the offered traffic that is carried on link k' with mean M^*_{of} and variance V^*_{of}. We will denote this operation by ERM_c and so

$$(M^*_{of}, V^*_{of}) = ERM_c(M_{of}, V_{of}, n_{min}).$$

For each link k in chain (i,j) :

Calculate the traffic carried on link k M_{ca}, V_{ca} from other traffic streams using link k, that is :

$$M_{ca} = \sum_{\substack{(i',j')\in S(k)\\(i',j')\neq(i,j)}} m_{ca}^{(i',j')}$$

and

$$V_{ca} = \sum_{\substack{(i',j')\in S(k)\\(i',j')\neq(i,j)}} v_{ca}^{(i',j')}.$$

The traffic which is carried on link k from the offered traffic M^*_{of}, V^*_{of} must now be calculated. This traffic will have mean M^*_{ca} and variance

V_{ca}^*. We will call the procedure that will give us this result the link congestion method, or LCM. So

$$(M_{ca}^*, V_{ca}^*) = LCM(M_{of}^*, V_{of}^*, M_{ca}, V_{ca}, n_k).$$

This procedure will fall into two related parts. As discussed earlier the first part is to find the equivalent offered traffic with mean M_{of}^E and variance V_{of}^E, which when offered to link k will result in the correct amount of carried traffic, M_{ca}, V_{ca}. The second part of this procedure is to find out how much of the traffic offered to link k is carried on link k, that is, given the two traffic streams offered to link k, M_{of}^*, V_{of}^* and M_{of}^E, V_{of}^E, find out the total traffic carried and split it up into the two respective streams.

The traffic carried on link k in chain (i,j) with mean M_{ca}^* and variance V_{ca}^* now becomes the traffic offered to the next link in chain (i,j) where the above process is repeated, so $M_{of}^* = M_{ca}^*$ and $V_{of}^* = V_{ca}^*$. If this is the final link of the chain M_{ca}^*, V_{ca}^* is the traffic carried on this whole chain and so $m_{ca}^{(i,j)} = M_{ca}^*$ and $v_{ca}^{(i,j)} = V_{ca}^*$. These results are used later in the program when looking at the chains which have links in common with chain (i,j).

Now calculate the traffic which has overflowed from chain (i,j):

$$M_{ov} = M_{of} - M_{ca}^*,$$

and approximate the variance of the overflow traffic as if chain (i,j) was a single link so that

$$V_{ov} = M_{ov}(1 - M_{ov} + a_{eq}/(n_{eq} + 1 + M - a_{eq}))$$

where a_{eq} and n_{eq} are calculated in the same way as in Wilkinson's Equivalent Random Method.

If chain (i, j) is the last chain that this traffic stream is offered to then the total traffic lost from this arrival stream is $M_{lost}^{(i)} = M_{ov}$. If chain (i, j) is not the last chain, then set $M_{of} = M_{of} - M_{ca}$ and $V_{of} = V_{ov}$ and repeat this process for the next chain.

The end-to-end congestion for the ith arrival stream is given by

$$\frac{M_{lost}^{(i)}}{a_i}.$$

After the traffic lost from every stream offered to this network has been calculated compare the values obtained for the end-to-end congestions from this iteration with the results calculated from the previous iteration. If these two sets of results are sufficiently close to one another then the results have converged and the desired results have been achieved.

3.5 Some Remarks on the Link Congestion Method.

An algorithm to find the congestion in a telephone network has been presented in section 3.4. There are, of course, many problems still associated with this algorithm. In the Link Congestion Method we offer some traffic to a link which has a known amount of traffic already being carried on it. Techniques must be devised to find the amount of traffic lost and carried on this link. Some techniques for approximating these traffics are presented here.

The problem we must solve in order for our algorithm to give accurate results is shown in figure 3.5, which depicts a single link in a chain of a network. We know the traffic from other streams carried on this link, with mean M_{ca} and variance V_{ca}, and we also know that the traffic carried on the previous links in this chain has mean M^*_{of} and variance V^*_{of}. We want to know how much of this traffic will be lost from this link due to the other traffics carried on it. This corresponds to the procedure that we called the Load Congestion Method, or LCM for short.

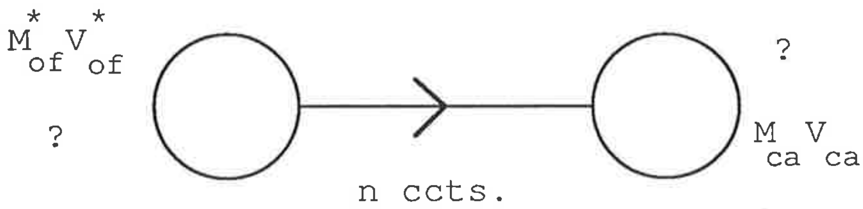

Figure 3.5 *A link in a chain with the associated offered and carried traffics.*

We must investigate the second part of the LCM problem before the first part. When we try to find the equivalent offered traffic in the LCM we must firstly decide how the traffic carried on this link is to be calculated. An analogy to this is, of course, Wilkinson's Equivalent Random Method. It is necessary to know the Erlang loss formula and Riordan's formula before the ERM can be formulated. In this section we will investigate the problem of finding the carried traffics on the link given the offered traffics keeping in mind that this problem will be inverted as part of the LCM.

The situation shown in figure 3.6 is the one which we are trying to solve. We know the values of the offered traffics and we wish to know the values of the

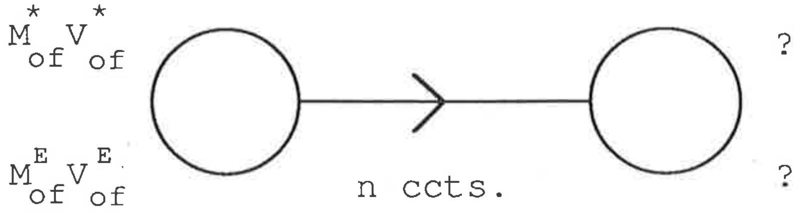

Figure 3.6 *A link in a chain with the associated offered traffics.*

carried traffics on this link, call it link k. We cannot just sum the two sets of offered traffics and then offer them to the link using the ERM. The results obtained this way would be wrong. The reason for this is that the offered traffic M^*_{of}, V^*_{of} has already been offered to and carried on another link. If no other traffic were carried on link k, that is $M^E_{of} = 0$, then all the traffic offered to link k, with mean M^*_{of} would be carried. We have ensured this will happen by offering the chain offered traffic to a link with n_{min} circuits and calculating how much traffic will be carried on this link. Here if any of the traffic carried on link k is blocked on the chain it is because of the other traffic streams carried on this network.

So we need a different technique. The approximate method we have tried is as follows. Firstly offer the sum of the two offered traffics to link k and find the amount of traffic overflowing from link k using the ERM, the lost traffic calculated this way is m_2. Next offer just the traffic carried on the chain so far, M^*_{of}, V^*_{of}, to link k and calculate how much of this traffic is lost using the ERM, the lost traffic calculated this way is m_1. Now we use $m_2 - m_1$ as an approximation for the total traffic that is lost due to link k. The reasoning behind this is that we know that if $M^E_{of} = 0$ then the traffic lost from link k is in fact zero. So we use

m_1 as an approximation to the extra traffic that would have been lost if M^*_{of}, V^*_{of} had not been previously offered to any links. So m_1 is the traffic lost from link k due to the restricted number of circuits on link k. Also m_2 is the amount of traffic lost from link k due to the combined effect of the number of circuits and the other traffics carried on link k. So $m_2 - m_1$ is taken as an approximation to the amount of traffic lost on link k due to the traffic carried on link k. For the case when $M^E_{of} = 0$ this result is exact, it is also exact when $M^*_{of} = 0$. This is a fairly rough estimate, but the results we get are actually quite reasonable.

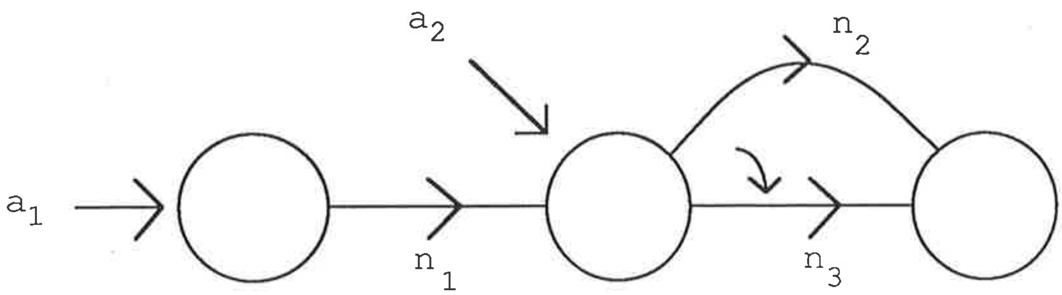

Figure 3.7 *A simple network that can be investigated to determine the accuracy of our approximate method for calculating the traffics carried on a link in a chain.*

To determine the accuracy of our method for finding the traffic lost from a link in a chain, we will investigate the simple system shown in figure 3.7. Note firstly that by Wilkinson's ERM we can represent the offered traffics M^E_{of}, V^E_{of} by the offered stream a_2 offered to the circuits n_2. This is true even though the traffic stream M^E_{of}, V^E_{of} is itself an equivalent offered traffic stream. In the system in figure 3.7 we offer one arrival stream of intensity a_1 to the two link chain, link 1 with n_1 circuits and link 3 with n_3 circuits, where $n_1 \leq n_3$. Another traffic stream of intensity a_2 is offered firstly to link 2 with n_2 circuits and the traffic overflowing from this link is then also offered to link 3. For sufficiently small values of n_1, n_2 and n_3 we can find exact results for this simple system by solving the steady state equations using the technique that was briefly described in chapter 2. We can also use the method described in the last paragraph to find approximate values for the traffics lost from this system. Some of these results are presented in table (3.1).

3.6 A Splitting Formula.

Having found the total lost traffic from the simple network in figure 3.6 we must now determine how the traffic is split between the two offered traffic streams. The splitting formula we use is one of many that have been attempted. This formula gave the best results out of all the techniques tried when compared to the exact results determined by the method described in the previous paragraph. We want to determine how much of the total traffic lost from the link, with mean M_{ov}, is due to the equivalent offered traffic stream with mean and variance M_{of}^E, V_{of}^E, this overflow traffic will have mean M_{ov}^E. We also want to determine how much of the total traffic lost from the link, with mean M_{ov}, is due to the traffic stream which has been carried on the previous links in this chain with mean and variance M_{of}^*, V_{of}^*, this overflow traffic will have mean M_{ov}^*. To do this we firstly calculate

$$p^E = \begin{cases} M_{of}^E p_{ca}^{0.4(V_{of}^E/M_{of}^E)} & V_{of}^E < M_{of}^E, \\ M_{of}^E p_{ca}^{0.4(M_{of}^E/V_{of}^E)} & M_{of}^E < V_{of}^E \end{cases} \tag{3.1}$$

and

$$p^* = M_{of}^* p_{ca}^{(M_{of}^*/V_{of}^*)},$$

where p_{ca} is the proportion of the total offered traffic which is carried on this link, i.e.

$$p_{ca} = \frac{M_{of}^E + M_{of}^* - M_{ov}}{M_{of}^E + M_{of}^*}.$$

Now we find

$$M_{ov}^E = M_{ov} \frac{p^E}{p^E + p^*}$$

and

$$M_{ov}^* = M_{ov} \frac{p^*}{p^E + p^*}.$$

When we perform this splitting operation we will say

$$(M_{ov}^E, M_{ov}^*) = SPL(M_{of}^E, V_{of}^E, M_{of}^*, V_{of}^*, M_{ov}).$$

Once the overflow traffics have been calculated it is an easy matter to also find out the carried traffics, M_{ca}^* and M_{ca}^E. In table (3.1) a few comparisons are made between the results obtained using this splitting formula and exact results for the system shown in figure 3.7.

Using the notation from chapter 2 we can use the above method to approximate the traffics lost from the system in figure 3.7 in the following way. Find the traffic from the stream 2 overflowing from link 2

$$M_{of}^E = a_2 E(n_2, a_2)$$

$$V_{of}^E = M_{of}^E (1 - M_{of}^E + a_2/(n_2 + M_{of}^E - a_2 + 1)).$$

Find the traffic from the first stream carried on link 1

$$M_{of}^* = a_1(1 - E(n_1, a_1))$$

$$V_{of}^* = M_{of}^* - (a_1 - M_{of}^*)(n_1 - M_{of}^*).$$

Now find the traffic that would overflow from link 3 due to the sum of the two offered traffics

$$(m_2, v_2) = ERM(M_{of}^E + M_{of}^*, V_{of}^E + V_{of}^*, n_3).$$

Also find the traffic that would overflow from link 3 if only the traffic carried on link 1 is offered to this link

$$(m_1, v_1) = ERM(M^*_{of}, V^*_{of}, n_3).$$

The total traffic overflowing from link 3 is approximately

$$M_{ov} = m_2 - m_1.$$

This overflow traffic is now split up into its respective streams

$$(M^E_{ov}, M^*_{ov}) = SPL(M^E_{of}, V^E_{of}, M^*_{of}, V^*_{of}, M_{ov}).$$

The traffic carried on this link from traffic stream 1 is

$$M^*_{ca} = M^*_{of} - M^*_{ov}$$

and the traffic carried on this link from traffic stream 2 is

$$M^E_{ca} = M^E_{of} - M^E_{ov}.$$

Table (3.1) shows some comparisons between the exact results and the results obtained using the above technique.

3.7 The Inverse Problem.

We have found out how to approximate the traffics carried on a link when we know the offered traffics. We must now work out the inverse problem, that is determining the traffics offered to this link when we know the traffics carried on this link. To do this, we use an iterative search technique varying M^E_{of} on each search iteration until the traffic carried on the link due to M^E_{of} is equal to the required result, M_{ca}. We must firstly find a way of determining V^E_{of} for any M^E_{of}

n_1	n_2	n_3	a_1	a_2		total	M^*_{ca}	M^E_{ca}
3	3	3	3.0	2.0	proc.	1.37	1.17	0.20
					exact	1.29	1.09	0.19
					% err.	5.6	6.5	0.3
10	5	15	30.0	25.0	proc.	35.70	21.77	13.93
					exact	35.90	22.17	13.73
					% err.	0.6	1.8	1.4
5	25	25	30.0	70.0	proc.	51.81	25.28	26.53
					exact	51.65	26.07	25.58
					% err.	0.3	3.1	3.6
20	45	35	70.0	55.0	proc.	52.81	50.12	2.69
					exact	53.45	50.74	2.71
					% err.	1.0	1.1	0.8
45	65	50	100.0	125.0	proc.	111.6	66.75	44.71
					exact	111.2	67.84	43.41
					% err.	0.2	1.6	2.9

Table (3.1) *A comparison of the traffic lost from the simple network shown in figure 3.7 obtained using exact analysis and the results obtained using the methods described in this chapter. The first row for each set of data is the results obtained by the described scheme, the second row is the exact results and the third row gives the percentage error.*

we choose to use. The technique we have used is a very rough method which may be improved upon later.

We would expect the variance to mean ratio of this offered traffic, $Z = V^E_{of}/M^E_{of}$, to depend on the variance to mean ratios of the traffics offered to the chains which include this link. These individual variance to mean ratios will be taken into consideration in proportion to the amount of traffic from each of the streams carried on this link. Let M^i_{of}, V^i_{of} be the traffic offered to chain i which includes the link in question and let M^i_{ca} be the traffic from this offered stream which is carried on this link.

Then our estimate of Z is given by :

$$\frac{\sum_i W_i}{\sum_i M_{ca}^i} \tag{3.2}$$

where

$$W_i = \frac{V_{of}^i M_{ca}^i}{M_{of}^i}. \tag{3.3}$$

Since we have now found an approximation for Z we can, for any M_{of}^E, find V_{of}^E since $V_{of}^E = ZM_{of}^E$. By using this simple formula for determining the variance we have avoided having to use a two parameter search scheme. In such a two parameter scheme, given M_{of}^E, we would have to use a search scheme to find a suitable V_{of}^E so that when this traffic was offered to the link the traffic carried on this link has mean M_{ca}^E and variance V_{ca}^E. In a two parameter search scheme the value of V_{ca}^E is taken into consideration therefore we would expect more accurate results using this scheme. The added complexity and time needed, however, would not make this scheme a realistic proposition. If a simple relationship between $V_{of}^E, M_{of}^E, M_{ca}$ and V_{ca} could be found a simple more accurate system may be devised. It is expected, however, that no simple relation exists without introducing the values of the total carried traffic on this link, which would depend in turn on M_{of}^E and V_{of}^E. Now our algorithm for finding the congestions in our telecommunications network is complete for the case when crankback is implemented.

3.8 No Crankback.

With respect to the algorithm presented in the preceding sections of this chapter, it is easier to find results for the case where crankback is used than if crankback is not used. For networks without crankback, however, we can still use this procedure to approximate the congestions in a telecommunications network after certain modifications have been made.

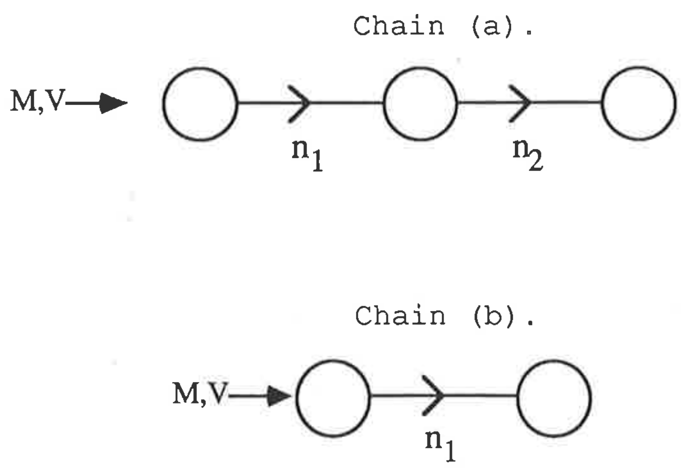

Figure 3.8 *Chain (a) is a two link chain. Chain (b) is just the first link of chain (a) with the same traffic offered to it.*

Two chains which are part of a large network are depicted in figure 3.8. We can find the traffic lost from these chains and the traffic streams carried on these chains whether or not crankback is used. If chain (a) is not the last choice route for an offered traffic stream then some calls overflowing from this chain will be offered to the next choice route for this stream. If crankback is used all the calls that overflow from this chain will be offered to the next choice chain. Without crankback some of the calls which cannot be carried on this chain will be offered to the next choice route, the other calls which cannot be carried on this chain are lost completely from the network. The amount of traffic which is calculated to be carried on chain (a) is independent of whether crankback is used or not. Let this traffic be $M(a)_c$, and so $M_{ov} = M - M(a)_c$ is the amount of traffic not carried on this chain. The traffic carried on chain (b), $M(b)_c$, is the amount of traffic whose signal would reach the second node in chain (a), without crankback all of this traffic is either carried on chain (a) or completely lost from the network. We know $M(a)_c$

is the amount of traffic carried on chain (a) so the amount of traffic lost from chain (a) and not offered to another route is approximately $M_{lost} = M(b)_c - M(a)_c$. The amount of traffic offered to the next choice route is $M_{ov} - M_{lost}$.

This scheme can be generalised for networks which have chains with more than two links, so we can use our procedure on networks with or without crankback.

3.9 Reservation Schemes.

To lessen the amount of traffic lost from a network it is sometimes useful to introduce a system of circuit reservation. We are interested in networks with two links per overflow chain, however the effects we describe will also occur in networks in which the overflow chains have more than two links. When a large amount of traffic is offered to a network and congestions are high, a lot of traffic overflows from the direct first choice route and is offered to the overflow chains. A call using an overflow route uses two links for the duration of that call instead of the one link it would use if it was using the direct route. As a result of this, calls which are offered to the links this call is using cannot get on, and these calls are either lost or use another two link chain. Although the number of possible routes a call may take increases the congestion increases.

A possible way to reduce this effect is to introduce a circuit reservation scheme. A possible reservation system is one in which, when a certain percentage of circuits on a link are busy, only calls using this link as a direct route can use the remaining circuits. Any other calls trying to use one of these circuits are blocked on this link and either use another route on this network or are completely lost from the network. Unfortunately, this reservation scheme cannot be modelled by the method described in this chapter. However, a scheme which approximates this one can be implemented. This approximate scheme is to reserve a certain percentage of the circuits in a link only for direct routed calls. All the other circuits in this link can be used by any type of call and are called common circuits. Calls trying to use a direct route try first to get on one of the common circuits and only if all of these circuits are busy is the call then offered to the circuits reserved only for direct calls. If these circuits are all busy the call is then offered to the overflow routes. This situation is depicted in figure 3.9.

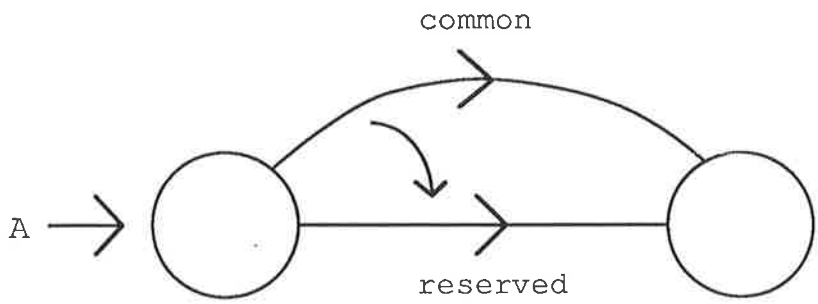

Figure 3.9. *A simple circuit reservation scheme.*

This scheme can be incorporated very easily into the method which has been described in this thesis by simply calling the reserved circuits a separate link. The second choice route of any offered stream's calls is this extra link and only direct routed calls may ever use this link.

The difference between this scheme and that previously described is that only a percentage of all the circuits in the links have to be busy for the first scheme to start reserving circuits for direct calls. It is possible, using the second scheme, to have more than this percentage of circuits occupied on the common plus reserved circuits and still have an overflow call being carried on the reserved link. If only one circuit in the whole link is not occupied, and this is one of the common circuits, then it can be used by a call overflowing from a previous chain. In the first scheme this situation is not possible. These two reservation systems are so similar in practice that the simulation results obtained for them are very close. So, in the results in the next chapter we will use this second reservation scheme to approximate the results from the first reservation scheme, this first scheme is currently favoured by Australian Telecom.

In this chapter a method for determining congestions in a telecommunications network has been outlined. An explanation of the ideas has been given and some of the problems associated with this method outlined. In the next chapter results obtained using this method are compared to simulation results.

CHAPTER 4
CONCLUSIONS

4.1 Results and Simulation.

If the procedure that we have developed is to be used for practical purposes it must not take too much time to give the desired results. A very accurate program which takes a prohibitively long time to give realistic results is not preferred to a program which quickly gives reasonably accurate results. When dealing with real networks, the values of the traffics offered to this network are only approximations with the errors involved as high as ten percent. Any algorithm which gives results with an accuracy less than about five percent is calculating the traffics to an unnecessarily high accuracy. The errors that we would hope to achieve using this procedure would be around the ten percent mark and yet we desire also to keep the run time of the program as low as possible. If a change in our method could lead to a program which takes a shorter time to run, even though the results may not be as accurate, then this change should be made. Therefore, when we are looking for ways of solving problems associated with this procedure we must keep in mind that to be useful the techniques we use must not take too long to run. In many cases we must use a technique or formula, even though we realise it is not the most accurate, because it is simple and gives reasonably accurate results.

Especially for the heuristic procedures we are using, the traffic variance is not as important as the traffic mean when calculating values of performance measures, so we can tolerate larger errors in the variance than in the mean. To use formulae which are more accurate would sometimes require added complexity and may only give a slight improvement to the final results. It may be possible that some of the procedures that we have used in our algorithm could be improved to give more accurate results without increasing the running time of the program substantially. If this is the case then the system described in this thesis may, as these more accurate techniques are developed, give better results. Thus the results shown in this chapter may be taken as an upper bound on the accuracy which can be achieved using the scheme described in chapter 3.

4.2 A Four Node Network

Before using our procedure to investigate properties of non-hierarchical networks, the accuracy of the results obtained from this procedure must be established. Since the networks that we are using are very large and complicated no simple analytical technique can be used to find results for these networks, it is necessary to use simulation. Because all the arrivals to a network are assumed to be Poisson and all the service times are assumed to be negative exponentially distributed, a Markov Chain Simulator is used to approximate the behaviour of these networks. In a Markov system the probability an event occurs is independent of the history of this system. So a Markov Chain Simulator only needs to record the state of the Markov system and so is much quicker than the stored event simulators that would have to used if the system was non-Markovian.

A telecommunications network has been investigated to determine the accuracy of the procedure described in chapter 3. This network is shown in figure 4.1. Any stream of offered traffic from any node in the network to any other node will firstly be offered to the direct chain composed of a single link. If this link is blocked, the next route used is the two link chain whose first link is the first link clockwise of the direct link. If this chain is blocked, the next route attempted is the two link chain whose link is the first link clockwise of the last route's first link

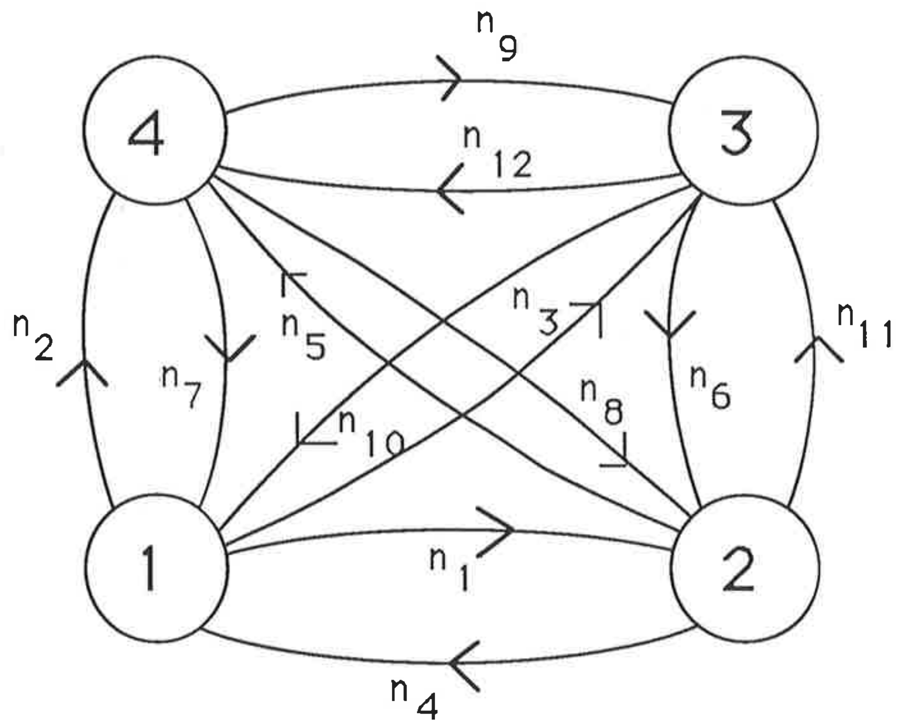

Figure 4.1 *A four node non-hierarchical telephone network which allows us to investigate the results obtained for the method descibed in Chapter 3.*

which is the only possible two link chain left that this call can take. For example, a call from node 1 to node 2 will firstly attempt to get on the direct single link chain which has n_1 circuits. If this link is blocked, the two link chain from node 1 to node 4 and then onto node 2 will then be attempted, since the link from node 1 to node 4 with n_2 circuits is the first link out of node 1 clockwise of the link from node 1 to node 2. If crankback is used all the calls overflowing from this route are offered to the last remaining two link chain, that is the chain from node 1 to node 3 and then onto node 2. If crankback is not used then a call which does not get onto the second choice route because the link from node 1 to node 4 is blocked will be offered to this final choice route. If the link from node 1 to node 4 is not blocked and a call's signal gets to node 4 then this call cannot be offered to the final choice route. If the link from node 4 to node 2 is not blocked the call will then be taken on this route, if it is blocked the call will be lost, whether or not the last chain is available.

Simulation results for this network have been calculated to determine the accuracy of the procedure described in this thesis and these results are shown in tables (4.1) to (4.10). We are primarily interested in the traffics lost from these networks although the procedure can also be used to determine the traffics carried on the various routes in these systems. The results shown are the lost traffics from the network. The offered traffic intensities from the origin node i to the destination node j is given in the column headed "a_{ij}". For each of these offered streams the total traffic lost as calculated by the procedure is given in the column headed "procedure" and ninety five percent confidence intervals for the results calculated using simulation are given in the "simulation" column. The total traffic offered to the network, the total traffic which is lost from the network calculated using the procedure and the total lost traffic calculated by simulation are all given in the final row of the table. At the end of each table the average congestion and the number of iterations required to get these results are displayed. For each particular example of the four node network, results will be given, first of all for the case when crankback is used and then for the case when crankback is not used.

A few things can be seen by looking at these results. The first thing to notice is that in most cases the total lost call intensity for the procedure lies within the confidence intervals for the total lost call intensity obtained by simulation. The individual traffic loss intensities, however, show a different picture. In table (4.1) only one of the procedure's results lie within the corresponding lost call intensity confidence interval. In this example the streams with a high number of lost calls appear to be underestimated and those with a low lost call intensity appear to be overestimated, thus the overall congestion appears to be close to the correct result.

Comparing table (4.1) and table (4.2) we find that crankback reduces the overall network congestion as we would expect. Unexpectedly, perhaps, table (4.3) and table (4.4) indicate the opposite. The overall network performance decreases slightly when crankback is introduced. A factor in these results is that the overall network congestion for the case shown in table (4.1) and table (4.2) is considerably lower than for the case shown in table (4.3) and table (4.4). Crankback is an efficient way of reducing network congestion when this congestion is low. A call cranked back is likely to find an alternate route under a low network congestion situation. If the network congestion is high an alternate route is less likely to be found so any benefit from crankback will be lessened. Moreover, in a heavy traffic situation overflow calls using many links are more likely to get in the way of direct calls on single links and so substantially decrease network performance. In fact there may be no benefit from using crankback and possibly even a deterioration in the network performance. Tables (4.5) to (4.8) show results for the network when the circuit allocation vector is kept constant. The total arrival rate for the case shown in tables (4.5) and (4.6) is higher than that shown in tables (4.7) and (4.8). In both cases there is low congestion and so crankback does lower the overall network congestion. However, when the overall congestion is higher the benefit that crankback has on network performance diminishes, both in terms of the total traffic lost and percentage of traffic lost. These results are tied in with the results shown in tables (4.1) to (4.4).

In tables (4.9) and (4.10) some results are shown for the case when some circuits are reserved for direct routed calls. We have reserved about ten percent of the circuits on eack link in this network. Tables (4.3) and (4.4) show the results for this case when there is no circuit reservation. Comparing these results we find that the circuit reservation does in fact reduce the congestion on the network for this case. The overall congestion on the network in this case is quite high. When the congestion is not as high a reservation scheme will not reduce the congestion as much. Notice also that the results using the procedure are slightly higher than those from the simulation. This is because the reservation scheme used by the simulation is slightly different from that used by the procedure as was mentioned in chapter 3. The procedure's reservation scheme comes into effect less often than the simulation's and hence does not reduce the congestion as much.

As well as reducing overall congestion we find that by including a circuit reservation scheme, introducing crankback now reduces the network congestion. Without circuit reservation crankback did not reduce the overall network congestion when this congestion was high. Our results seem to indicate that crankback will reduce network congestion when this congestion is both high and low if a circuit reservation scheme is introduced. This is because under high congestion conditions it is not desirable that calls will use alternate multi-link chains whereas under low congestion conditions it is desirable that alternate chains are used. A combination of crankback and circuit reservation will ensure the preferred network behaviour under both high and low congestion situations. These results can be seen by looking at both the simulation results and those obtained by using our algorithm.

4.3 Final Conclusions.

It has been shown in this part of the thesis how a method for determining end-to-end congestions in a non-hierarchical telecommunications network was created. The reasoning behind some of the ideas used to create an accurate method has been shown. The aim was to make the method as accurate as possible while preserving speed of execution. As a result problems arose which, as yet, are not completely resolved. However, the method is fairly stable and fast and gives reasonable results under both high and low congestions for the networks analysed. When much larger networks are used errors can creep in due to some numerical problems. These are tied in with some of the heuristic procedures that are used in the algorithm. The algorithm tends to give the correct result for the overall congestion but not for the individual arrival stream congestions. This would appear to indicate that our methods for determining the traffics lost on the chains work reasonably well but our splitting formulae do not.

Unless some of these problems can be solved, our algorithm can not be expected to give more accurate results than those presented in this chapter. These problems are fairly difficult and no solution so far attempted has been entirely satisfactory. It remains to be seen if any satisfactory solution can be discovered in the future.

Results for the 4 node network when the circuit allocation vector used is N=(9,11,20,8,5,13,15,4,51,29,13,34).

i	j	a_{ij}	procedure	simulation
1	2	5.0	0.150	(0.139,0.151)
1	3	15.0	0.294	(0.299,0.329)
1	4	7.0	0.172	(0.175,0.197)
2	1	4.0	0.115	(0.096,0.103)
2	3	8.0	0.161	(0.139,0.153)
2	4	2.0	0.093	(0.049,0.053)
3	1	23.0	0.299	(0.343,0.366)
3	2	8.0	0.154	(0.124,0.131)
3	4	28.0	0.403	(0.417,0.502)
4	1	10.0	0.262	(0.221,0.237)
4	2	1.0	0.063	(0.022,0.029)
4	3	45.0	0.704	(0.774,0.909)
		156.0	2.871	(2.798,3.162)

Table (4.1) Crankback The number of iterations taken was 6 and the average calculated network congestion was 1.8 percent.

i	j	a_{ij}	procedure	simulation
1	2	5.0	0.175	(0.193,0.202)
1	3	15.0	0.330	(0.351,0.371)
1	4	7.0	0.197	(0.232,0.263)
2	1	4.0	0.119	(0.112,0.132)
2	3	8.0	0.167	(0.147,0.165)
2	4	2.0	0.087	(0.063,0.070)
3	1	23.0	0.410	(0.478,0.546)
3	2	8.0	0.157	(0.126,0.151)
3	4	28.0	0.445	(0.457,0.490)
4	1	10.0	0.289	(0.263,0.292)
4	2	1.0	0.062	(0.037,0.042)
4	3	45.0	0.790	(0.815,0.868)
		156.0	3.228	(3.274,3.592)

Table (4.2) No Crankback The number of iterations taken was 3 and the total network congestion was 2.1 percent.

Results for the 4 node network when the circuit allocation vector used is N=(8,10,18,7,4,11,13,3,46,26,11,30).

i	j	a_{ij}	procedure	simulation
1	2	5.0	0.611	(0.602,0.648)
1	3	15.0	1.100	(1.181,1.281)
1	4	7.0	0.690	(0.723,0.787)
2	1	4.0	0.512	(0.478,0.511)
2	3	8.0	1.019	(0.764,0.796)
2	4	2.0	0.428	(0.251,0.272)
3	1	23.0	1.556	(1.728,1.769)
3	2	8.0	1.059	(0.741,0.758)
3	4	28.0	2.689	(1.990,2.149)
4	1	10.0	1.106	(1.033,1.080)
4	2	1.0	0.258	(0.134,0.144)
4	3	45.0	2.461	(3.480,3.580)
		156.0	13.498	(13.105,13.775)

Table (4.3) Crankback The number of iterations taken was 4 and the total network congestion was 8.6 percent.

i	j	a_{ij}	procedure	simulation
1	2	5.0	0.682	(0.675,0.744)
1	3	15.0	1.095	(1.189,1.250)
1	4	7.0	0.746	(0.790,0.859)
2	1	4.0	0.535	(0.505,0.556)
2	3	8.0	0.991	(0.683,0.735)
2	4	2.0	0.389	(0.248,0.275)
3	1	23.0	1.655	(1.897,2.001)
3	2	8.0	0.988	(0.659,0.708)
3	4	28.0	2.591	(1.599,1.793)
4	1	10.0	1.044	(0.972,1.114)
4	2	1.0	0.255	(0.142,0.174)
4	3	45.0	2.427	(2.857,3.281)
		156.0	13.398	(12.216,13.492)

Table (4.4) No Crankback The number of iterations taken was 4 and the total network congestion was 8.6 percent.

Results for the 4 node network when the circuit allocation vector used is N=(14,17,20,12,10,13,15,9,31,19,13,24).

i	j	a_{ij}	procedure	simulation
1	2	9.0	0.343	(0.356,0.393)
1	3	16.0	0.315	(0.293,0.373)
1	4	12.0	0.505	(0.420,0.502)
2	1	7.0	0.485	(0.439,0.509)
2	3	10.0	0.403	(0.324,0.386)
2	4	7.0	0.438	(0.377,0.436)
3	1	18.0	1.506	(1.034,1.126)
3	2	8.0	0.512	(0.384,0.458)
3	4	20.0	0.863	(0.885,0.966)
4	1	10.0	0.345	(0.344,0.397)
4	2	7.0	0.179	(0.194,0.246)
4	3	18.0	0.036	(0.039,0.063)
		142.0	5.930	(5.089,5.855)

Table (4.5) Crankback The number of iterations taken was 4 and the total network congestion was 4.2 percent.

i	j	a_{ij}	procedure	simulation
1	2	9.0	0.382	(0.445,0.526)
1	3	16.0	0.526	(0.479,0.609)
1	4	12.0	0.526	(0.474,0.579)
2	1	7.0	0.522	(0.410,0.467)
2	3	10.0	0.454	(0.396,0.463)
2	4	7.0	0.440	(0.368,0.415)
3	1	18.0	1.629	(1.233,1.336)
3	2	8.0	0.417	(0.292,0.351)
3	4	20.0	0.739	(0.655,0.750)
4	1	10.0	0.391	(0.335,0.394)
4	2	7.0	0.314	(0.346,0.404)
4	3	18.0	0.023	(0.020,0.037)
		142.0	6.363	(5.453,6.331)

Table (4.6) No Crankback The number of iterations taken was 6 and the total network congestion was 4.5 percent.

Results for the 4 node network when the circuit allocation vector used is N=(14,17,20,12,10,13,15,9,31,19,13,24).

i	j	a_{ij}	procedure	simulation
1	2	9.0	0.088	(0.123,0.156)
1	3	11.0	0.003	(0.001,0.005)
1	4	8.0	0.013	(0.008,0.022)
2	1	7.0	0.208	(0.174,0.218)
2	3	8.0	0.030	(0.013,0.033)
2	4	5.0	0.046	(0.031,0.049)
3	1	18.0	1.410	(0.846,1.006)
3	2	8.0	0.439	(0.358,0.395)
3	4	20.0	0.341	(0.443,0.543)
4	1	10.0	0.189	(0.286,0.356)
4	2	7.0	0.160	(0.182,0.205)
4	3	15.0	0.004	(0.001,0.005)
		126.0	2.931	(2.466,2.993)

Table (4.7) Crankback The number of iterations taken was 4 and the total network congestion was 2.3 percent.

i	j	a_{ij}	procedure	simulation
1	2	9.0	0.224	(0.260,0.324)
1	3	11.0	0.007	(0.006,0.012)
1	4	8.0	0.017	(0.015,0.024)
2	1	7.0	0.276	(0.236,0.271)
2	3	8.0	0.031	(0.018,0.029)
2	4	5.0	0.045	(0.030,0.042)
3	1	18.0	1.567	(1.032,1.115)
3	2	8.0	0.412	(0.277,0.335)
3	4	20.0	0.314	(0.343,0.419)
4	1	10.0	0.383	(0.313,0.353)
4	2	7.0	0.244	(0.247,0.281)
4	3	15.0	0.003	(0.001,0.003)
		126.0	3.523	(2.778,3.208)

Table (4.8) No crankback The number of iterations taken was 5 and the total network congestion was 2.8 percent.

Results for the 4 node network when the circuit allocation vector used is N=(8,10,18,7,4,11,13,3,46,26,11,30) and the number of reserved circuits are given by the vector NR=(1,1,2,1,1,1,1,1,5,3,1,3).

i	j	a_{ij}	procedure	simulation
1	2	5.0	0.372	(0.319,0.361)
1	3	15.0	1.292	(1.225,1.281)
1	4	7.0	0.488	(0.478,0.501)
2	1	4.0	0.297	(0.253,0.289)
2	3	8.0	0.603	(0.656,0.692)
2	4	2.0	0.231	(0.148,0.156)
3	1	23.0	1.545	(1.292,1.329)
3	2	8.0	0.664	(0.638,0.694)
3	4	28.0	2.144	(1.887,2.028)
4	1	10.0	0.884	(1.060,1.114)
4	2	1.0	0.112	(0.062,0.075)
4	3	45.0	3.420	(3.115,3.236)
		156.0	11.992	(11.365,11.220)

Table (4.9) Crankback The number of iterations taken was 4 and the total network congestion was 7.3 percent.

i	j	a_{ij}	procedure	simulation
1	2	5.0	0.381	(0.365,0.391)
1	3	15.0	1.278	(1.175,1.236)
1	4	7.0	0.482	(0.420,0.455)
2	1	4.0	0.303	(0.230,0.268)
2	3	8.0	0.621	(0.662,0.671)
2	4	2.0	0.236	(0.182,0.189)
3	1	23.0	1.440	(1.302,1.352)
3	2	8.0	0.651	(0.626,0.700)
3	4	28.0	2.246	(2.181,2.325)
4	1	10.0	1.135	(1.058,1.126)
4	2	1.0	0.089	(0.060,0.074)
4	3	45.0	3.452	(3.226,3.331)
		156.0	12.096	(11.705,11.901)

Table (4.10) No Crankback The number of iterations taken was 4 and the total network congestion was 7.6 percent.

CHAPTER 5

INTRODUCTION TO SENSITIVITY BOUNDS.

In this second part of the thesis, a way of determining sensitivity bounds for certain performance measures of some simple Generalised Semi-Markov Processes (GSMPs) will be presented. This method shall be applied to a number of examples of GSMPs and some interesting numeric and analytic results will be obtained. One of these results will be an analytic proof of an upper bound for the time congestion in the $GI/M/n/n$ loss system.

Roughly speaking, a GSMP is a process whose states comprise sets of lifetimes which may be negative exponentially distributed or generally distributed. If all the lifetimes are negative exponentially distributed we would, of course, just have a Markov Process. Whenever a lifetime ends, be it from a negative exponential distribution or a general distribution, other lifetimes may be started depending on the state of the process at that time. Only one lifetime may end at any one time thus restricting the set of states that the process can move to when a lifetime ends. Many queueing systems can be represented within the framework of a GSMP.

A GSMP is insensitive if its steady state distribution depends only on the mean of the generally distributed lifetimes involved. Thus, in an insensitive GSMP the steady state distribution remains unchanged when a general distribution is allowed to range over a set of distributions with a fixed mean. There is an extensive literature which discusses criteria for a GSMP or other similar structures to be insensitive (see, for example, König and Jansen (1974), Schassberger (1977), (1978a) and (1978b), Kelly (1976), Franken et al. (1982), Burman (1981), Henderson (1983), Whittle (1985) and (1986), and Taylor (1988)).

There are, perhaps surprisingly, many GSMPs which turn out to be insensitive, including the Erlang loss system, the infinite server queue and many queueing network models. (See Baskett et al (1975), Chandy, Howard and Towsley (1977), Kelly (1979), Hordijk and van Dijk (1983a) and (1983b), and Chandy and Martin (1983)). However there are also many simple GSMPs which are not insensitive, including the $GI/M/n/n$ loss system. This is so since the partial balance criterion for a GSMP cannot be satisfied when an arrival process is generally distributed. If a GSMP is not insensitive then the value of the steady state probabilities of the system do depend on the actual form that the generally distributed lifetimes take, not only on their mean. So, in a non insensitive GSMP there must exist at least two different lifetime distributions with a fixed mean value which give rise to different steady state probability distributions. The supremum and infimum of a performance measure as the lifetime distribution range over the set of distributions with fixed mean form the sensitivity bounds for that performance measure.

Briefly, the question which we will try to answer in this part of the thesis is : "Given a GSMP with only one generally distributed lifetime, how much do the steady state probabilities vary as the distribution of the general lifetime varies over the set of distributions with a given mean?"

The problem of bounding various performance measures in queueing systems is an important practical problem. Finding an upper bound on measures such as the congestion in a loss system or the queue length in a queueing system will help in designing networks of queues. Two methods for obtaining bounds for certain performance measures in queues with generally distributed lifetimes will be mentioned. Some of the techniques used by these methods will be used later in this thesis.

The first of these methods has been studied recently by van Dijk (1987), van Dijk, Walrand and Tsoucas (1987), Hordijk and Ridder (1988) and Ridder (1987). Their methods involve the comparison of queues with other queues which are known to be insensitive, thus providing maximum and minimum values for performance measures associated with the insensitive queues. An example of a queue that these methods can be applied to is the $M/G/c/n$ queueing system, that is a queue with c servers and $n-c$ waiting places, Poisson arrivals and a common generally distributed service time. Van Dijk, Walrand and Tsoucas (1987) have proved that the call congestion in the $M/G/c/n$ queue is bounded above by the call congestion of the $M/G/c/c$ queue and bounded below by the call congestion of the $M/G/n/n$ queue. Since the $M/G/j/j$ queue is just the Erlang system with j servers, which is known to be insensitive, these two bounds can be easily calculated using the Erlang loss formula. This type of approach when used to find bounds on performance measures in certain queues can yield some very quick and useful results.

Another method of bounding certain performance measures in queues uses the Laplace-Stieltjes transforms of the generally distributed lifetimes probability distribution function. Several performance measures in queueing systems can be expressed in terms of these Laplace-Stieltjes transforms. Bounds on such performance measures can be found by finding bounds for the Laplace-Stieltjes transforms of the probability distribution functions. Establishing such bounds is achieved by using the theory of complete Tchebycheff systems. (See Kreĭn and Nudel'man (1977)). Examples of these types of results can be found in Eckberg (1977) and Whitt (1984).

In Eckberg (1977), constraints on the Laplace-Stieltjes transforms of a probability distribution function are obtained when various characteristics of this distribution are stipulated. These constraints are then used to find bounds on various performance measures of queues if these performance measures can be found in terms of the Laplace-Stieltjes transforms. A simple example of a queueing system for which this technique is used is the $GI/M/1/1$ queueing system. For this system the call congestion is given by $\phi(1/a)$ where a is the mean call arrival intensity and $\phi(s)$ is the Laplace-Stieltjes transform of the call arrival distribution function. The maximum and minimum possible values of the Laplace-Stieltjes transform of the distribution function will directly give bounds on the call congestion in this queue. These results can be extended by specifying more characteristics of the call arrival distribution function. For example, the variance of the distribution function may be specified, further parameter constraints may also be added and the bound corresponding to a distribution function with these characteristics can be found. The addition of more parameters, however, can make the form of these bounds very messy.

In Whitt (1984), methods using complete Tchebycheff systems have been used to investigate how much information about a distribution is contained in the moments of that distribution. Whitt has focused on the mean queue length of the $GI/M/1$ queue. The mean queue length of the $GI/M/1$ queue is given by $\rho/(1-\sigma)$ where σ satisfies the equation $\sigma = \phi[\mu(1-\sigma)]$ with ρ the mean service rate, μ the mean inter-call arrival time and $\phi(s)$ the Laplace-Stieltjes transform of the call inter-arrival distribution. The maximum mean queue length occurs when σ is maximised, this occurs when an upper bound on $\phi(s)$ is used. Bounds on the mean queue length of the $GI/M/1$ queue are found given the first two moments of the inter-arrival time distribution and the mean service rates. These results show the range of possible values that the mean queue length may take if a two moment model is used to approximate a queueing system. The distributions that achieved these bounds are unusual in that they are discrete probability distributions with positive probability at just two points. In Klincewicz and Whitt (1984), sharper bounds on the mean queue length were obtained by introducing further constraints on the probability distribution function. These constraints were introduced so that no distribution function with peculiar properties would be considered, the bounds so obtained would be expected to have more practical importance. The motivation behind Whitt's work was the investigation of the congestion in non-Markovian networks. If any stream offered to a node is represented as a renewal stream characterised by its first two moments then bounds on the congestion at this node can be found by applying the techniques briefly described above.

In Chapter 6 of this thesis a method for finding sensitivity bounds on certain performance measures of GSMPs with a single generally distributed lifetime will be introduced. This method will involve solving a non-linear optimisation problem where the variables are the Laplace-Stieltjes transforms of the general lifetime distribution at certain points. Constraints on these variables will be obtained by using the fact that a probability distribution is completely monotone. This method will be applied to certain simple queueing systems and some interesting numerical and analytic results will be obtained.

In Chapter 7 bounds on the time congestion in the $GI/M/n/n$ queueing system will be presented. A proof of this result for the case when $n = 2$ will be given. A general proof of this result is long and complicated and appears in the appendix. These sensitivity bounds are then discussed.

In Chapter 8, the conclusion, the results presented in this part of the thesis will be summarised. There are many ways that this method of determining bounds in a GSMP can be extended and improved which are beyond the scope of this thesis. These possible extensions will be mentioned briefly and hopefully can be investigated in the near future.

CHAPTER 6

DETERMINING SENSITIVITY BOUNDS

6.1 The Model

The model described in this chapter is due to Whittle (1985), with slight modifications as made in Taylor (1988). Schassberger (1985) showed that the standard GSMP framework is essentially equivalent to Whittle's model. Whittle's model will be used since we find it a simpler system than the standard GSMP model.

Take an irreducible Markov process $\mathcal{M}$ on a finite set $\mathcal{H}$ and let A be a subset of $\mathcal{H}$. Let the process have transition rates $q(x, x')$ from state x to state x'. If x and x' are both in A, transitions between them can be of two types, those which can be regarded as leaving A and immediately returning, and those which occur within A, with rates $q^E(x, x')$ and $q^I(x, x')$ respectively. Thus the total transition rate between two such states is $q(x, x') = q^E(x, x') + q^I(x, x')$.

The sojourn time in A can be considered to be a random variable with unit mean. When the current state is $x \in A$ this random variable is worked off at rate $c(x, A) = \sum_{x' \in A} q^E(x, x') + \sum_{x' \in A^c} q(x, x')$ where $A^c = \mathcal{H} \backslash A$. Processes where $c(x, A)$ can be zero for some states x can be treated by a suitable modification to our method (see Taylor (1988)) but, for the sake of simplicity, we assume that $c(x, A) > 0$.

The process $\mathcal{M}$ can be modified so that it has an arbitrarily distributed sojourn time in A according to the following rules:

(1) On jumping into A (either from A^c or by an external transition) the process is assigned a nominal sojourn time from a general distribution $G(.)$ with unit

mean. When the process is in a particular state $x \in A$ this sojourn time is worked off at rate $c(x, A)$.

(2) Until the nominal sojourn time is worked off transitions occur between states x and x' in A at rate $q^I(x, x')$.

(3) When the sojourn time is complete the process must immediately jump out of A and it is then assigned to x' with probability $q(x, x')/c(x, A)$ for $x' \in A^c$ or $q^E(x, x')/c(x, A)$ for $x' \in A$. If the process is transferred to x' in A a new sojourn time is selected.

Kuczura (1973) introduced the concept of a piecewise Markov process and used it to model queues with general arrival streams. Because it has only one generally distributed lifetime the model described above is similar to a piecewise Markov process, but covers more cases because the general lifetime may not always be alive and, even if it is alive, can be worked off at a speed dependent on the current state of the process.

In the following analysis we use the method of supplementary variables to analyse the process $\mathcal{M}$, adding a supplementary variable for spent sojourn time in A. Miyazawa (1988) has discussed conditions under which the Kolmogorov differential equations ((6.3) below) characterise the stationary distribution of such a process supplemented with a variable denoting residual sojourn time. However, because we use spent sojourn time we make the *a priori* assumption that a unique supplemented stationary distribution exists, so that it must satisfy the appropriate Kolmogorov differential equations.

We will need to introduce a partitioning of the transition probability matrix for the Markov process $\mathcal{M}$. Hence we write $Q = [q(x, x')]$ as

$$\begin{pmatrix} Q^E + Q^I - C & Q_{AA^c} \\ Q_{A^cA} & Q_{A^cA^c} \end{pmatrix}$$

where

$$[Q..]_{x,x'} = \begin{cases} q(x, x') & x \neq x' \\ -\sum_{z \in \mathcal{H}} q(x, z) & x = x' \end{cases} \quad (x, x' \text{ not both in } A),$$

$$[Q^I]_{x,x'} = \begin{cases} q^I(x, x') & x \neq x' \\ -\sum_{z \in A} q^I(x, z) & x = x' \end{cases} \quad (x, x' \in A),$$

$$[Q^E]_{x,x'} = q^E(x, x') \qquad (x, x' \in A),$$

$$[C]_{x,x'} = \begin{cases} 0 & x \neq x' \\ c(x, A) & x = x' \end{cases} \qquad (x, x' \in A)$$

and we define

$$Q_1 = Q^I \tag{6.1}$$

and

$$Q_2 = -C + Q^E - \left[Q_{AA^c}(Q_{A^cA^c})^{-1}Q_{A^cA}\right]. \tag{6.2}$$

The invertibility of $Q_{A^cA^c}$ follows from the irreducibility of $\mathcal{M}$ (see Taylor 1987).

Let $p_x(y, G)$ be the stationary probability density that the process is in state x with a spent lifetime y for $x \in A$ when the general lifetime is distributed according to $G(.)$. Define $\mathbf{P}(y, G) = (p_x(y, G), x \in A)$ as the vector containing these densities and $P(G)_{A^c} = (p_x, x \in A^c)$ as the vector containing the stationary probabilities that the process is in state x for $x \in A^c$. Then it is shown in Taylor (1987) that

$\mathbf{P}(y, G)$ and $P(G)_{A^c}$ satisfy

$$P(G)_{A^c} Q_{A^c A^c} + \int_0^\infty \mathbf{P}(y, G) h(y) dy Q_{A A^c} = \mathbf{0}$$

$$\left[h(y)\mathbf{P}(y, G) + \frac{d}{dy}\mathbf{P}(y, G) \right] C - \mathbf{P}(y, G) Q^I = \mathbf{0} \tag{6.3}$$

$$P(G)_{A^c} Q_{A^c A} + \int_0^\infty \mathbf{P}(y, G) h(y) dy Q^E = \mathbf{P}(0, G) C$$

where $h(y)$ is the hazard function associated with the distribution $G(.)$.

By some elementary manipulations on equations (6.3) it is easy to show that they can be rewritten as

$$\left[h(y)\mathbf{P}(y, G) + \frac{d}{dy}\mathbf{P}(y, G) \right] C = \mathbf{P}(y, G) Q_1 \tag{6.4a}$$

$$\int_0^\infty \mathbf{P}(y, G) dy Q_1 + \int_0^\infty \mathbf{P}(y, G) h(y) dy Q_2 = \mathbf{0} \tag{6.4b}$$

$$-\int_0^\infty \mathbf{P}(y, G) h(y) dy Q_{A A^c} (Q_{A^c A^c})^{-1} = P(G)_{A^c} \tag{6.4c}$$

where Q_1 and Q_2 are given by (6.1) and (6.2).

It is shown in Taylor (1987) that zero is an eigenvalue of $Q_1 C^{-1}$ and that all other eigenvalues have negative real parts. In the argument below we assume, for simplicity, that $Q_1 C^{-1}$ has n distinct real eigenvalues $0 = -\alpha_0 > -\alpha_1 > \ldots > -\alpha_{n-1}$ with corresponding left eigenvectors $\mathbf{w}_i$. If this is not the case the argument can be suitably modified. An example of a queueing system when the eigenvalues are not distinct is analysed later in this chapter.

The differential equation in (6.4a) has solution

$$\mathbf{P}(y, G) = \sum_{i=0}^{n-1} A_i^{(G)} e^{-\alpha_i y} \left[1 - G(y)\right] \mathbf{w}_i \tag{6.5}$$

where the constants $A_i^{(G)}$ depend on $G(.)$ because (6.4b) depends on $G(.)$. From (6.5) we get

$$\int_0^\infty \mathbf{P}(y, G) dy = A_0^{(G)} \mathbf{w}_0 + \sum_{i=1}^{n-1} \frac{A_i^{(G)}}{\alpha_i} \left[1 - \widehat{G}(\alpha_i)\right] \mathbf{w}_i \tag{6.6}$$

and

$$\int_0^\infty \mathbf{P}(y,G)h(y)dy = A_0^{(G)}\mathbf{w}_0 + \sum_{i=1}^{n-1} A_i^{(G)}\widehat{G}(\alpha_i)\mathbf{w}_i \tag{6.7}$$

where $\widehat{G}(\alpha_i) = \int_0^\infty e^{-\alpha_i y}dG(y)$, the Laplace-Stieltjes transform of the generally distributed lifetime's probability distribution function at α_i. Substituting (6.6) and (6.7) into (6.4b) gives

$$\left[A_0^{(G)}\mathbf{w}_0 + \sum_{i=1}^{n-1}\frac{A_i^{(G)}}{\alpha_i}\left[1-\widehat{G}(\alpha_i)\right]\mathbf{w}_i\right]Q_1 + \left[A_0^{(G)}\mathbf{w}_0 + \sum_{i=1}^{n-1}A_i^{(G)}\widehat{G}(\alpha_i)\mathbf{w}_i\right]Q_2 = \mathbf{0}. \tag{6.8}$$

The constants $A_i^{(G)}$ are determined up to a constant multiple by (6.8). This constant multiple can be evaluated by substituting (6.6) and (6.7) into the normalising condition

$$\int_0^\infty \mathbf{P}(y,G)dy\mathbf{e} - \int_0^\infty \mathbf{P}(y,G)h(y)dyQ_{AA^c}(Q_{A^cA^c})^{-1}\mathbf{e} = 1$$

(where $\mathbf{e}$ represents a column of ones of appropriate dimension) to get

$$\begin{aligned}&\left[A_0^{(G)}\mathbf{w}_0 + \sum_{i=1}^{n-1}\frac{A_i^{(G)}}{\alpha_i}\left[1-\widehat{G}(\alpha_i)\right]\mathbf{w}_i\right]\mathbf{e}\\ &\qquad-\left[A_0^{(G)}\mathbf{w}_0 + \sum_{i=1}^{n-1}A_i^{(G)}\widehat{G}(\alpha_i)\mathbf{w}_i\right]Q_{AA^c}(Q_{A^cA^c})^{-1}\mathbf{e} = 1.\end{aligned} \tag{6.9}$$

For a process with $C = I$ the product $\mathbf{w}_i\mathbf{e} = 0$, $\forall i = 1,\ldots,n-1$ and so (6.9) can be reduced to

$$A_0^{(G)}\mathbf{w}_0\mathbf{e} - \left[A_0^{(G)}\mathbf{w}_0 + \sum_{i=1}^{n-1}A_i^{(G)}\widehat{G}(\alpha_i)\right]Q_{AA^c}(Q_{A^cA^c})^{-1}\mathbf{e} = 1. \tag{6.10}$$

Thus the stationary distribution of the process is determined by equations (6.8) together with (6.9) or (6.10).

6.2. A Method for Deriving Bounds on Sensitivity.

Equations (6.8) and (6.9) form a system of non-linear equations that must be satisfied by the variables $A_i^{(G)}$'s and $\widehat{G}(\alpha_i)$'s in order that these variables correspond to a GSMP, with Q matrix as described in the above section. These constraints, together with some extra ones derived below, will form the constraints of a non-linear optimisation problem. The objective function of this problem is a performance measure of the GSMP, given by a combination of the elements of the vector $\mathbf{P}(y, G)$. By solving this non-linear optimisation problem, bounds on a performance measure of this GSMP will be established.

In most cases the precise values of the $\widehat{G}(\alpha_i)$ will not be known. In fact in many cases the only characteristic of the general distribution that is known will be its mean, and so it is of interest to determine the maximum variation that is possible assuming that the mean of $G(.)$ is fixed. As in section 6.1 we can, without loss of generality, take $G(.)$ to have unit mean and incorporate any variation into the speeds at which the lifetime is worked off.

Let $\mathcal{G} = \{G(.) : \int_0^\infty y dG(y) = 1\}$ and $H(G) \equiv F(\int_0^\infty \mathbf{P}(y, G)dy, P(G)_{A^c})$ be a bounded function of the stationary distribution of the process when the general lifetime is distributed according to $G(.) \in \mathcal{G}$. Then we wish to find the $\sup_{G \in \mathcal{G}} H(G)$ and $\inf_{G \in \mathcal{G}} H(G)$.

If $H(G)$ is taken to be $[\int_0^\infty \mathbf{P}(y, G)dy]_j$ then optimising the value of $H(G)$ is equivalent to finding the maximum and minimum possible values of the stationary probability that the process is in state j. In practical situations we may be interested in this for some particular j to obtain, for example, bounds on the loss probability of a queue or on the probability of emptiness of the system. However, more complicated functions $H(G)$ are possible.

Our approach will be to treat the constants $A_i^{(G)}$ and the values of $X_i \equiv \widehat{G}(\alpha_i)$ as variables in a non-linear constrained optimisation problem. It follows from Jensen's inequality that the X_i have to satisfy the inequality constraints

$$e^{-\alpha_i} \leq X_i \quad i = 1, ..., n \tag{6.11}$$

and from the fact that $X_0 = 1$ and $\widehat{G}'(\alpha) < 0 \; \forall \; \alpha > 0$ (see Feller (1966), p. 415) that

$$X_i < 1 \quad i = 1, ..., n \tag{6.12}$$

and the X_i's and the A_i's satisfy the equality constraints (6.8) and (6.9). In fact it is shown in Pearce (1978) that equality holds in (6.11) only if $G(.)$ is deterministic with unit mean. This result can also be obtained by using the method of complete Tchebycheff systems. It is also possible to make $\widehat{G}(\alpha)$ arbitrarily close to 1, still preserving the unit mean of $G(.)$ by defining $G(.)$ by

$$G(.) = \begin{cases} 1 - \epsilon & 0 \leq y \leq \frac{1}{\epsilon} \\ 1 & y > \frac{1}{\epsilon} \end{cases} \tag{6.13}$$

for sufficiently small ϵ.

If we were to try to formulate a non-linear program based on what has been presented so far we would find the results obtained of little practical use. This is because we have not introduced any constraints on the variables X_i to ensure that they at least resemble values of Laplace-Stieltjes transforms we would expect to get from a probability distribution. These constraints would set bounds on the value of Laplace-Stieltjes transform, X_m, in terms of the values of the previous Laplace-Stieltjes transforms, $X_i, i < m$. The use of complete Tchebycheff systems would enable us to get exact bounds on X_m given that $G(.)$ has unit mean and that X_i is fixed for $i < m$. However, the constraints that are given by the use of complete Tchebycheff systems can become very complicated and necessitates the introduction of extra variables.

One way to get reasonable constraints in some problems which are not too complicated is to use the fact that $\widehat{G}(\alpha)$ is completely monotone (see Feller (1966)). If we know the value of $\widehat{G}(\beta)$ for some $0 < \beta < \alpha$ we can obtain tighter bounds on $\widehat{G}(\alpha)$ than given in (6.11) and (6.12). This follows from the following lemma.

Lemma 6.1

Let $0 < \alpha_1 < \alpha_2 < \alpha_3$ and suppose that $X_1 = \widehat{G}(\alpha_1)$ and $X_2 = \widehat{G}(\alpha_2)$. Then

$$\frac{X_2(\alpha_3 - \alpha_1) + X_1(\alpha_2 - \alpha_3)}{\alpha_2 - \alpha_1} < \widehat{G}(\alpha_3) < X_2. \tag{6.14}$$

Proof

The right hand inequality is an easy consequence of the fact that $\widehat{G}(\alpha)$ is decreasing. For the left hand ineqality we use the fact that $\widehat{G}(\alpha)$ is completely monotone (see Feller (1966)). In particular $\widehat{G}''(\alpha) > 0$ for $\alpha > 0$. Thus

$$\widehat{G}(\alpha_2) = \widehat{G}\left(\frac{\alpha_3 - \alpha_2}{\alpha_3 - \alpha_1}\alpha_1 + \frac{\alpha_2 - \alpha_1}{\alpha_3 - \alpha_1}\alpha_3\right)$$

$$< \frac{\alpha_3 - \alpha_2}{\alpha_3 - \alpha_1}\widehat{G}(\alpha_1) + \frac{\alpha_2 - \alpha_1}{\alpha_3 - \alpha_1}\widehat{G}(\alpha_3)$$

$$\Rightarrow \widehat{G}(\alpha_3) > \frac{\alpha_3 - \alpha_1}{\alpha_2 - \alpha_1}X_2 + \frac{\alpha_2 - \alpha_3}{\alpha_2 - \alpha_1}X_1.$$

□

Theorem 6.1

The supremum of $H(G)$ as $G(.)$ ranges over $\mathcal{G}$ is less than or equal to the solution of the constrained optimisation problem $\mathcal{P}^*$, in the variables A_i, $i =$

$0,\ldots,n-1$ and $X_i,\ i=1,\ldots,n-1$ defined by

$$\max F\left(A_0\mathbf{w}_0+\sum_{i=1}^{n-1}\frac{A_i}{\alpha_i}\left[1-X_i\right]\mathbf{w}_i,\left[A_0\mathbf{w}_0+\sum_{i=1}^{n-1}A_iX_i\mathbf{w}_i\right]Q_{AA^c}(Q_{A^cA^c})^{-1}\right) \tag{6.16}$$

subject to the constraints

$$e^{-\alpha_i}\leq X_i\leq 1,\ i=1,...,n-1, \tag{6.17a}$$

$$\frac{X_1\alpha_2+\alpha_1-\alpha_2}{\alpha_1}\leq X_2\leq X_1, \tag{6.17b}$$

$$\frac{X_i(\alpha_{i+1}-\alpha_{i-1})+X_{i-1}(\alpha_i-\alpha_{i+1})}{\alpha_i-\alpha_{i-1}}\leq X_{i+1}\leq X_i \text{ for } 1<i<n-2, \tag{6.17c}$$

$$\left[A_0\mathbf{w}_0+\sum_{i=1}^{n-1}\frac{A_i}{\alpha_i}\left[1-X_i\right]\mathbf{w}_i\right]Q_1+\left[A_0\mathbf{w}_0+\sum_{i=1}^{n-1}A_iX_i\mathbf{w}_i\right]Q_2=\mathbf{0} \tag{6.18}$$

and

$$\left[A_0\mathbf{w}_0+\sum_{i=1}^{n-1}\frac{A_i}{\alpha_i}\left[1-X_i\right]\mathbf{w}_i\right]\mathbf{e}-\left[A_0\mathbf{w}_0+\sum_{i=1}^{n-1}A_iX_i\mathbf{w}_i\right]Q_{AA^c}(Q_{A^cA^c})^{-1}\mathbf{e}=1. \tag{6.19}$$

Proof

By Lemma 6.1 the values of $\widehat{G}(\alpha_i)$ as G ranges over $\mathcal{G}$ must satisfy the inequalities (6.17) and by the discussion in this chapter the constants A_i must satisfy the equality constraints (6.18) and (6.19). Thus $H(G)$ is less than or equal to the solution of $\mathcal{P}^*$. □

6.3. Some simple examples.

The non-linear optimisation problem given in Theorem 6.1 can be solved analytically for some simple problems. However, numerical results are all that can be expected in large, complicated systems. A computer program to obtain results for the above problem has been written and results obtained. The program, written in Fortran and using NAG routines is straightforward. The only point to note is that convergence to a feasible solution is guaranteed when the initial point is chosen to be feasible. This point can be found by setting the X_i's equal to some values which satisfy (6.17) (for example $X_i = 1.0$ for all i or $X_i = e^{-\alpha_i}$ for all i) and then finding the corresponding A_i's by solving the linear equations that result from substituting these values of X_i in (6.18) and (6.19). In fact it turns out that in many cases the supremum occurs at one of the possibilities for the X_i's suggested above.

We have not been able to show, in general, that the local maximum obtained from the constrained optimisation problem $\mathcal{P}^*$ is also a global maximum. However, for any problem which we have treated analytically the local maximum obtained does in fact turn out to be a global maximum. Also, we have varied the starting points for the problems which we have solved numerically and found that the the same solution is obtained, thus indicating that this solution is probably the global maximum.

6.3(a) The Hyp/G/1/1 system.

Using the above method we can get analytic results for small simple systems. An example of one of these systems is the $Hyp/G/1/1$ system in which arrivals with a hyperexponential distribution are offered to a single general service queue with service rate μ. The hyperexponential distribution is often used to model "bursty" traffic, when the mean to variance ratio of the arrival stream distribution is larger than one. The distribution function for the hyperexponential function with n separate arrival phases is

$$F(t) = \begin{cases} 0 & t \leq 0 \\ \sum_{k=1}^{n} \pi_k (1 - e^{-\lambda_k t}), & t > 0, \end{cases}$$

where

$$\sum_{k=1}^{n} \pi_k = 1.0.$$

The parameter λ_k is the rate at which the arrival lifetime is worked off in phase k and π_k is the probability that a new arrival will enter phase k. We use a phase interpretation for the state space, defining the process to be in state (k, j) when the arrival process is in phase k, $k = 1, ..., n$, and there are j customers in the system, $j = (0, 1)$. The time congestion is then given by

$$H(G) = \sum_{k=1}^{n} \pi(k, 1)(G)$$

where the $\pi(k, j)(G)$'s are the stationary distributions of the process being in state (k, j) when the service distribution is $G(.)$.

For the case when $n = 2$ we can obtain analytic results for the sensitivity bounds for this system using the scheme described in this paper. This is done in Taylor and Henderson (1988), where the problem of finding the maximum or minimum time congestion in a $Hyp/G/1/1$ system when there are two arrival phases is shown to reduce to

$$(max/min) \quad \left[\frac{\lambda_1 \lambda_2 \alpha}{\mu \alpha^2 + \lambda_1 \lambda_2 \alpha - \alpha(\lambda_1 - \lambda_2)^2 \pi_1 \pi_2 (X_1 - 1)} \right]$$

where

$$\alpha = (\lambda_1 \pi_2 + \lambda_2 \pi_1)$$

such that

$$e^{-\alpha} \leq X_1 < 1.0.$$

The maximum time congestion obviously occurs when $X_1 = 1.0$ and is equal to

$$\frac{\lambda_1 \lambda_2}{\mu\alpha + \lambda_1 \lambda_2}.$$

This corresponds to the service distribution described by (6.13). The minimum time congestion occurs when $X_1 = e^{-\alpha}$ and this corresponds to the deterministic service distribution as described earlier in this chapter.

λ_1	λ_2	Max.	Min.
1.0	1.0	0.5	0.5
2.0	0.6667	0.5	0.4218
5.0	0.5556	0.5	0.3846
10.0	0.5263	0.5	0.3564
1000.0	0.5003	0.5	0.3336

Table(6.1). *The sensitivity bounds on the time congestion for the $Hyp/G/1/1$ queueing system with two phases, $\pi_1 = \pi_2 = 0.5$ and an overall arrival rate of 1.0.*

Table (6.1) and table (6.2) show two sets of numerical results obtained from the program. In both cases the general service rate is 1.0 and the arrival rates are chosen so that the overall arrival rate offered to the queue is 1.0. In the first case, the phase transition probabilities are the same, that is $\pi_1 = \pi_2 = 0.5$, and in the second case the phase transition probabilities $\pi_1 = 0.9$ and $\pi_2 = 0.1$ have been chosen. The numerical results obtained from the program correspond, as expected, to the analytic results. Figure 6.1 and figure 6.2 display these results. Note that in the graphs the maximum and minimum time congestions are the same when

λ_1	λ_2	Max.	Min.
0.9009	100.0	0.6410	0.4762
0.9184	5.0	0.6081	0.5168
0.9310	3.0	0.5870	0.5344
1.0	1.0	0.5	0.5
1.05	0.7	0.4565	0.4508
1.8	0.2	0.2657	0.2024
9.9	0.11	0.1800	0.1012
90.9	0.101	0.1679	0.0918

Table (6.2). *The sensitivity bounds on the time congestion for the $Hyp/G/1/1$ queueing system with two phases, $\pi_1 = 0.9$ and $\pi_2 = 0.1$ and an overall arrival rate of 1.0.*

$\lambda_1 = \lambda_2 = 1.0$. This is not surprising since in this case the hyperexponential input is equivalent to a single Poisson arrival stream with an arrival rate of 1.0. It is well known that a $M/G/1/1$ queue is insensitive. Also note that even though the method only gives an upper and lower bound for the time congestion, in this case the actual bounds are obtained.

When there are more than two arrival phases analytic results are much harder to obtain. In many cases only numerical results can be obtained. Table (6.3) gives some numerical results for the $Hyp/G/1/1$ system for a varying number of arrival phases. For these results a hyperexponential stream with n separate arrival phases is offered to a single general server queue with phase arrival rates equal to $1, 2, 3, ..., n$ respectively and $\pi_i = 1/n$ for $i = 1, ..., n$.

6.3(b) The M/GM/n/n system.

Another queueing system that can be investigated using the above method is one in which a Poisson arrival stream is offered initially to a server which has

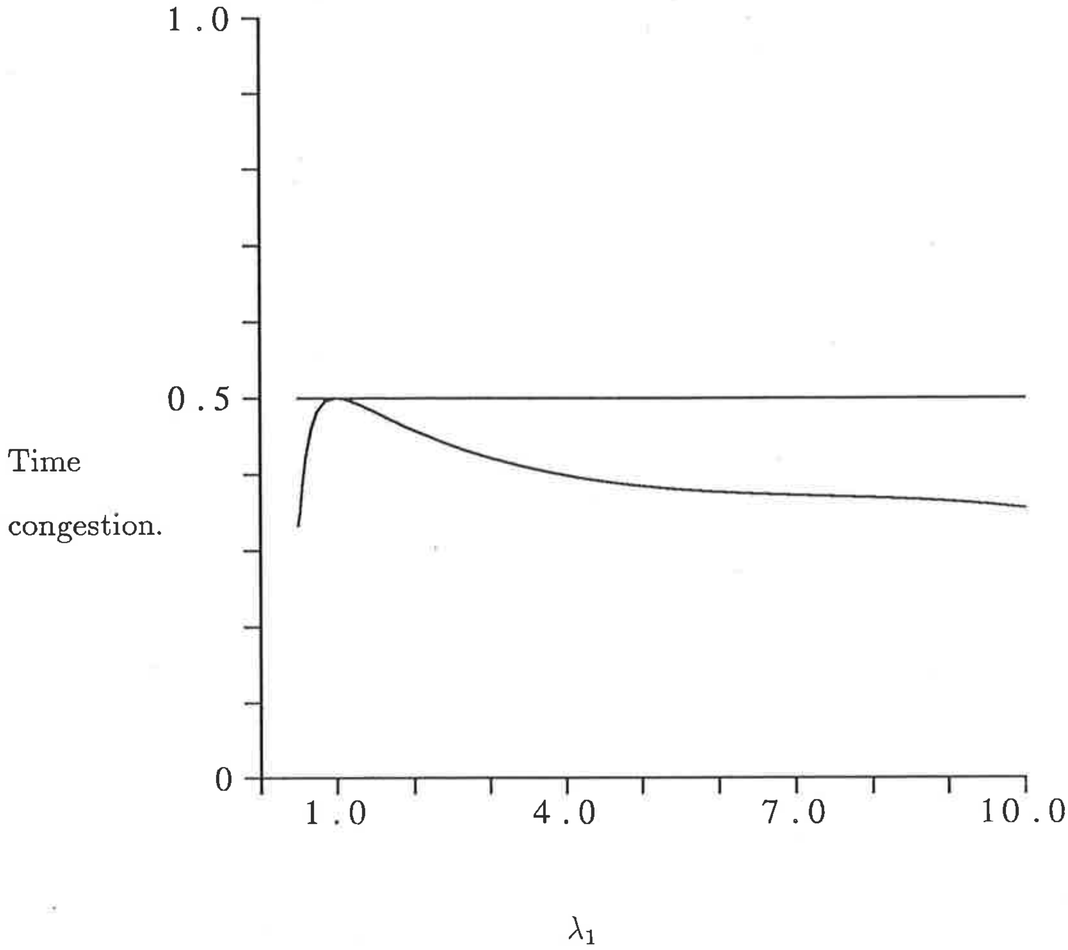

Figure 6.1. *The sensitivity bounds on the time congestion for the $Hyp/G/1/1$ queueing system with two phases, $\pi_1 = \pi_2 = 0.5$ and an overall arrival rate of 1.0.*

a general service distribution but if that server is busy the customer will then be offered to $n-1$ negative exponential servers, all having service rates 1.0. A system similar to this one with all servers having the same general service distribution was investigated by van Dijk (1987). However, the requirement of our model that there is only a single generally distributed lifetime necessitate the modification to exponentially distributed overflow servers. Numerical results for the time conges-

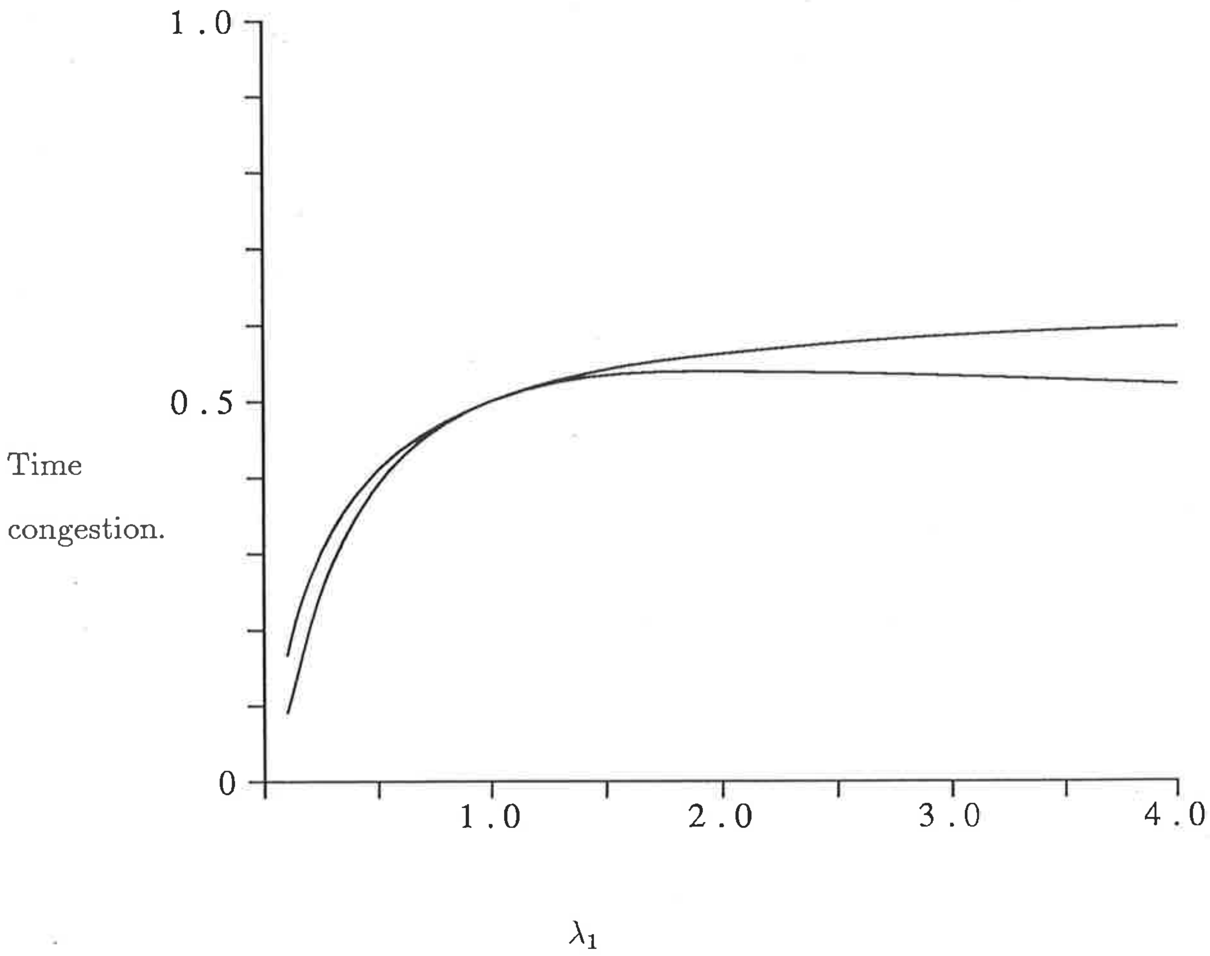

Figure 6.2. *The sensitivity bounds on the time congestion for the $Hyp/G/1/1$ queueing system with two phases, $\pi_1 = 0.9$ and $\pi_2 = 0.1$ and an overall arrival rate of 1.0.*

tion when the arrival rate is 4.0 are shown in table (6.4) and graphical results are shown in figure 6.3. Numerical results when the single general server is replaced by a negative exponential server are also shown. In this case the results can be calculated analytically as the system is just an $M/M/n/n$ system. The numerical and graphical results show that the calculated maximum and minimum values for

n	Max.	Min.
1	0.5	0.5
2	0.571	0.551
3	0.6206	0.5828
4	0.6575	0.6052
5	0.6865	0.6221
6	0.7101	0.6356
7	0.7297	0.6467
8	0.7464	0.6560
9	0.7608	0.6640
10	0.7735	0.6711
11	0.7846	0.6772

Table (6.3). *The sensitivity bounds on the time congestion for the $Hyp/G/1/1$ queueing system with n phases, $\lambda_i = i$ and $\pi_i = 1/n$.*

the time congestion do in fact straddle the values obtained when the general server is replaced by a server with a negative exponential distribution. When $n = 1$, the three values are the same as at this point the system is insensitive. As n gets very large, the three values all converge to zero as would be expected. For this queueing system there exists a very simple way of getting an upper bound on the time congestion using similar techniques to those used by van Dijk and others described in chapter 5. If the general server in the $M/GM/n/n$ queue is removed we get an $M/M/n-1/n-1$ queue. The time congestion for this queue is obviously less than that for the $M/GM/n/n$ queue so a simple upper bound on the time congestion in the $M/GM/n/n$ queue is just the time congestion in the Erlang system with $n-1$ servers. This result is easily calculated. In figure 6.3 this simple upper bound on the time congestion is compared with the results that have been obtained using the method described in this chapter. When a low number of circuits are used the simple upper bound is considerably larger than the bound we have obtained, for a large number of circuits the two results are close to one another. The results for

n	Max.	-ve exp.	Min.
1	0.6400	0.6154	0.6144
2	0.4923	0.4507	0.4484
3	0.3605	0.3107	0.3071
4	0.2485	0.1991	0.1947
5	0.1593	0.1172	0.1127
6	0.0937	0.0627	0.0588
7	0.0502	0.0304	0.0274
8	0.0244	0.0133	0.0114
9	0.0108	0.0053	0.0042

Table (6.4). *The sensitivity bounds on the time congestion for the $M/GM/n/n$ queueing system with the arrival rate equal to 4.0. The time congestion for the case when the general lifetime is negative exponentially distributed is also given.*

this system once again correspond to the values that are expected.

6.3(c) The GI/M/1/2 system.

Theorem 6.1 in this chapter has been derived for cases in which the eigenvalues of the matrix Q_1C^{-1} are distinct. It was stated that when this is not the case the theorem can be modified accordingly. A system for which it is necessary to implement these modifications is the $GI/M/1/2$ system which will be investigated in this section. The methods for this queue can be extended to the more general $GI/M/1/n+1$ system, although the problem very quickly becomes much more complicated. To get realistic results for this system, bounds on the derivative of the Laplace-Stieljes transform of the call inter-arrival time probability distribution must be obtained. We obtain these bounds by using the methods of complete Tchebycheff systems (see Kreĭn and Nudel'man (1973)).

A complete Tchebycheff system on the interval $[0, \infty)$ is a system of $n+1$

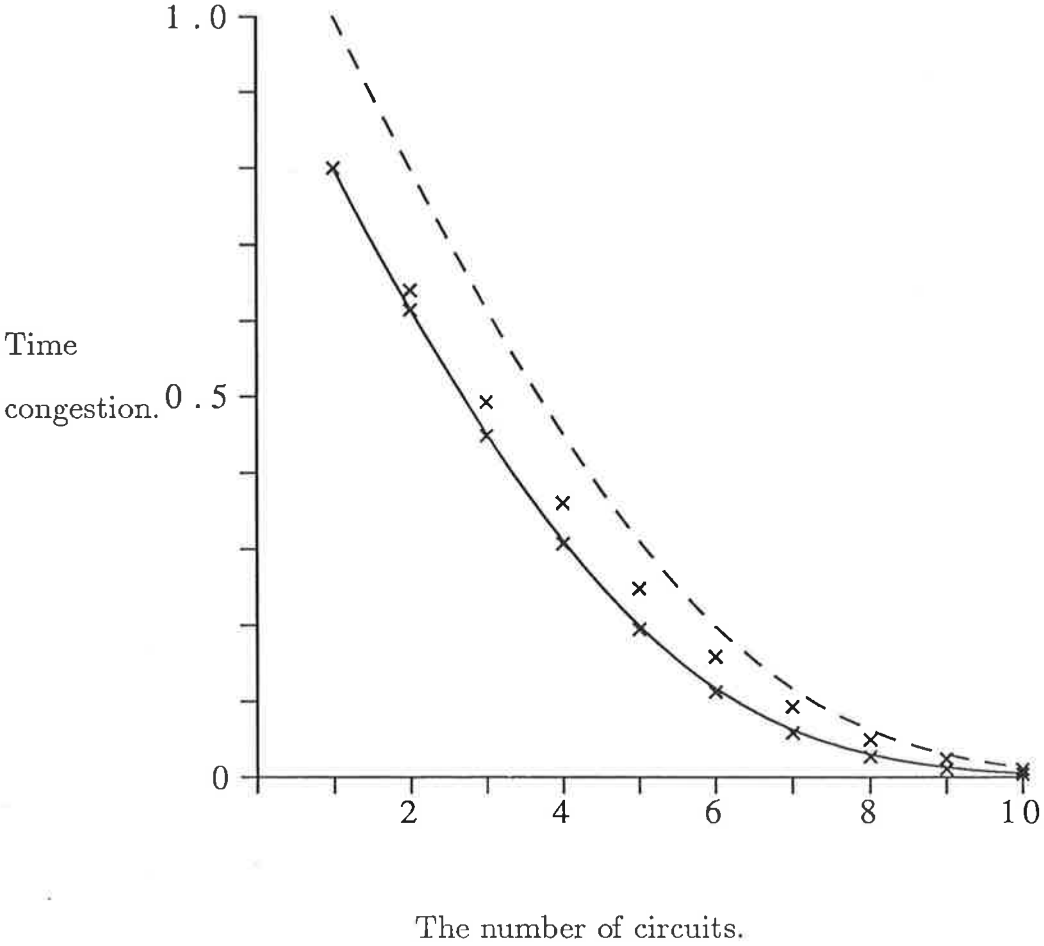

Figure 6.3. *The sensitivity bounds on the time congestion for the $M/GM/n/n$ queueing system with the arrival rate equal to 4.0. The continuous line represents the time congestion when the general lifetime is negative exponentially distributed. The dashed line is another upper bound for the time congestion obtained by looking at the $M/M/n-1/n-1$ queue.*

real valued, continuous functions $y_0, y_1, ..., y_n$ such that any linear combination of these functions has at most n zeros on $[0, \infty)$. One of the simplest examples of a complete Tchebycheff system is a system of $n+1$ algebraic polynomials each with degree at most n. The geometry of the complete Tchebycheff system $1, y, y^2, ..., y^n$ can be generalised to encompass all other complete Tchebycheff systems. This work has ramifications in areas such as approximation theory and the theory of inequalities. It is with this second application that we are interested in this section.

Lemma 6.2.

For $s > 0$, $\{1, e^{-sy}, y\}$ and $\{1, e^{-sy}, y, ye^{-sy}\}$ are complete Tchebycheff systems in the interval $0 \leq y < \infty$.

Proof.

Note first that the function ye^{-sy} is positive for all $0 \leq y < \infty$. Also note that the two conditions

$$\lim_{y \to \infty} \frac{1}{y} = 0$$

$$\lim_{y \to \infty} \frac{e^{-sy}}{y} = 0$$

are satisfied. For $\{1, e^{-sy}, y, ye^{-sy}\}$ to be a complete Tchebycheff system any linear combination of the four continuous functions 1, e^{-sy}, y and ye^{-sy} must have at the most 3 zeros for any $s > 0$ and any $y > 0$. This result is clearly apparent using Rolle's theorem and the fact that e^{-sy}, ye^{-sy} is a complete Tchebycheff system. Using Rolle's theorem we also find that the system $\{1, e^{-sy}, y\}$ has at the most 2 zeros for $s > 0$ and $y > 0$. So Lemma 6.2 is true using the results of Kreĭn and Nudel'man (1973), section **V**.1.2. □

Lemma 6.3.

For any probability distribution with distribution function $G(.)$ with Laplace-

Stieltjes transform of $G(.)$ given by

$$\widehat{G}(s) = \int_0^\infty e^{-ys} dG(y)$$

and the derivative of the Laplace-Stieltjes transform of $G(.)$ given by

$$\widehat{G}'(s) = -\int_0^\infty ye^{-ys} dG(y),$$

the following bounds on $\widehat{G}'(s)$ hold:

$$\frac{\widehat{G}(s) ln(\widehat{G}(s))}{s} \leq \widehat{G}'(s) \leq -e^{-s\xi}$$

where ξ is the solution of the following equation

$$(1 - \widehat{G}(s))\xi = (1 - e^{-s\xi})$$

and $s > 0$.

Proof.

It has been shown in Lemma 6.2 that $\{e^{-sy}, 1, y, ye^{-sy}\}$ is a complete Tchebycheff system where $s > 0$ and $y > 0$. Using this fact we can find some bounds on $\widehat{G}'(s)$. Theorem 1.1 in section V of Kreĭn and Nudel'man (1973) shows how a minimum bound on the Laplace-Stieltjes transform can be found. Firstly we must solve the system of equations

$$\widehat{G}(s) = \rho_1 e^{-s\xi_1} + \rho_2 e^{-s\xi_2}$$

$$1 = \rho_1 + \rho_2$$

$$1 = \rho_1\xi_1 + \rho_2\xi_2$$

and then the minimum bound on $\widehat{G}'(s)$ is given by

$$\rho_1\xi_1 e^{-s\xi_1} + \rho_2\xi_2 e^{-s\xi_2}$$

where $\xi_1 = 0$. The solution to these equations is found to be

$$\rho_1 = 1 - \frac{1}{\xi_2}$$

$$\rho_2 = \frac{1}{\xi_2}$$

where ξ_2 is the solution to the equation

$$(1 - \widehat{G}(s))\xi_2 = (1 - e^{-s\xi_2}).$$

The minimum bound on $-\widehat{G}'(s)$ is now found to be $e^{-s\xi_2}$.

Theorem 1.2 in section **V** of Kreĭn and Nudel'man (1973) shows how a maximum bound on the Laplace-Stieltjes transform can be found. Firstly we must ensure that

$$c = \lim_{t \to \infty} \frac{te^{-st}}{t} < \infty.$$

This is true with $c = 0$. Now we must solve the system of equations

$$\widehat{G}(s) = \rho_1 e^{-s\xi_1}$$

$$1 = \rho_1$$

$$1 = \rho_1 \xi_1 + M$$

and then the maximum bound on $-\widehat{G}'(s)$ is given by

$$\rho_1 \xi_1 e^{-s\xi_1} + cM.$$

It can be seen, therefore, that since $c = 0$ the maximum bound on $-\widehat{G}'(s)$ is

$$\frac{\widehat{G}(s) ln(\widehat{G}(s))}{s}.$$

Hence the lemma is proved. $\square$

Using theorem 6.1 and lemma 6.3 we can now find bounds on the time congestion in the $GI/M/1/2$ queueing system. For the $GI/M/1/2$ queueing system it is found that

$$Q_1 = \begin{pmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix},$$

$$Q_2 = \begin{pmatrix} -a & a & 0 \\ 0 & -a & a \\ 0 & 0 & 0 \end{pmatrix}$$

and

$$C = Ia.$$

There are two distinct eigenvalues of Q_1C^{-1}, $\alpha_0 = 0$ and the repeated eigenvalue $\alpha_1 = -1/a$. The eigenvector associated with α_0 is $\mathbf{w}_0 = (1,0,0)$. An eigenvector for the eigenvalue α_1 is $\mathbf{w}_1 = (1,-1,0)$ and using the Jordan-Canonical form, a generalised eigenvector of Q_1C^{-1} is found to be $\mathbf{w}_2 = (0,a,-a)$. Solving (6.4a) it is found that

$$\mathbf{P}(y,G) = \mathbf{A}^{(G)}e^{\Lambda y}\mathbf{T}(1-G(y))$$

where Λ is the canonical form of Q_1C^{-1} and $\mathbf{T}$ is a matrix of eigenvectors that transforms Q_1C^{-1} to its Jordan-Canonical form. So

$$\Lambda = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1/a & 1 \\ 0 & 0 & -1/a \end{pmatrix}$$

and therefore

$$e^{\Lambda y} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & e^{-y/a} & ye^{-y/a} \\ 0 & 0 & e^{-y/a} \end{pmatrix}.$$

Also

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & a & -a \\ 1 & -1 & 0 \end{pmatrix}.$$

From the above it is found that

$$\mathbf{P}(y,G) = \begin{pmatrix} (A_0 + A_1e^{-y/a} + A_2ye^{-y/a})(1-G(y)) \\ (-A_1e^{-y/a} + aA_2e^{-y/a} - A_2ye^{-y/a})(1-G(y)) \\ -aA_2e^{-y/a}(1-G(y)) \end{pmatrix}.$$

Now

$$\int_0^\infty ye^{-y/a}(1-G(y))dy$$

$$= [-ay(1-G(y))e^{-y/a}]_0^\infty + \int_0^\infty ae^{-y/a}(1-G(y))dy - \int_0^\infty aye^{-y/a}g(y)dy$$

$$= a^2 - a^2\widehat{G}(1/a) + a\widehat{G}'(1/a).$$

Let $\widehat{G} = \widehat{G}(1/a)$ and $\widehat{G}' = \widehat{G}'(1/a)$ so that

$$\int_0^\infty \mathbf{P}(y,G)dy = \begin{pmatrix} A_0 + aA_1(1-\widehat{G})) + A_2(a^2 - a^2\widehat{G} + a\widehat{G}') \\ -aA_1(1-\widehat{G}) - aA_2\widehat{G}' \\ -a^2A_2(1-\widehat{G}) \end{pmatrix}$$

and

$$\int_0^\infty \mathbf{P}(y,G)h(y)dy = \begin{pmatrix} A_0 + A_1\widehat{G} - A_2\widehat{G}' \\ -A_1\widehat{G} + aA_2\widehat{G} + A_2\widehat{G}' \\ -aA_2\widehat{G} \end{pmatrix}.$$

Using these results and (6.4b) we must have

$$-aA_0 - aA_1 = 0$$

$$-aA_0 - aA_1 + a^2A_2 - aA_1\widehat{G} + aA_2\widehat{G}' = 0$$

$$a^2A_2 - aA_1\widehat{G} + aA_2\widehat{G}' = 0.$$

Solving these equations and using the fact that $A_0 = 1$ we find that

$$A_2 = \frac{-\widehat{G}}{a + \widehat{G}'}.$$

So the time congestion in the $GI/M/1/2$ queueing system is given by

$$\frac{a^2\widehat{G}(1-\widehat{G})}{a+\widehat{G}'}. \tag{6.20}$$

To find the maximum time congestion (6.20) must be maximised subject to

$$e^{-1/a} \leq \widehat{G} \leq 1$$

and

$$\frac{\widehat{G}ln(\widehat{G})}{s} \leq \widehat{G}' \leq -e^{-\xi/a}$$

where ξ is the solution to

$$(1-\widehat{G})\xi = (1-e^{-\xi/a}).$$

It is found that the maximum time congestion for the $GI/M/1/2$ queue is achieved when

$$\widehat{G} = \begin{cases} 0.4578 & a \leq 1.2797 \\ e^{-1/a} & a > 1.2797 \end{cases}$$

and

$$\widehat{G}' = -a\widehat{G}ln(\widehat{G}).$$

The maximum possible time congestion in the $GI/M/1/2$ queue is achieved by different call inter-arrival probability distributions depending on the call arrival intensity a. When the call arrival intensity is greater than 1.2797 the call inter-arrival probability distribution that maximises the time congestion is the deterministic distribution. When the call arrival intensity is below 1.2797, the call inter-arrival probability distribution that maximises the time congestion is one which has mass $a \times 0.7813$ at $1/a$ and a mass at infinity such that the overall call arrival intensity is a. This distribution is something like a deterministic distribution with intensity $a \times 0.7813$ but every so often there is a large inter-arrival time, such that the overall call arrival intensity is a.

The upper bound on the time congestion in the $GI/M/1/2$ queue is shown in figure 6.4. A comparison is also made between this upper bound on the time congestion and the upper bound on the time congestion in a $GI/M/1/1$ queue which obviously is also an upper bound for the time congestion in the $GI/M/1/2$ queue. The results obtained using the technique described in this chapter gives a much better bound than the bound found by looking at the $GI/M/1/1$ queue, especially for low intensities.

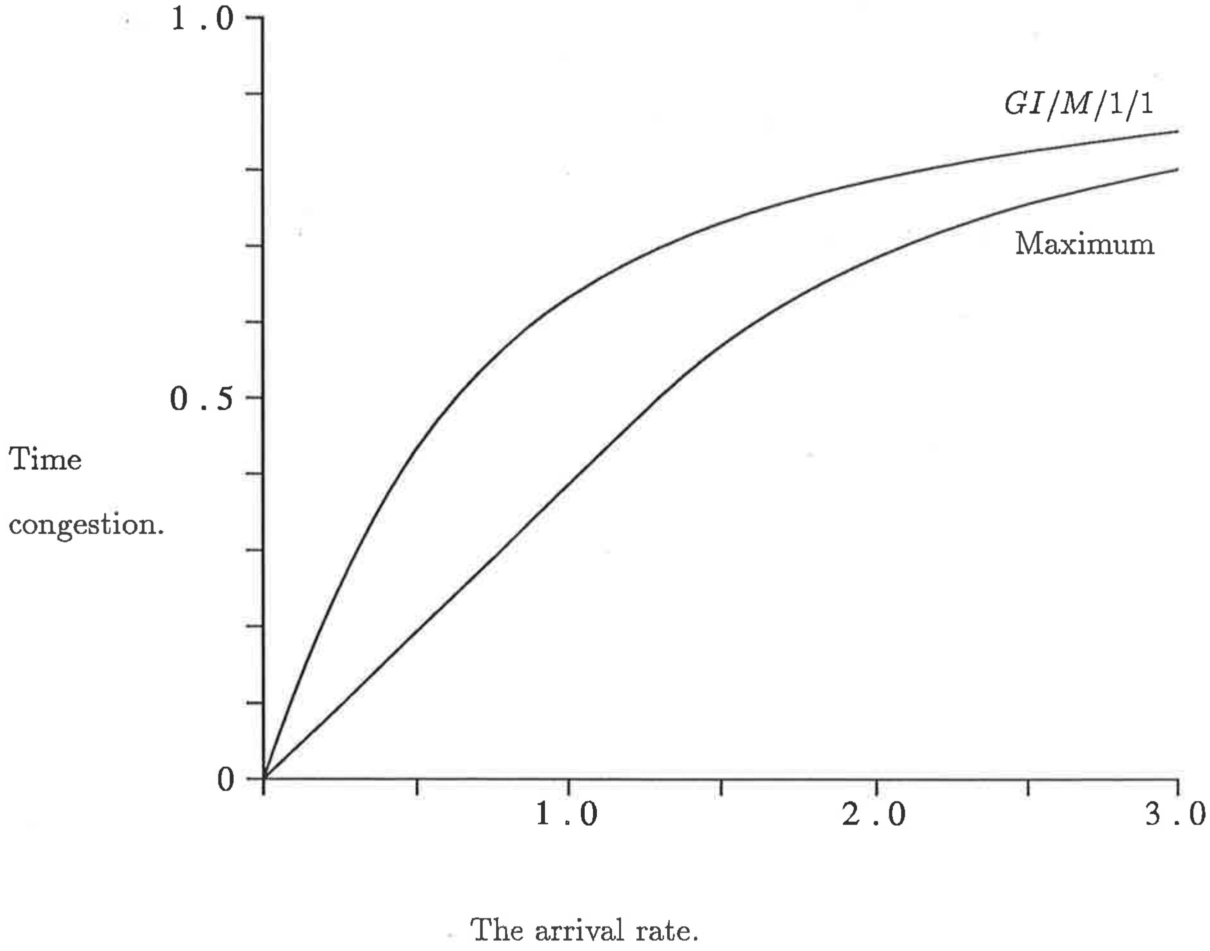

Figure 6.4. *The upper bound on the time congestion in the $GI/M/1/2$ queue. The upper line in the figure is the upper bound for the time congestion in the $GI/M/1/1$ queue.*

CHAPTER 7

SENSITIVITY BOUNDS FOR THE GI/M/n/n QUEUE.

7.1 The GI/M/n/n queue.

We will now look at a specific GSMP, the $GI/M/n/n$ queue with arrival rate a. A GI arrival distribution is one for which the inter-arrival lifetimes are generally and independently distributed. For this system the arrivals are offered to n servers each with negative exponentially distributed service times. The states of this GSMP correspond to the number of busy servers in the system and so there are $n+1$ possible states. If $c(x, A) = a, \ \forall x, A$ the general lifetime is being worked off at rate a. This is equivalent to an arrival rate a. When a general lifetime has been worked off, an arrival occurs and another general lifetime begins. If there is a spare server this server will service the new arrival and if no spare server exists the call is lost. Without loss of generality assume that the service times have mean 1 and therefore the rate at which a transition from state i to state $i-1$ takes place is i, $i \in \{1, ..., n\}$. So we have

$$q^E(i,j) = \begin{cases} a, & j = i+1, \\ 0, & \text{otherwise} \end{cases} \quad i,j = 0, ..., n,$$

$$q^I(i,j) = \begin{cases} i, & j = i-1, \\ -i, & j = i, \\ 0, & \text{otherwise} \end{cases} \quad i,j = 0, ..., n, .$$

$$C(i,j) = \begin{cases} a, & j = i, \\ 0 & \text{otherwise,} \end{cases} \quad i = 0, ..., n$$

It can be shown that

$$Q_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & . & . & . & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & . & . & . & 0 & 0 & 0 \\ 0 & 2 & -2 & 0 & . & . & . & 0 & 0 & 0 \\ 0 & 0 & 3 & -3 & . & . & . & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & . & . & . & 2-n & 0 & 0 \\ 0 & 0 & 0 & 0 & . & . & . & n-1 & 1-n & 0 \\ 0 & 0 & 0 & 0 & . & . & . & 0 & n & -n \end{pmatrix}$$

and

$$Q_2 = \begin{pmatrix} -a & a & 0 & 0 & . & . & . & 0 & 0 & 0 \\ 0 & -a & a & 0 & . & . & . & 0 & 0 & 0 \\ 0 & 0 & -a & a & . & . & . & 0 & 0 & 0 \\ 0 & 0 & 0 & -a & . & . & . & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & . & . & . & -a & a & 0 \\ 0 & 0 & 0 & 0 & . & . & . & 0 & -a & a \\ 0 & 0 & 0 & 0 & . & . & . & 0 & 0 & 0 \end{pmatrix}.$$

The eigenvalues of $Q_1 C^{-1}$ are

$$-\alpha_i = \frac{-i}{a}, \qquad i = 0, ..., n$$

and the corresponding left eigenvectors are $\mathbf{w}_i$ with components

$$(\mathbf{w}_i)_j = \begin{cases} (-1)^j \binom{i}{j}, & 0 \leq j \leq i \leq n, \\ 0, & \text{otherwise.} \end{cases}$$

Given these results, from (6.17b) and (6.17c) we get

$$-1 \leq X_2 - 2X_1 \tag{7.1}$$

and

$$0 \leq X_{i+1} - 2X_i + X_{i-1} \qquad i = 2, ..., n-1. \tag{7.2}$$

The dot product of the right and left eigenvectors corresponding to different eigenvalues of a matrix is 0. Therefore since $\mathbf{e}$ is the right eigenvector of $Q_1 C^{-1}$ corresponding to the eigenvalue 0

$$\mathbf{w}_i \cdot \mathbf{e} = \begin{cases} 1, & i = 0, \\ 0, & i = 1, ..., n \end{cases}$$

and so from (6.19)

$$A_0 = 1. \tag{7.3}$$

Also

$$\mathbf{w}_i Q_1 = (-a)\alpha_i \mathbf{w}_i$$

and

$$\mathbf{w}_i Q_2 = \begin{cases} (-a)\mathbf{w}_{i+1}, & \text{i=0,...,n-1,} \\ (-a)\hat{\mathbf{w}}, & \text{i=n} \end{cases}$$

where

$$(\hat{\mathbf{w}})_k = \begin{cases} (-1)^k \binom{n+1}{k}, & \text{k=0,...,n-1,} \\ (-1)^n \binom{n}{n-1}, & \text{k=n.} \end{cases}$$

Hence from (6.18)

$$\sum_{i=1}^{n} A_i [X_i - 1] \mathbf{w}_i - A_0 \mathbf{w}_1 - \sum_{i=1}^{n-1} A_i X_i \mathbf{w}_{i+1} - A_n X_n \hat{\mathbf{w}} = 0. \tag{7.4}$$

In this case we want to look at the maximum possible time congestion in the system and so the objective function we are looking at is the probability that n servers are occupied. Since

$$\int_0^\infty \mathbf{P}(y, G) dy = A_0 \mathbf{w}_0 + \sum_{i=1}^{n} \frac{A_i}{\alpha_i} [1 - X_i] \mathbf{w}_i$$

is the probability density vector, the nth component of this vector is the probability that n servers are occupied. The nth component of all the eigenvectors apart from $\mathbf{w}_n$ is zero. The nth component of $\mathbf{w}_n$ is $(-1)^n$ and so the probability that n servers are busy is given by

$$P_n = (-1)^n \frac{aA_n}{n} [1 - X_n]. \tag{7.5}$$

7.2 The Method of Lagrange Multipliers.

To solve a non-linear bounded optimisation problem with non-linear equality and inequality constraints the method of Lagrange Multipliers can be used (see for example Avriel (1976)). Suppose the optimisation problem is

$$Max \quad f(\mathbf{x}) \tag{7.6}$$

$$\text{such that} \qquad h_j(\mathbf{x}) = 0, \qquad j = 1, ..., p \tag{7.7}$$

$$\text{and} \qquad g_i(\mathbf{x}) \geq 0, \qquad i = 1, ..., m. \tag{7.8}$$

If $\mathbf{x}^*$ is a feasible solution to (7.6), (7.7) and (7.8) and there exist vectors $\mathbf{\Lambda}^* = (\lambda_i^*,\ i = 1, ..., m)$ and $\mathbf{\Phi}^* = (\phi_j^*,\ j = 1, ..., p)$ satisfying

$$\nabla_x L(\mathbf{x}^*, \mathbf{\Lambda}^*, \mathbf{\Phi}^*) \equiv \nabla_x f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_x g_i(\mathbf{x}^*) + \sum_{j=1}^{p} \phi_j^* \nabla_x h_j(\mathbf{x}^*) = 0, \tag{7.9}$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0, \quad i = 1, ..., m, \tag{7.10}$$

$$\lambda_i^* \geq 0, \quad i = 1, ..., m \tag{7.11}$$

and for every $\mathbf{z} \neq 0$ such that $\mathbf{z} \in Z(\mathbf{x}^*)$

$$\mathbf{z}^T \left[\nabla_x^2 f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_x^2 g_i(\mathbf{x}^*) + \sum_{j=1}^{p} \phi_j^* \nabla_x^2 h_j(\mathbf{x}^*) \right] \mathbf{z} < 0 \tag{7.12}$$

where

$$Z(\mathbf{x}^*) = \{\mathbf{z} : \mathbf{z}^T \nabla_x g_i(\mathbf{x}^*) = 0, i \in I'(\mathbf{x}^*), \mathbf{z}^T \nabla_x g_i(\mathbf{x}^*) \geq 0, i \in I(\mathbf{x}^*),$$

$$\mathbf{z}^T \nabla_x h_j(\mathbf{x}^*) = 0, \quad j = 1, ..., p\}, \tag{7.13}$$

with $I(\mathbf{x}^*)$ the set of indices for which $g_i(\mathbf{x}^*) = 0$ and $I'(\mathbf{x}^*)$ the set of indices for which $g_i(\mathbf{x}^*) = 0$ and $\lambda_i^* > 0$,then $\mathbf{x}^*$ is a strict local maximum of $f(\mathbf{x})$.

Note that in general a point that is a local maximum of a problem is not necessarily a global maximum. To prove that a point satisfying the above is in fact a global maximum is often very difficult, if not impossible. In the problem that follows, it can be shown that the local maximum presented is also a global maximum.

7.3 The Problem.

If we let the vector $\mathbf{x}$ correspond to $(A_i,\ i = 0, ..., n,\ X_i,\ i = 1, ..., n)$, then using the above results we find that the problem of finding an upper bound for the time congestion in a $GI/M/n/n$ queue using (6.17), (7.1), (7.2), (7.3) and (7.4) is equivalent to

$$Max \quad f(\mathbf{x}) = P_n = (-1)^n \frac{aA_n}{n} [1 - X_n] \tag{7.14}$$

such that

$$\begin{aligned}
h_j(\mathbf{x}) &\equiv \sum_{i=1}^{n} A_i [X_i - 1] (\mathbf{w}_i)_j - A_0(\mathbf{w}_1)_j \\
&\qquad - \sum_{i=1}^{n-1} A_i X_i (\mathbf{w}_{i+1})_j - A_n X_n (\hat{\mathbf{w}})_j = 0,\ j = 0, ..., n, \\
h_{n+1}(\mathbf{x}) &\equiv A_0 - 1 = 0, \\
g_1(\mathbf{x}) &\equiv X_2 - 2X_1 + 1 \geq 0, \\
g_i(\mathbf{x}) &\equiv X_{i+1} - 2X_i + X_{i-1} \geq 0, \quad i = 2, ..., n-1, \\
g_n(\mathbf{x}) &\equiv 1 - X_1 \geq 0, \\
g_{i+n-1}(\mathbf{x}) &\equiv X_{i-1} - X_i \geq 0, \quad i = 2, ..., n, \\
g_{i+2n-1}(\mathbf{x}) &\equiv X_i - e^{-i/a} \geq 0, \quad i = 1, ..., n
\end{aligned} \tag{7.15}$$

and the method of Lagrange Multipliers can be used.

7.4 The Solution.

Theorem 7.1

The solution to the problem formulated in (7.14) and (7.15) is

$$A_i^* = (-1)^i \binom{n}{i} \quad i = 0, ..., n \tag{7.16}$$

and

$$X_i^* = \left(\frac{e^{-n/a} - 1}{n}\right) i + 1 \quad i = 1, ..., n. \tag{7.17}$$

So

$$\begin{aligned} P_n^* &= (-1)^n \frac{aA_n^*}{n} [1 - X_n^*] \\ &= \frac{a}{n}(1 - e^{-n/a}). \end{aligned} \tag{7.18}$$

Proof.

The proposed solution given by (7.16), (7.17) and (7.18) can be shown to satisfy the feasible solution conditions (7.6), (7.7) and (7.8) and also satisfy the necessary optimality conditions (7.9) to (7.13) when we have

$$\begin{aligned} \phi_0^* &= -P_n^* \left(1 + \sum_{k=1}^{n-1} \frac{1}{X_k^*}\right), \\ \phi_j^* &= -P_n^* \sum_{k=j}^{n-1} \frac{1}{X_k^*}, \quad j = 1, ..., n-1, \\ \phi_n^* &= 0, \\ \phi_{n+1}^* &= -P_n^*, \end{aligned} \tag{7.19a}$$

$$\lambda_i^* = -\sum_{k=i+1}^{n} (k-i) A_k^* \mathcal{C}_1(\Phi^*, k) \quad i = 1, ..., n-1,$$

$$\lambda_n^* = 0,$$

$$\lambda_{i+n-1}^* = 0, \quad i = 2, ..., n \tag{7.19b}$$

$$\lambda_{i+2n-1}^* = 0, \quad i = 1, ..., n-1,$$

and

$$\lambda_{3n-1}^* = (-1)^n \frac{aA_n^*}{n}$$

where

$$\mathcal{C}_i(\Phi^*, j) = \sum_{k=i}^{i+j} (-1)^{k-i} \phi_k^* \binom{j}{k-i}$$

for $0 \le i \le n$ and $i + j \le n$ and

$$\mathcal{C}_1(\Phi^*, n) = \phi_1^* - n\phi_2^* + ... + (-1)^{n-2} \binom{n}{n-2} \phi_{n-1}^* + (-1)^{n-1}(n-1)\phi_n^*$$

$$= \sum_{k=1}^{n-1} (-1)^{k+1} \phi_k^* \binom{n}{k+1} + (-1)^{n-1}(n-1)\phi_n^*.$$

The proof of this result is achieved by showing that (7.7) to (7.14) are satisfied by (7.18) when (7.14) defines the objective function and (7.15) define the constraints. The algebra to show this result for the general case is messy and is given in the appendix. A proof of the result when there are 2 servers is given here. The case when there is only one server is relatively simple.

The case n=2

The vector of variables, $\mathbf{x}$, for this problem is $(A_0, A_1, A_2, X_1, X_2)$, the eigenvalues of $Q_1 C^{-1}$ are $-\alpha_0 = 0$, $-\alpha_1 = -1/a$, $-\alpha_2 = -2/a$ and the corresponding eigenvectors are $\mathbf{w}_0 = (1, 0, 0)$, $\mathbf{w}_1 = (1, -1, 0)$ and $\mathbf{w}_2 = (1, -2, 1)$, also $\mathbf{w}_2 Q_2 = -a\hat{\mathbf{w}} = -a(1, -3, 2)$. The optimisation problem as formulated by (7.14) and (7.15) can be written as

$$Max \quad f(\mathbf{x}) = P_2 = \frac{a}{2}A_2(1 - X_2)$$

such that

$$\begin{aligned}
&h_0(\mathbf{x}) \equiv -A_0 - A_1 - A_2 = 0, \\
&h_1(\mathbf{x}) \equiv A_0 + A_1 + 2A_2 + A_1X_1 + A_2X_2 = 0, \\
&h_2(\mathbf{x}) \equiv -A_2 - A_1X_1 - A_2X_2 = 0, \\
&h_3(\mathbf{x}) \equiv A_0 - 1 = 0, \\
&g_1(\mathbf{x}) \equiv X_2 - 2X_1 + 1 \geq 0, \\
&g_2(\mathbf{x}) \equiv 1 - X_1 \geq 0, \\
&g_3(\mathbf{x}) \equiv X_1 - X_2 \geq 0, \\
&g_4(\mathbf{x}) \equiv X_1 - e^{-1/a} \geq 0
\end{aligned} \tag{7.20}$$

and

$$g_5(\mathbf{x}) \equiv X_2 - e^{-2/a} \geq 0.$$

The proposed solution to this problem given by (7.16) and (7.17) is

$$\begin{aligned}
A_0^* &= 1, \\
A_1^* &= -2, \\
A_2^* &= 1, \\
X_1^* &= \frac{e^{-2/a} + 1}{2}, \\
X_2^* &= e^{-2/a},
\end{aligned} \tag{7.21}$$

and so

$$P_2^* = \frac{a}{2}(1 - e^{-2/a}).$$

The solution given by (7.21) satisfies all the constraints in (7.20). The partial derivatives of the Lagrangian given by (7.9) for this case are equivalent to

$$\begin{aligned}
\frac{\partial L}{\partial A_0} &= -\phi_0 + \phi_1 + \phi_3 = 0, \\
\frac{\partial L}{\partial A_1} &= -\phi_0 + \phi_1(X_1+1) - \phi_2 X_1 = 0, \\
\frac{\partial L}{\partial A_2} &= -\phi_0 + \phi_1(X_2+2) - \phi_2(X_2+1) + \frac{a}{2}(1-X_2) = 0, \\
\frac{\partial L}{\partial X_1} &= A_1\phi_1 - A_1\phi_2 - 2\lambda_1 - \lambda_2 + \lambda_3 + \lambda_4 = 0
\end{aligned} \tag{7.22}$$

and

$$\frac{\partial L}{\partial X_2} = A_2\phi_1 - A_2\phi_2 + \lambda_1 - \lambda_3 + \lambda_5 - a\frac{A_2}{2} = 0.$$

These equations are satisfied when we choose

$$\begin{aligned}
\phi_0^* &= -P_2^*(1 + \frac{1}{X_1^*}), \\
\phi_1^* &= -P_2^*\frac{1}{X_1^*}, \\
\phi_2^* &= 0, \\
\phi_3^* &= -P_2^*, \\
\lambda_1^* &= -\phi_1^*, \\
\lambda_2^* &= 0, \\
\lambda_3^* &= 0, \\
\lambda_4^* &= 0
\end{aligned} \tag{7.23}$$

and

$$\lambda_5^* = a\frac{A_2^*}{2}$$

as proposed by (7.19). These values also satisfy (7.10) and (7.11). To show that (7.12) is satisfied the vectors $\mathbf{z} \in Z(\mathbf{x}^*)$ must be found. These are given by (7.13). The components of $\mathbf{z}$ correspond to the variables in this problem so let $\mathbf{z}^T = (z_{A_0}, z_{A_1}, z_{A_2}, z_{X_1}, z_{X_2})$. The vector $\mathbf{z}$ must firstly satisfy $\mathbf{z}^T\nabla_x g_i(\mathbf{x}^*) = 0$ for the inequality constraints where $g_i(\mathbf{x}^*) = 0$. The inequality constraints g_1 and

g_5 are equal to zero so

$$\mathbf{z}^T \nabla_x g_1(\mathbf{x}^*) = \mathbf{z}^T (0,0,0,-2,1)^T = 0$$

$$\Rightarrow \quad -2z_{X_1} + z_{X_2} = 0$$

and

$$\mathbf{z}^T \nabla_x g_5(\mathbf{x}^*) = \mathbf{z}^T (0,0,0,0,1)^T = 0$$

$$\Rightarrow \quad z_{X_2} = 0$$

and therefore $z_{X_1} = z_{X_2} = 0$. It is also required that

$$\mathbf{z}^T \nabla_x h_j(\mathbf{x}^*) = 0, \quad j = 0, ..., 3$$

and so

$$\mathbf{z}^T \nabla_x h_0(\mathbf{x}^*) = \mathbf{z}^T (-1,-1,-1,0,0)^T = 0,$$

$$\mathbf{z}^T \nabla_x h_1(\mathbf{x}^*) = \mathbf{z}^T (1, 1 + X_1, 2 + X_2, A_1, A_2)^T = 0,$$

$$\mathbf{z}^T \nabla_x h_2(\mathbf{x}^*) = \mathbf{z}^T (0, -X_1, -(1 + X_2), -A_1, -A_2)^T = 0$$

and

$$\mathbf{z}^T \nabla_x h_3(\mathbf{x}^*) = \mathbf{z}^T (1,0,0,0,0)^T = 0.$$

For all of these to be satisfied it is necessary for $z_{A_0} = z_{A_1} = z_{A_2} = 0$. Therefore the only possible vector $\mathbf{z} \in Z(\mathbf{x}^*)$ is the zero vector and (7.12) is trivially satisfied. The necessary conditions (7.7) through to (7.13) have been shown to be satisfied by the proposed solution which is therefore a strict local maximum of $f(\mathbf{x})$. So P_n^* is a local maximum for the time congestion in the $GI/M/n/n$ queue when $n = 2$.

To prove that this local maximum is also a global maximum, it must be shown that no other local maximum exist that is larger than the presented result. The equality constraints h_0, h_1, h_2 and h_3 can be reduced to the single constraint

$$h(\mathbf{x}) \equiv -A_2 \left[1 + X_2 - X_1\right] + X_1 = 0.$$

The problem is now in the three variables X_1, X_2 and A_2. Rearranging the above gives

$$A_2 = \frac{X_1}{1 - X_1 + X_2}$$

and for any feasible solution A_2 must be positive since $0 \leq X_1 \leq 1$ and $0 \leq X_2$.

The partial derivatives of the Lagrangian are now given by

$$\frac{\partial L}{\partial A_2} = \phi(1 - X_1 + X_2) + \frac{a}{2}(1 - X_2) = 0,$$
$$\frac{\partial L}{\partial X_1} = -\phi(A_2 + 1) - 2\lambda_1 - \lambda_2 + \lambda_3 + \lambda_4 = 0 \tag{7.24}$$

and

$$\frac{\partial L}{\partial X_2} = \phi A_2 + \lambda_1 - \lambda_3 + \lambda_5 - \frac{a}{2}A_2 = 0.$$

Rearranging the first of these partial derivatives gives

$$\phi = -\frac{a(1 - X_2)}{2(1 - X_1 + X_2)}$$

therefore ϕ must be negative or zero, since $1 - X_2 \geq 0$ and $1 - X_1 + X_2 \geq 0$. So it is found from the other two partial derivatives that

$$2\lambda_1 + \lambda_2 - \lambda_3 - \lambda_4 = -\phi(A_2 + 1)$$

$$\Rightarrow 2\lambda_1 + \lambda_2 - \lambda_3 - \lambda_4 \geq 0 \tag{7.25}$$

and similarly

$$\lambda_1 - \lambda_3 + \lambda_5 > 0. \tag{7.26}$$

Any local maximum, which includes the global maximum, of the optimisation problem must satisfy both the above two equations and the inequalities of (7.20). Note that from (7.11) the λ's must be positive or zero and from (7.10) either an inequality constraint must be satisfied with equality or the corresponding Lagrange Multiplier must be zero.

For constraint (7.25) to be an equality $\phi = 0$ and this means that $X_2 = 1$, $X_1 = 1$ and $P_2 = 0$. Whilst this is a feasible solution it is obviously not the global maximum of the problem.

Constraint (7.25) must be a strict inequality constraint and therefore either λ_1 or λ_2 or both must be positive therefore $X_1 = 1$ or $X_2 - 2X_1 + 1 = 0$ or both. The case when $X_1 = 1$ has already been shown to be non-optimal so $X_2 - 2X_1 + 1 = 0$ and $X_1 < 1$. Since $X_1 < 1$ it follows that $\lambda_2 = 0$. Using this and manipulating (7.25) and (7.26) it is found that

$$-3\lambda_3 - \lambda_4 + 2\lambda_5 > 0$$

so λ_5 must also be positive which means that $X_2 = e^{-2/a}$. The only feasible solution is therefore $X_1 = (1 + e^{-2/a})/2$, $A_2 = 1$ and so P_2^* is the same as the solution given by (7.18). It has been shown that for the case when $n = 2$ the value of the time congestion in a $GI/M/n/n$ queue given by (7.18) is the maximum possible value. □

The proof above is much briefer than the proof for general n given in the appendix although the general steps followed are similar. The above proof will help to investigate the $GI/M/n/n$ queue for the case when $n = 2$ in more detail.

7.5 An Analysis of the Bounds.

An obvious lower bound for the time congestion in the $GI/M/n/n$ queue is 0 for all values of a and n. The results that are obtained using the above method for a lower bound of the $GI/M/n/n$ correspond to this lower bound of 0. Unlike the upper bound, the lower bound is arbitrarily close to a known distribution given by

$$G(.) = \begin{cases} 1-\epsilon, & 0 \le y \le \frac{1}{\epsilon}, \\ 1, & y > \frac{1}{\epsilon} \end{cases}$$

for the case where $\epsilon \to 0$. This distribution corresponds to the case when $X_i = 1$ for all i. Since this bound can be achieved by a known distribution and there is no possible bound lower than this one, the proof of this bound is trivial. This result and the upper bound described above give bounds on the sensitivity of the time congestion in the $GI/M/n/n$ system. Figure 7.1 plots the maximum time congestion in the $GI/M/n/n$ queue versus the arrival rate for different numbers of circuits. Figure 7.2 plots the time congestion in the $GI/M/n/n$ queue versus the number of circuits n for various arrival rates.

Call Congestion and Time Congestion.

Work by Kuczura (1972) has shown that the call congestion in a $GI/M/1/1$ system is minimised when the call arrival distribution is deterministic and conjectured this was also true for cases when there was more than one server. In fact, in Chapter 4 of Takács (1962) it is shown that the call congestion in a $GI/M/n/n$ loss system is given by

$$\left[1 + \sum_{j=1}^{n} \binom{n}{j} \prod_{i=1}^{j} \frac{1-X_i}{X_i}\right]^{-1} \tag{7.27}$$

where X_i is the Laplace-Stieltjes transform of the call inter-arrival distribution calculated at the ith eigenvalue of $Q_1 C^{-1}$. The terms $(1-X_i)/X_i$ are maximised and hence the call congestion is minimised when the X_i's are minimised which is when $X_i = e^{-i/a}$. It is shown in Pearce (1978) that this implies that the inter-arrival lifetime distribution is deterministic, thus proving Kuczura's conjecture. The maximum of the time congestion occurs when $X_1 = e^{-1/a}$ by (7.17) and this also corresponds to the inter-arrival lifetime distribution being deterministic. Thus

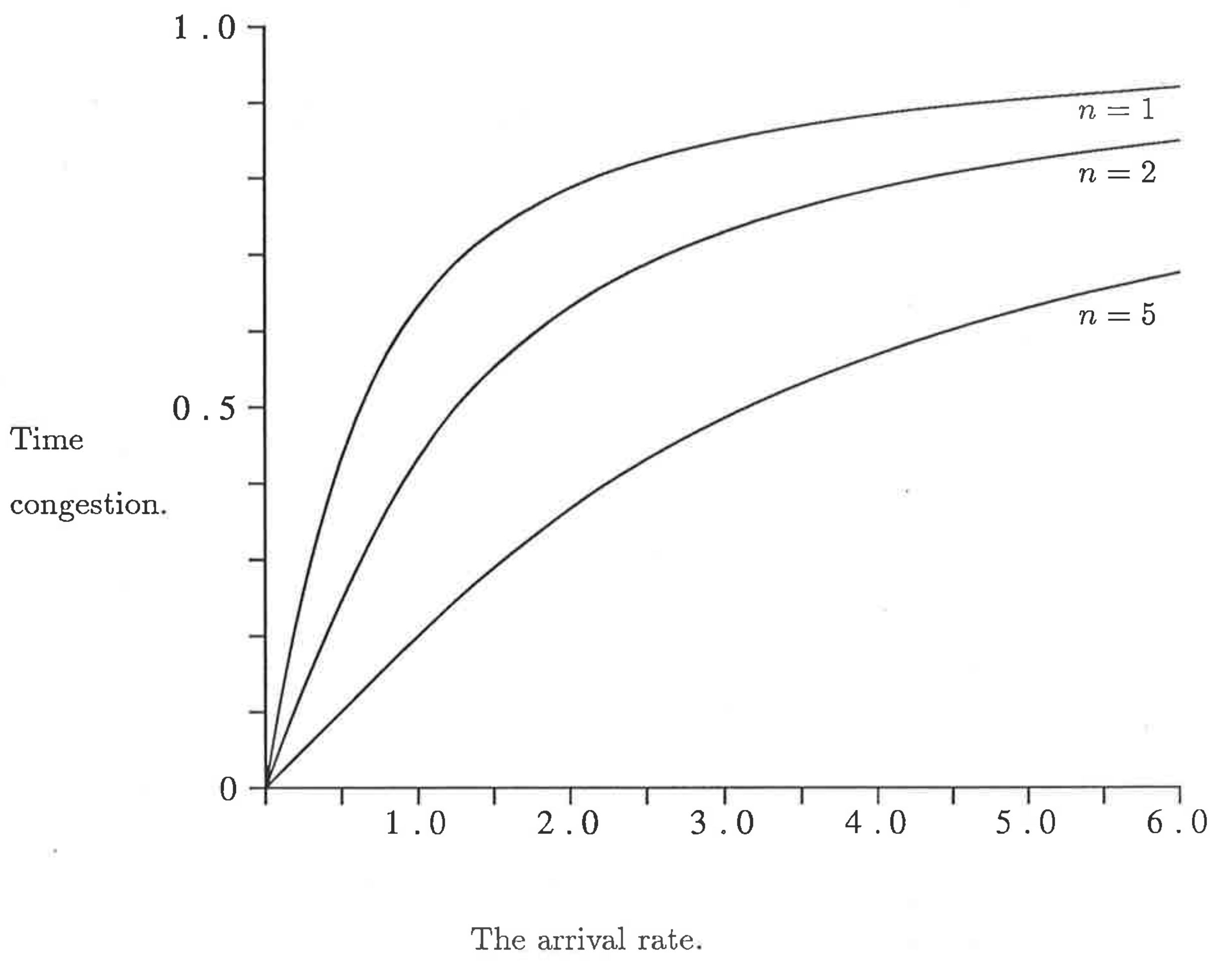

Figure 7.1. *The maximum time congestion for the $GI/M/n/n$ queueing system versus the arrival rate for various values for the number of circuits, n.*

in the single server case the time congestion is maximised when the call congestion is minimised. This occurs when there is a deterministic inter-arrival distribution. However, for situations when there is more than one server this is no longer true.

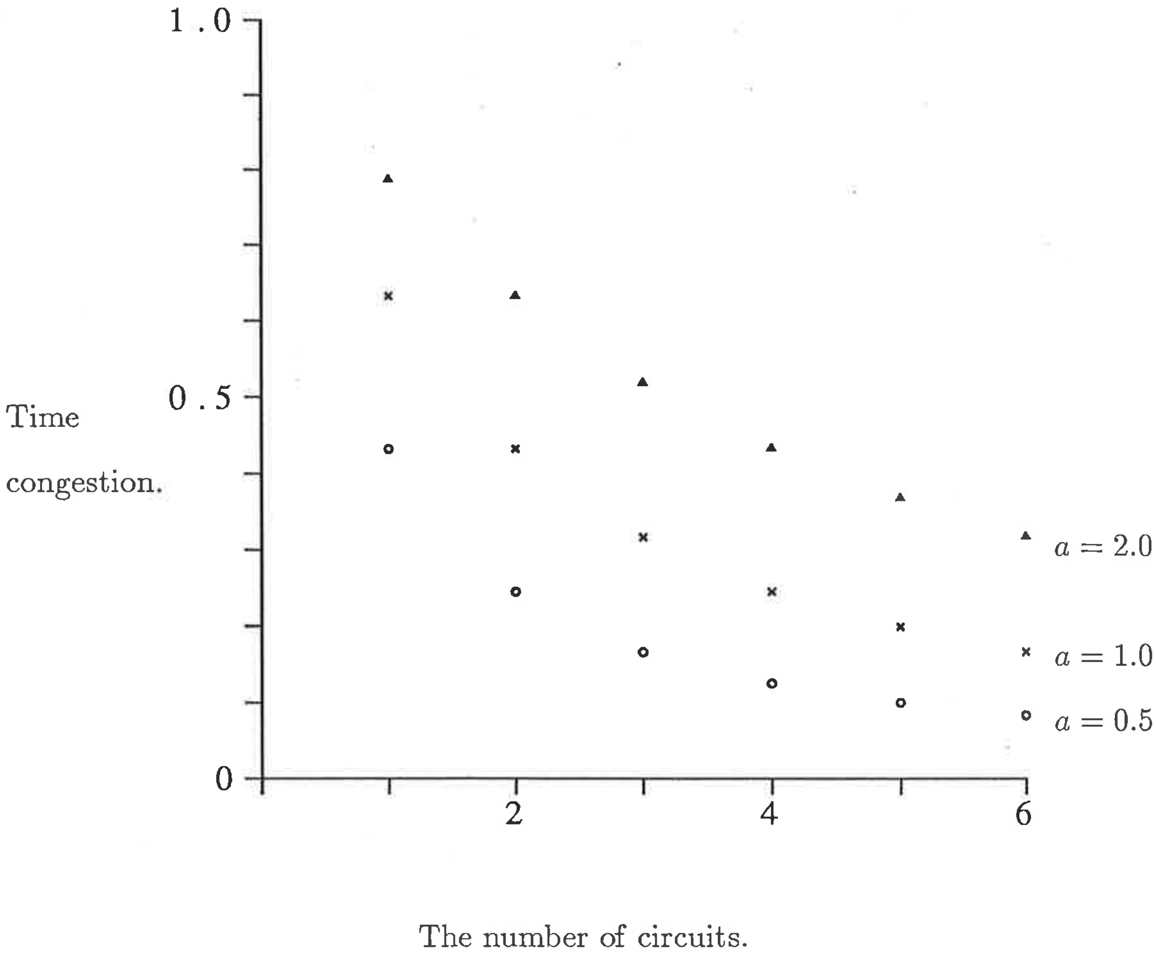

Figure 7.2. *The maximum time congestion for the $GI/M/n/n$ queueing system versus the number of circuits for various values of the arrival rate, a.*

Two servers.

We can investigate various properties of the $GI/M/n/n$ queueing system by using some results of this system that were deduced in the proof of the sensitivity bounds on the time congestion. Some of what follows may be generalised to cases when n is larger than 2, but the mathematics involved becomes very messy as can be appreciated by comparing the proof of theorem 7.1 for the special case when $n = 2$ to the general proof given in the appendix. It has been shown that the time congestion in the $GI/M/2/2$ queueing system is given by

$$\frac{a}{2}\frac{X_1(1-X_2)}{(1-X_1+X_2)}$$

where X_1 and X_2 are the Laplace-Stieltjes Transforms of the inter-arrival time distribution calculated at the eigenvalues of the Q_1C^{-1} matrix which are $-1/a$ and $-2/a$ respectively. This is true no matter what the inter-arrival distribution is. The maximum time congestion was found by maximising the above expression subject to certain constraints on the X_i's. An upper bound on the time congestion was shown to occur when $X_1 = (e^{-2/a}+1)/2$ and $X_2 = e^{-2/a}$. Whilst this is an upper bound on the congestion, these X_i's need not correspond to any actual distribution. In fact, it is known that the only actual distribution such that $X_2 = e^{-2/a}$ is the deterministic distribution, for which $X_1 = e^{-1/a}$, see Pearce (1978).

The time congestion for a deterministic inter-arrival time with rate a is given by

$$\frac{a}{2}\frac{e^{-1/a}(1-e^{-2/a})}{(1-e^{-1/a}+e^{-2/a})}.$$

Initially we thought that the time congestion in the queue when the inter-arrival time is deterministic may in fact be the maximum possible time congestion. If this were true then no matter what the value of a the result proven above is still an upper bound on the time congestion but a better upper bound corresponding to the deterministic arrivals could, at least in theory, be calculated. However, by examining a number of other well known inter-arrival time distributions we have been able to show that this is not true.

The hyperexponential distribution is an interesting distribution to investigate as by making an appropriate choice of the parameters of this distribution we can model a bursty inter-arrival distribution stream in which arrivals tend to occur in batches separated by large gaps. The hyperexponential distribution function with

n separate arrival phases is

$$F(t) = \begin{cases} 0 & t \le 0 \\ \sum_{k=1}^{n} \pi_k (1 - e^{-p_k t}), & t > 0, \end{cases} \tag{7.28a}$$

where

$$\sum_{k=1}^{n} \pi_k = 1.0. \tag{7.28b}$$

The parameters associated with this distribution were discussed in section (6.3)(a) of this thesis. Here p_k is the ratio of the arrival rate at the kth phase a_k to the overall average arrival rate a, i.e. $p_k = a_k/a$. This change is made since an overall arrival rate of a has been specified in the matrix $Q_1 C^{-1}$ and p_k is just the proportion of a that is assigned to the kth arrival phase.

Using (7.28) the Laplace-Stieltjes transforms of the hyperexponential distribution with two separate arrival phases is found to be

$$L(\lambda) = \pi_1 \frac{p_1}{\lambda + p_1} + \pi_2 \frac{p_2}{\lambda + p_2}.$$

So the Laplace-Stieltjes transforms at $1/a$ and $2/a$ are given respectively by

$$X_1 = \pi_1 \frac{a_1}{1 + a_1} + \pi_2 \frac{a_2}{1 + a_2}$$

and

$$X_2 = \pi_1 \frac{a_1}{2 + a_1} + \pi_2 \frac{a_2}{2 + a_2}.$$

By looking at various values of π_1 and a_1 such that

$$a = \frac{a_1 a_2}{a_2 \pi_1 + a_1 \pi_2}$$

is kept constant, different forms of the hyperexponential distribution all with arrival rate a can be investigated. This has been done, the time congestions corresponding to these distributions have been calculated and the value of the parameters which gave the highest value of the time congestion have been used in a comparison with the time congestion calculated when the inter-arrival time is deterministic. In figure 7.3 these values have been plotted along with the upper bound on the time congestion. It can be seen that for $0 \leq a \leq 1$ the time congestion corresponding to the hyperexponential distribution is higher than that for the deterministic distribution with the same average arrival rate a. Neither the hyperexponential distribution nor the deterministic distribution can possibly be used as an upper bound on the time congestion for all a. These are just two of the many possible distributions the inter-arrival distribution may be. It appears that no single actual distribution will have a time congestion greater than the time congestions corresponding to all the other possible distributions for all possible values of the arrival rate. So no single distribution will make the given upper bound redundant for all a.

More than two servers.

It is possible for some cases when there are more than two servers to find an expression for the time congestion in the $GI/M/n/n$ queue. We have shown that for $n \leq 5$ the time congestion in a $GI/M/n/n$ queue is

$$\frac{a}{n}\frac{1-X_n}{X_n}\left[1+\sum_{j=1}^{n}\binom{n}{j}\prod_{i=1}^{j}\frac{1-X_i}{X_i}\right]^{-1} \tag{7.29}$$

but we have not found a proof for general n. It is interesting to compare this expression with the expression for the call congestion given by (7.27).

As was done in the case when there were two servers, the time congestion for the queue can be calculated using (7.29) by replacing the variables in this expression with the values of the Laplace-Stieltjes transforms for the type of distribution being used. The results that we get when we find the time congestions in the queues corresponding to the deterministic inter-arrival distribution and the

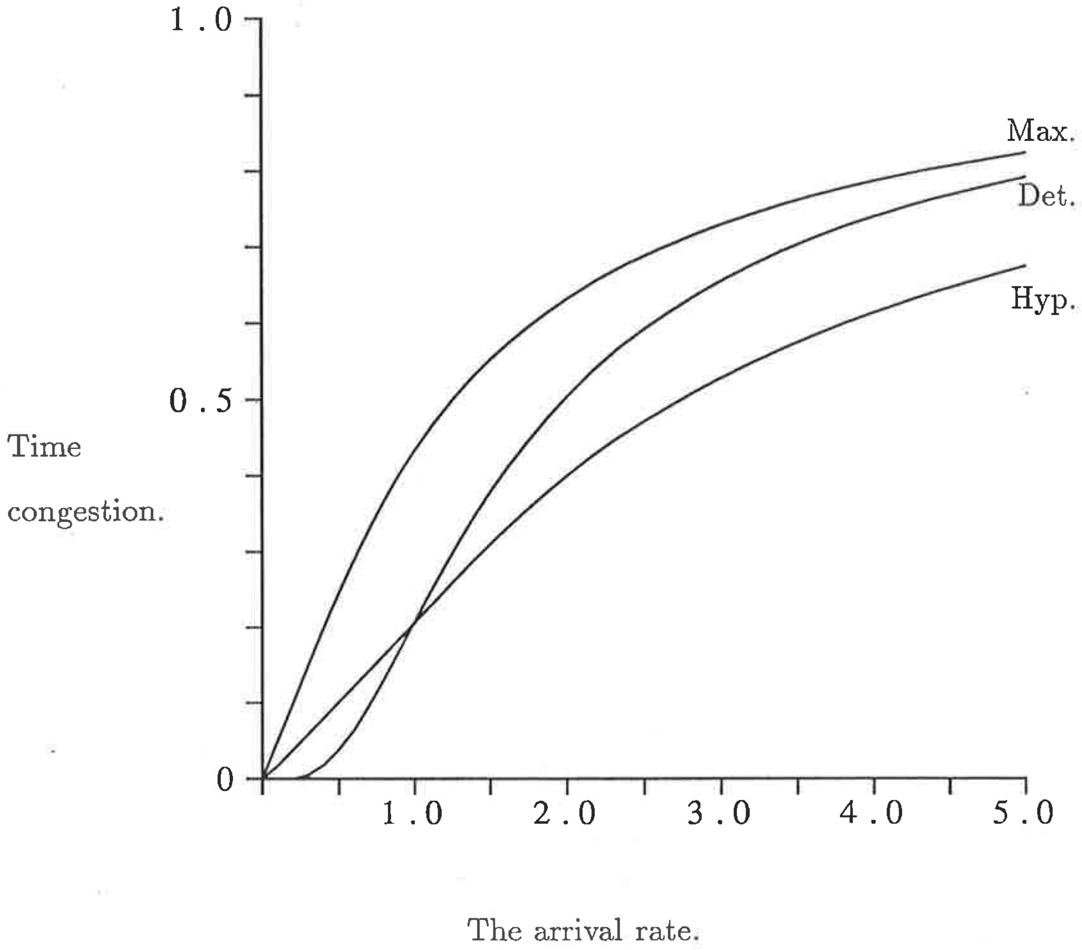

Figure 7.3. *The time congestion for the $GI/M/2/2$ queueing system for various arrival distributions. The postulated upper bound on the time congestion is given as well as the time congestion for the case when the call inter-arrival time distribution is deterministic and the case when the call inter-arrival time distribution is hyperexponential.*

hyperexponential inter-arrival distribution are similar to the results that were obtained for the case when there were two servers. At low arrival rates the time congestion from the hyperexponential distribution is higher than the time congestion from the deterministic distribution. Around about when the arrival rate is one less then the number of circuits the time congestion from the deterministic distribution becomes larger than the time congestion for the hyperexponential distribution. As the arrival rate increases, the time congestion from the deterministic distribution remains larger than the time congestion from the hyperexponential distribution and approaches the upper bound for the time congestion in this queue. Results for the case when there are five servers is shown in figure 7.4.

7.6 Conclusion.

Even though the proof is very messy an upper bound on the time congestion in a $GI/M/n/n$ queue has been obtained. This bound has a very simple form and is easy to use in practice. A comparison of this bound with some simple distributions has been presented.

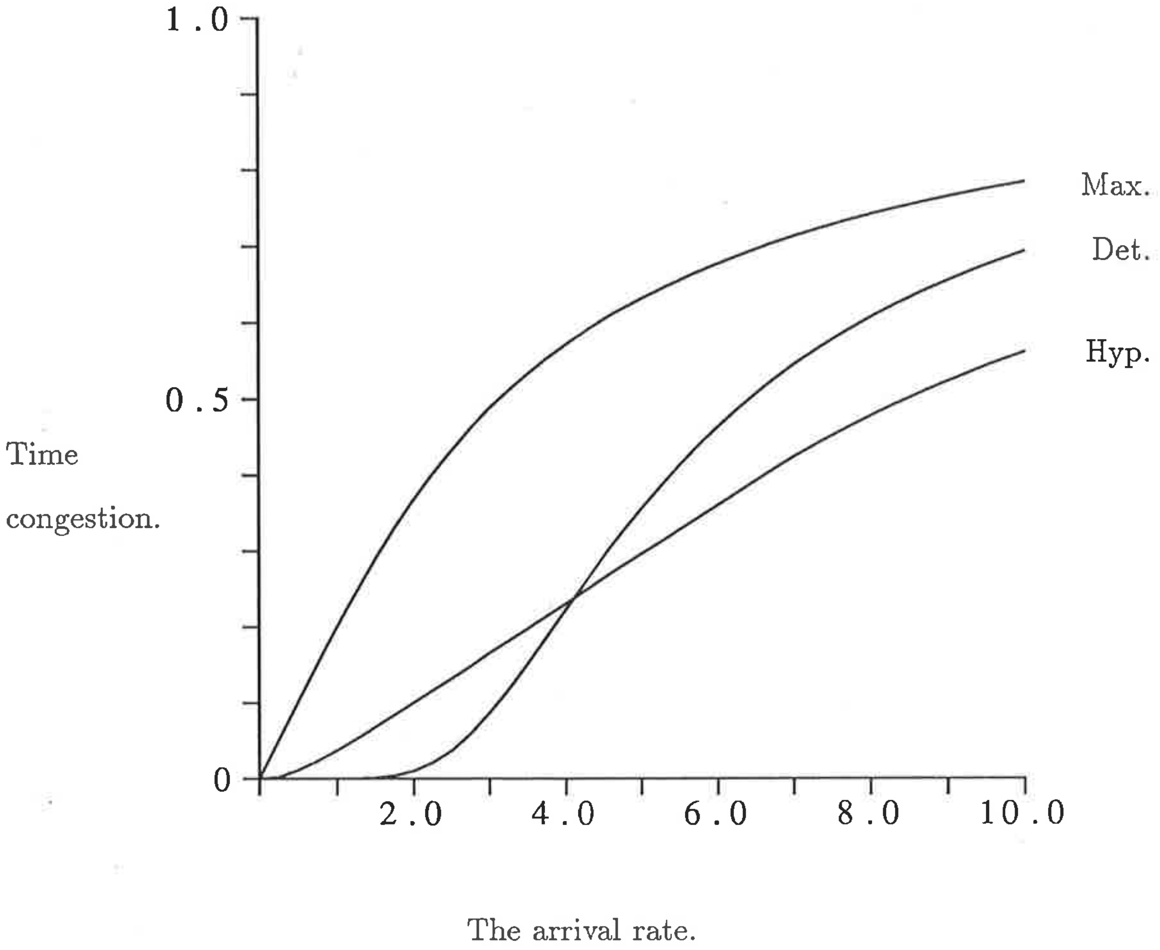

Figure 7.4. *The time congestion for the $GI/M/5/5$ queueing system for various arrival distributions. The postulated upper bound on the time congestion is given as well as the time congestion for the case when the call inter-arrival time distribution is deterministic and the case when the call inter-arrival time distribution is hyperexponential.*

CHAPTER 8
CONCLUSIONS

8.1 Restrictions on the General Distribution.

In Chapter 6, a method for determining bounds on some performance measures in certain simple queueing systems was demonstrated. A non-linear constrained optimisation problem was established which must be solved in order to maximise performance measures of some GSMPs. The constraints that are imposed in this optimisation problem are necessary constraints that all feasible probability distributions must satisfy. There are no constraints imposed on the type of distribution that could satisfy this problem. In a simple extension of this method, the types of distributions that are looked at can be restricted by imposing extra constraints. For example, Eckberg (1977) has shown that any distribution with mean m and variance σ^2 must satisfy

$$e^{-sm} \leq \phi(s) \leq \frac{\sigma^2}{\sigma^2 + m^2} + \frac{m^2}{\sigma^2 + m^2} e^{-s(m+\sigma^2/m)} \tag{8.1}$$

where $\phi(s)$ is the Laplace-Stieltjes transform of the distribution at the point s. Therefore, if we want to impose the further restriction on the general distribution that its variance must be a certain value then the constraint (8.1) will be added to the set of necessary constraints that already exist. This in practice will mean that the constraint (6.17a) will be replaced by (8.1). Note that if the variance is constrained to equal zero then to satisfy (8.1) we must have $\phi(s) = e^{-sm}$ and the distribution must be deterministic. Other examples of constraints corresponding to restrictions on the general distribution are given by Eckberg (1977) and Klincewicz and Whitt (1984).

8.2 Conclusions.

The work presented in this part of the thesis introduced a technique for finding sensitivity bounds on measures of GSMPs in which there is only a single generally distributed lifetime. A number of examples of queueing systems that can be analysed using this technique have been investigated. In all these cases some interesting realistic results were obtained. The $GI/M/n/n$ queue was investigated in detail and an upper bound on the time congestion in this queue obtained. The proof was presented not only to prove this result, but also to try and shed more light on the overall method. The fact that the result obtained is useful and simple is encouraging. However, the long and messy proof of this result unfortunately does not lead to any major insights into how this method can be improved or whether the results obtained are actually global maxima in the general case. Every queueing system investigated that can be checked, has been checked and shown to be a global maximum but this does not mean that this is always the case. A way of showing that the results obtained by this method are indeed global maxima would be very useful.

We have still found this method to be a very useful tool when investigating certain queueing systems. In most cases the numerical results obtained from this method lead to some interesting qualitative results that can be confirmed but that would have been much more difficult to find without the aid of this method. The proof given in this thesis of the upper bound on the time congestion in the $GI/M/n/n$ queue is a case in point. Only after looking at the numerical results obtained from the program written to solve the non-linear program was the form of the time congestion surmised and a proof of this result then established. Many other results of this nature could, no doubt, be found using this technique.

There are many directions of future research that can be undertaken into the above method and using the above method. Any queueing system with a single general server may be investigated numerically, and possibly analytically, using the above method. An investigation into whether, and perhaps under what conditions, the results given by this method are known to be global maximum is of great importance. Further research could also include the creation of methods similar to the above to look at queueing systems with more than one general distribution, to investigate queueing systems in which batch arrivals are possible or to look at systems where the general lifetimes are replaced by increment times on a non-renewal point process, such as a Markov modulated point process. In conclusion, the work in this thesis whilst by itself can act as a useful tool in investigating some queueing systems, can also be taken to be a starting point for much further research on this topic.

APPENDIX.

The Proof of Theorem 7.1.

Preliminary Lemmas.

Some combinatorial identities which are used in the proof of theorem 7.1 shall be given in a series of lemmas. A proof of most of these identities shall be given but some are trivial and can be found in elementary textbooks.

Lemma 1.

$$(1-x)^n = \sum_{i=0}^{n} (-x)^i \binom{n}{i}.$$

Taking $x = 1$ gives

$$\sum_{i=0}^{n} (-1)^i \binom{n}{i} = \begin{cases} 0, & n > 0 \\ 1, & n = 0. \end{cases} \tag{L.1}$$

Lemma 2.

$$\frac{d}{dx}(1-x)^n = \sum_{i=0}^{n} i(-x)^{i-1} \binom{n}{i}.$$

Taking $x = 1$ gives

$$\sum_{i=0}^{n} i(-1)^{i-1} \binom{n}{i} = \begin{cases} 0, & n > 1 \\ 1, & n = 1. \end{cases} \tag{L.2}$$

Lemma 3.

$$\sum_{j=k}^{n} (-1)^j \binom{n}{j} = (-1)^k \binom{n-1}{k-1}, \quad 1 \le k \le n. \tag{L.3}$$

Proof. Case $n = k$ is trivial. Now suppose (L.3) is true for $k = m < n$ then

$$\sum_{j=m-1}^{n} (-1)^j \binom{n}{j} = (-1)^m \binom{n-1}{m-1} + (-1)^{m-1} \binom{n}{m-1}$$
$$= (-1)^{m-1} \left\{ \binom{n}{m-1} - \binom{n-1}{m-1} \right\}$$
$$= (-1)^{m-1} \binom{n-1}{m-2}$$

and so (L.3) is true for $1 \leq k \leq n$. $\square$

Lemma 4.

$$\sum_{i=j}^{k-1} \binom{i}{j} = \binom{k}{j+1}, \quad 1 \leq j+1 \leq k \qquad (L.4)$$

Proof. The case $k - 1 = j$ is trivial. Now suppose that (L.4) is true for $k = m$ then

$$\sum_{i=j}^{m} \binom{i}{j} = \binom{m}{j+1} + \binom{m}{j} = \binom{m+1}{j+1}$$

and so (L.4) is true for $1 \leq j + 1 \leq k$. $\square$

Lemma 5.

$$\sum_{k=j}^{n} (-1)^k \binom{n}{k} \binom{k}{j} = 0, \quad n > j \qquad (L.5)$$

Proof. Now

$$\binom{n}{k} \binom{k}{j} = \binom{n}{j} \binom{n-j}{k-j}$$

and so

$$\sum_{k=j}^{n} (-1)^k \binom{n}{k} \binom{k}{j} = \sum_{k=j}^{n} (-1)^k \binom{n}{j} \binom{n-j}{k-j}$$
$$= \binom{n}{j} \sum_{k=j}^{n} (-1)^k \binom{n-j}{k-j}$$
$$= \binom{n}{j} \sum_{k=0}^{n-j} (-1)^{k+j} \binom{n-j}{k}$$
$$= 0$$

using (L.1) provided $n-j>0$. □

Lemma 6.

$$\sum_{i=j}^{n} i(-1)^i \binom{n-j}{i-j} = 0, \quad n > j+1. \tag{L.6}$$

Proof.

$$\begin{aligned}
\sum_{i=j}^{n} i(-1)^i \binom{n-j}{i-j} &= \sum_{i=0}^{n-j} (i+j)(-1)^{i+j} \binom{n-j}{i} \\
&= (-1)^j \sum_{i=0}^{n-j} i(-1)^i \binom{n-j}{i} + j(-1)^j \sum_{i=0}^{n-j} (-1)^i \binom{n-j}{i} \\
&= 0
\end{aligned}$$

using (L.1) and (L.2) provided $n-j>1$. □

Lemma 7.

$$\sum_{k=j}^{n} (-1)^k k \binom{n}{k} \binom{k}{j} = 0, \quad n > j+1. \tag{L.7}$$

Proof. In the same way as for (L.5)

$$\begin{aligned}
\sum_{k=j}^{n} (-1)^k k \binom{n}{k} \binom{k}{j} &= \sum_{k=j}^{n} (-1)^k k \binom{n}{j} \binom{n-j}{k-j} \\
&= \binom{n}{j} \sum_{k=j}^{n} (-1)^k k \binom{n-j}{k-j} \\
&= 0
\end{aligned}$$

using (L.6) provided $n-j>1$. □

Lemma 8.

$$\sum_{i=j}^{n} i(-1)^i \binom{n-j+1}{i-j} = (n+1)(-1)^n \tag{L.8}$$

Proof. Using (L.6)

$$\sum_{i=j}^{n+1} i(-1)^i \binom{n-j+1}{i-j} = 0$$

for $n - j + 1 > 0$ so

$$\sum_{i=j}^{n} i(-1)^i \binom{n-j+1}{i-j} = (n+1)(-1)^n$$

provided $n > j$. □

Lemma 9.

$$\sum_{i=j}^{n} (-1)^i \binom{n}{i-1}\binom{i}{j} = (-1)^n \binom{n+1}{j} \quad n > j \tag{L.9}$$

Proof.

$$\begin{aligned}
\sum_{i=j}^{n} (-1)^i \binom{n}{i-1}\binom{i}{j} &= \sum_{i=j}^{n} (-1)^i \frac{n!}{(i-1)!(n-i+1)!}\frac{i!}{(i-j)!j!} \\
&= \frac{n!}{j!(n-j+1)!}\sum_{i=j}^{n} i(-1)^i \frac{(n-j+1)!}{(i-j)!(n-i+1)!} \\
&= \frac{n!}{j!(n-j+1)!}\sum_{i=j}^{n} i(-1)^i \binom{n-j+1}{i-j} \\
&= \frac{n!}{j!(n-j+1)!}(n+1)(-1)^n \\
&= (-1)^n \binom{n+1}{j}
\end{aligned}$$

using (L.8) provided $n > j$. □

The Theorem.

Theorem 2 in Chapter 7 proposes that the solution to the following non-linear optimisation problem

$$Max \quad f(\mathbf{x}) = P_n = (-1)^n \frac{A_n}{n/a}\left[1 - X_n\right] \tag{A.1}$$

such that

$$h_j(\mathbf{x}) \equiv \sum_{i=1}^{n} A_i \left[X_i - 1\right](\mathbf{w}_i)_j - A_0(\mathbf{w}_1)_j$$
$$- \sum_{i=1}^{n-1} A_i X_i (\mathbf{w}_{i+1})_j - A_n X_n (\widehat{\mathbf{w}})_j = 0, \; j = 0, ..., n, \tag{A.2}$$

$$h_{n+1}(\mathbf{x}) \equiv A_0 - 1 = 0, \tag{A.3}$$

$$g_1(\mathbf{x}) \equiv X_2 - 2X_1 + 1 \geq 0, \tag{A.4}$$

$$g_i(\mathbf{x}) \equiv X_{i+1} - 2X_i + X_{i-1} \geq 0, \quad i = 2, ..., n-1, \tag{A.5}$$

$$g_n(\mathbf{x}) \equiv 1 - X_1 \geq 0, \tag{A.6}$$

$$g_{i+n-1}(\mathbf{x}) \equiv X_{i-1} - X_i \geq 0, \quad i = 2, ..., n, \tag{A.7}$$

$$g_{i+2n-1}(\mathbf{x}) \equiv X_i - e^{-i/a} \geq 0, \quad i = 1, ..., n \tag{A.8}$$

is given by

$$A_i^* = (-1)^i \binom{n}{i}, \; i = 0, ..., n \tag{A.9}$$

and

$$X_i^* = \left(\frac{e^{-n/a} - 1}{n}\right) i + 1, \quad i = 1, ..., n \tag{A.10}$$

so that

$$\begin{aligned} P_n^* =& (-1)^n \frac{aA_n^*}{n} \left[1 - X_n^*\right] \\ =& \frac{a}{n}(1 - e^{-n/a}). \end{aligned} \tag{A.11}$$

To prove this result it is first necessary to show that the solution given by (A.9) and (A.10) is a feasible solution, that is it satisfies (A.2) to (A.8).

A feasible solution?

Equation (A.2) represents $n+1$ different equations with each one corresponding to a separate element of the eigenvectors. If we look at the jth eigenvector element we shall get the jth equation. Each of these $n+1$ equations must be satisfied by the proposed solution.

Case $j \neq n$. Firstly let us introduce a dummy variable $X_0 = 1$. Then we can write (A.2) as

$$\sum_{i=1}^{n} A_i \left[X_i - 1\right](\mathbf{w}_i)_j - \sum_{i=0}^{n-1} A_i X_i (\mathbf{w}_{i+1})_j - A_n X_n (\widehat{\mathbf{w}})_j = 0$$

$$\Leftrightarrow \sum_{i=1}^{n} A_i \left[X_i - 1\right](\mathbf{w}_i)_j - \sum_{i=1}^{n} A_{i-1} X_{i-1} (\mathbf{w}_i)_j - A_n X_n (\widehat{\mathbf{w}})_j = 0$$

$$\Leftrightarrow \sum_{i=1}^{n} \left\{A_i \left[X_i - 1\right] - A_{i-1} X_{i-1}\right\}(\mathbf{w}_i)_j = A_n X_n (\widehat{\mathbf{w}})_j. \qquad (A.13)$$

Equation (A.13) must be satified by our proposed solution. Now

$$\begin{aligned}
&A_i^* \left[X_i^* - 1\right] - A_{i-1}^* X_{i-1}^* \\
&= (-1)^i \binom{n}{i} \left(\frac{e^{-n/a} - 1}{n}\right) i - (-1)^{i-1} \binom{n}{i-1} \left[\left(\frac{e^{-n/a} - 1}{n}\right)(i-1) + 1\right] \\
&= (-1)^i \left[\frac{n! i (e^{-n/a} - 1)}{(n-i)! i! n} + \frac{n!(i-1)(e^{-n/a} - 1)}{(i-1)!(n-i+1)! n} + \binom{n}{i-1}\right] \\
&= (-1)^i \left[\binom{n-1}{i-1}(e^{-n/a} - 1) + \binom{n-1}{i-2}(e^{-n/a} - 1) + \binom{n}{i-1}\right] \\
&= (-1)^i \left[\binom{n}{i-1}(e^{-n/a} - 1) + \binom{n}{i-1}\right] \\
&= (-1)^i \binom{n}{i-1} e^{-n/a}.
\end{aligned}$$

Putting this in (A.13) we get

$$\sum_{i=1}^{n} (-1)^i \binom{n}{i-1} e^{-n/a} (\mathbf{w}_i)_j = A_n^* X_n^* (\widehat{\mathbf{w}})_j.$$

Since $(\mathbf{w}_i)_j = 0$ when $i < j$ this can be written

$$\sum_{i=j}^{n} (-1)^i \binom{n}{i-1} e^{-n/a} (-1)^j \binom{i}{j} = (-1)^n e^{-n/a} (-1)^j \binom{n+1}{j}$$

$$\Leftrightarrow \sum_{i=j}^{n}(-1)^i \binom{n}{i-1}\binom{i}{j} = (-1)^n \binom{n+1}{j},$$

which is true by (L.9). So (A.2) holds for $j < n$.

Case $j = n$. Only $\mathbf{w}_n$ and $\widehat{\mathbf{w}}$ have a non-zero nth element, therefore (A.2) for this case becomes

$$A_n \left[X_n - 1\right](-1)^n \binom{n}{n} = A_{n-1}X_{n-1}(-1)^n \binom{n}{n} + A_nX_n(-1)^n \binom{n}{n-1}$$

$$\Leftrightarrow A_n \left[X_n - 1\right] = A_{n-1}X_{n-1} + nA_nX_n \tag{A.14}$$

For the proposed solution to be feasible solution we need

$$A_n^* \left[X_n^* - 1\right] = A_{n-1}^* X_{n-1}^* + nA_n^* X_n^*$$

$$\Leftrightarrow \binom{n}{n}\left[(e^{-n/a} - 1)\right] = -\binom{n}{n-1}\left[\left(\frac{e^{-n/a}-1}{n}\right)(n-1) + 1\right] + n\binom{n}{n} e^{-n/a}$$

$$\Leftrightarrow e^{-n/a} - 1 = -n\left[\left(\frac{e^{-n/a}-1}{n}\right)(n-1) + 1\right] + ne^{-n/a}$$

$$\Leftrightarrow e^{-n/a} - 1 = -(e^{-n/a} - 1)(n-1) - n + ne^{-n/a}$$

which is true and so (A.2) is satisfied for the case $j = n$ and therefore for all n.

$A_0^* = 1$ and so (A.3) is obviously satisfied.

The X_i^*'s are monotonically decreasing so (A.7) is satisfied and since the X_i^*'s form a straight line (A.4) and (A.5) are also satisfied.

Now $1 - X_1^* = (1 - e^{-n/a})/n > 0$ since $1 > e^{-n/a}$, so (A.6) holds.

For (A.8) to hold we need,

$$e^{-i/a} \leq \left(\frac{e^{-n/a}-1}{n}\right) i + 1 \quad 0 \leq i \leq n$$

$$\Leftrightarrow \frac{(e^{-i/a}-1)}{i} \leq \frac{(e^{-n/a}-1)}{n} \quad 0 \leq i \leq n.$$

Now

$$\frac{d}{di}\left\{\frac{(e^{-i/a}-1)}{i}\right\} = \frac{1-\frac{a+i}{a}e^{-i/a}}{i^2}$$

which is positive or zero for $1 \leq i \leq n$ and since the above is trivially true for the case when $i = n$ (A.8) must hold.

The solution given by (A.9) and (A.10) has been shown to be a feasible solution.

The Partial Derivatives.

In Chapter 7 necessary and sufficient conditions for a feasible solution of a nonlinear optimisation problem to be a strict local maximum were described. These conditions (7.9) to (7.13) used the method of Lagrange Multipliers. In order to show these conditions are satisfied, the partial derivatives of the Lagrangian must be found. For this we must first find the partial derivatives of the constraints and objective function, given by (A.1) to (A.8), with respect to the variables of this problem. Recall that the eigenvectors of the matrix Q_1C^{-1} are

$$(\mathbf{w}_i)_j = \begin{cases} (-1)^j \binom{i}{j}, & 0 \leq j \leq i \leq n, \\ 0, & \text{otherwise} \end{cases}$$

and the vector $\widehat{\mathbf{w}}$ is given by

$$(\widehat{\mathbf{w}})_k = \begin{cases} (-1)^k \binom{n+1}{k}, & \text{k=0,...,n-1,} \\ (-1)^n \binom{n}{n-1}, & \text{k=n.} \end{cases}$$

Then the partial derivatives of the constraints are:

$$\frac{\partial g_1(\mathrm{x})}{\partial X_i} = \begin{cases} -2, & i = 1, \\ 1, & i = 2, \\ 0, & i = 3, ..., n. \end{cases}$$

$$\frac{\partial g_j(\mathbf{x})}{\partial X_i} = \begin{cases} 1, & i = j-1, \\ -2, & i = j, \\ 1, & i = j+1, \\ 0 & \text{otherwise} \end{cases} \qquad j = 2, ..., n-1.$$

$$\frac{\partial g_n(\mathbf{x})}{\partial X_i} = \begin{cases} 1, & i = 1, \\ 0, & i = 2, ..., n. \end{cases}$$

$$\frac{\partial g_{j+n-1}(\mathbf{x})}{\partial X_i} = \begin{cases} 1, & i = j-1, \\ -1, & i = j, \\ 0, & \text{otherwise} \end{cases} \qquad j = 2, ..., n.$$

$$\frac{\partial g_{j+2n-1}(\mathbf{x})}{\partial X_i} = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases} \qquad j = 1, ..., n.$$

$$\frac{\partial g_j(\mathbf{x})}{\partial A_i} = 0, \quad j = 1, ..., 3n-1, \quad i = 0, ..., n.$$

$$\begin{aligned} \frac{\partial h_j(\mathbf{x})}{\partial A_0} &= -(\mathbf{w}_1)_j, \quad j = 0, ..., n \\ &= \begin{cases} 1, & j = 0, \\ -1, & j = 1, \\ 0, & j = 2, ..., n. \end{cases} \end{aligned}$$

$$\frac{\partial h_0(\mathbf{x})}{\partial A_i} = -1, \quad i = 1, ..., n.$$

$$\begin{aligned} \frac{\partial h_j(\mathbf{x})}{\partial A_i} &= (X_i - 1)(\mathbf{w}_i)_j - X_i(\mathbf{w}_{i+1})_j \\ &= \begin{cases} X_i(-1)^{j+1} \binom{i}{j-1} + (-1)^{j+1} \binom{i}{j} & i = 1, ..., n-1, \quad j \leq i, \\ X_i(-1)^{j+1} & i = 1, ..., n-1, \ j = i+1 \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

$$\frac{\partial h_j(\mathbf{x})}{\partial A_n} = (X_n - 1)(\mathbf{w}_n)_j - X_n(\widehat{\mathbf{w}})_j$$

$$= \begin{cases} X_n(-1)^{j+1}\binom{n}{j-1} + (-1)^{j+1}\binom{n}{j} & j = 1, ..., n-1, \\ (n-1)X_n(-1)^{n+1} + (-1)^{n+1}\binom{n}{n} & j = n. \end{cases}$$

$$\frac{\partial h_j(\mathbf{x})}{\partial X_i} = A_i(\mathbf{w}_i)_j - A_i(\mathbf{w}_{i+1})_j$$

$$= A_i(-1)^{j+1}\binom{i}{j-1} \quad j = 0, ..., n, \quad i = 1, ..., n-1.$$

$$\frac{\partial h_j(\mathbf{x})}{\partial X_n} = A_n(\mathbf{w}_n)_j - A_n(\widehat{\mathbf{w}})_j$$

$$= \begin{cases} A_n(-1)^{j+1}\binom{n}{j-1} & j = 0, ..., n-1, \\ (n-1)A_n(-1)^{n+1} & j = n. \end{cases}$$

$$\frac{\partial h_{n+1}(\mathbf{x})}{\partial A_i} = \begin{cases} 1, & i = 0, \\ 0, & \text{otherwise.} \end{cases}$$

$$\frac{\partial h_{n+1}(\mathbf{x})}{\partial X_i} = 0, \quad i = 1, ..., n.$$

$$\frac{\partial f}{\partial A_i} = \begin{cases} (-1)^{n+1}\frac{X_n - 1}{n/a} & i = n, \\ 0 & i = 0, ..., n-1. \end{cases}$$

$$\frac{\partial f}{\partial X_i} = \begin{cases} (-1)^n \frac{A_n}{n/a} & i = n, \\ 0 & i = 1, ..., n-1. \end{cases}$$

Using the above the Lagrangian can now be calculated.

Some preliminaries

For simplification in the following proof we will define

$$\mathcal{C}_i(\Phi, j) = \sum_{k=i}^{i+j} (-1)^{k-i} \phi_k \binom{j}{k-i}$$

for $0 \leq i \leq n$ and $i + j \leq n$ and

$$\mathcal{C}_1(\Phi, n) = \phi_1 - n\phi_2 + ... + (-1)^{n-2} \binom{n}{n-2} \phi_{n-1} + (-1)^{n-1}(n-1)\phi_n$$

$$= \sum_{k=1}^{n-1} (-1)^{k+1} \phi_k \binom{n}{k+1} + (-1)^{n-1}(n-1)\phi_n.$$

It follows that

$$\mathcal{C}_i(\Phi, j+1) = \mathcal{C}_i(\Phi, j) - \mathcal{C}_{i+1}(\Phi, j)$$

for $0 \leq i \leq n$ and $i + j \leq n$. Also define the $\mathcal{C}_i(\Phi^*, j)$'s in the same way as the $\mathcal{C}_i(\Phi, j)$'s with all the ϕ_k's replaced by ϕ_k^*'s.

The Lagrangian.

The partial derivatives of the Lagrangian given in Chapter 7 by (7.9) are calculated as follows

$$\frac{\partial L}{\partial A_0} = \phi_0 - \phi_1 + \phi_{n+1}$$

$$\begin{aligned}
\frac{\partial L}{\partial A_i} &= \Phi \cdot \frac{\partial \mathbf{h}}{\partial A_i} + \frac{\partial f}{\partial A_i} \\
&= \sum_{j=0}^{i+1} \phi_j \frac{\partial h_j}{\partial A_i} + \frac{\partial f}{\partial A_i} \\
&= -\phi_0 + \phi_1 [X_i + i] - \phi_2 \left[\binom{i}{1} X_i + \binom{i}{2} \right] + ... \\
&+ \phi_i (-1)^{i-1} \left[\binom{i}{i-1} X_i + \binom{i}{i} \right] + (-1)^i \phi_{i+1} X_i \\
&= -\mathcal{C}_0(\Phi, i) + X_i \mathcal{C}_1(\Phi, i)
\end{aligned}$$

for $i = 1, ..., n-1$.

$$\begin{aligned}
\frac{\partial L}{\partial A_n} &= \Phi \cdot \frac{\partial \mathbf{h}}{\partial A_n} + \frac{\partial f}{\partial A_n} \\
&= \sum_{j=0}^{n+1} \phi_j \frac{\partial h_j}{\partial A_n} + \frac{\partial f}{\partial A_n} \\
&= -\phi_0 + \phi_1 [X_n + n] - \phi_2 \left[\binom{n}{1} X_n + \binom{n}{2} \right] + ... \\
&+ \phi_n (-1)^{n-1} [(n-1) X_n + 1] + (-1)^{n+1} \frac{X_n - 1}{n/a} \\
&= -\mathcal{C}_0(\Phi, n) + X_n \mathcal{C}_1(\Phi, n) - (-1)^{n+1} \frac{X_n - 1}{n/a}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial L}{\partial X_i} &= \Phi \cdot \frac{\partial \mathbf{h}}{\partial X_i} + \Lambda \cdot \frac{\partial \mathbf{g}}{\partial X_i} + \frac{\partial f}{\partial X_i} \\
&= \sum_{j=0}^{n+1} \phi_j \frac{\partial h_j}{\partial X_i} + \sum_{j=1}^{3n-1} \lambda_j \frac{\partial g_j}{\partial X_i} + \frac{\partial f}{\partial X_i} \\
&= A_i \left\{ \phi_1 \binom{i}{0} - \phi_2 \binom{i}{1} + ... + \phi_{i+1} (-1)^i \binom{i}{i} \right\} \\
&+ \lambda_{i-1} - 2\lambda_i + \lambda_{i+1} - \lambda_{i+n-1} + \lambda_{i+n} + \lambda_{i+2n-1} \\
&= A_i \mathcal{C}_1(\Phi, i) + \lambda_{i-1} - 2\lambda_i + \lambda_{i+1} - \lambda_{i+n-1} + \lambda_{i+n} + \lambda_{i+2n-1}
\end{aligned}$$

for $i = 2, ..., n-2$.

Similarly

$$\frac{\partial L}{\partial X_1} = A_1 \{\phi_1 - \phi_2\} - 2\lambda_1 + \lambda_2 - \lambda_n + \lambda_{n+1} + \lambda_{2n}$$

$$= A_1 \mathcal{C}_1(\Phi, 1) - 2\lambda_1 + \lambda_2 - \lambda_n + \lambda_{n+1} + \lambda_{2n}$$

$$\frac{\partial L}{\partial X_{n-1}} = A_{n-1} \left\{ \phi_1 \binom{n-1}{0} - \phi_2 \binom{n-1}{1} + ... + (-1)^{n-1} \binom{n-1}{n-1} \phi_n \right\}$$

$$+\lambda_{n-2} - 2\lambda_{n-1} - \lambda_{2n-2} + \lambda_{2n-1} + \lambda_{3n-2}$$

$$= A_{n-1}\mathcal{C}_1(\Phi, n-1) + \lambda_{n-2} - 2\lambda_{n-1} - \lambda_{2n-2} + \lambda_{2n-1} + \lambda_{3n-2}$$

$$\frac{\partial L}{\partial X_n} = A_n\left\{\phi_1 - n\phi_2 + \ldots + (-1)^{n-2}n\phi_n + (-1)^{n-1}(n-1)\phi_n + (-1)^{n+1}\frac{1}{n/a}\right\}$$

$$+\lambda_{n-1} - \lambda_{2n-1} + \lambda_{3n-1}$$

$$= A_n\left\{\mathcal{C}_1(\Phi, n) + (-1)^{n+1}\frac{1}{n/a}\right\} + \lambda_{n-1} - \lambda_{2n-1} + \lambda_{3n-1}.$$

For the proposed solution to be a local maximum of the optimisation problem the partial derivatives of the Lagrangian must all be equal to zero. Therefore using the above we find the following equations that must be satisfied.

$$\frac{\partial L}{\partial A_0} = 0 \Leftrightarrow \phi_0 - \phi_1 + \phi_{n+1} = 0 \tag{A.15}$$

$$\frac{\partial L}{\partial A_i} = 0 \Leftrightarrow \frac{\mathcal{C}_0(\Phi, i)}{\mathcal{C}_1(\Phi, i)} = X_i \tag{A.16}$$

for $i = 1, \ldots, n-1$.

$$\frac{\partial L}{\partial A_n} = 0 \Leftrightarrow \frac{\mathcal{C}_0(\Phi, n)}{\mathcal{C}_1(\Phi, n) - (-1)^{n+1}\frac{X_n - 1}{n/aX_n}} = X_n. \tag{A.17}$$

$$\frac{\partial L}{\partial X_i} = 0$$

$$\Leftrightarrow \lambda_{i-1} - 2\lambda_i + \lambda_{i+1} - \lambda_{i+n-1} + \lambda_{i+n} + \lambda_{i+2n-1} = -A_i\mathcal{C}_1(\Phi, i) \tag{A.18}$$

for $i = 2, \ldots, n-2$.

$$\frac{\partial L}{\partial X_1} = 0$$

$$\Leftrightarrow -2\lambda_1 + \lambda_2 - \lambda_n + \lambda_{n+1} + \lambda_{2n} = -A_1\mathcal{C}_1(\Phi, 1) \qquad (A.19)$$

$$\frac{\partial L}{\partial X_{n-1}} = 0$$

$$\Leftrightarrow \lambda_{n-2} - 2\lambda_{n-1} - \lambda_{2n-2} + \lambda_{2n-1} + \lambda_{3n-2} = -A_{n-1}\mathcal{C}_1(\Phi, n-1) \qquad (A.20)$$

$$\frac{\partial L}{\partial X_n} = 0 \Leftrightarrow \lambda_{n-1} - \lambda_{2n-1} + \lambda_{3n-1} = -A_n\left\{\mathcal{C}_1(\Phi, n) + (-1)^{n+1}\frac{1}{n/a}\right\} \qquad (A.21)$$

So for the proposed solution to be a local maximum of the problem (A.15) to (A.21) must all be satisfied. Note that the above equations have been found independently of the solution that has been proposed. The above equations must be satisfied for all local maxima of this problem.

The Lagrange Multipliers.

To find solutions to the equations (A.15) to (A.21) corresponding to the feasible solution to the non-linear optimisation problem given by (A.9) and (A.10) suitable values for the vectors of Lagrange Multipliers Λ^* and Φ^* must be found. The following values for the Lagrange Multipliers are proposed.

$$\phi_0^* = -P_n^* \left(1 + \sum_{k=1}^{n-1} \frac{1}{X_k^*}\right),$$

$$\phi_j^* = -P_n^* \sum_{k=j}^{n-1} \frac{1}{X_k^*}, \quad j = 1, ..., n-1,$$

$$\phi_n^* = 0,$$

$$\phi_{n+1}^* = P_n^*,$$

$$\lambda_i^* = -\sum_{k=i+1}^{n} (k-i) A_k^* \mathcal{C}_1(\Phi^*, k) \quad i = 1, ..., n-1, \tag{A.22}$$

$$\lambda_n^* = 0,$$

$$\lambda_{i+n-1}^* = 0, \quad i = 2, ..., n$$

$$\lambda_{i+2n-1}^* = 0, \quad i = 1, ..., n-1,$$

and

$$\lambda_{3n-1}^* = (-1)^n \frac{aA_n}{n}.$$

A local maximum?

In this section it will be shown that (A.15) to (A.21) are satisfied by the proposed solution given by (A.9) and (A.10) and the proposed Lagrange Multipliers given by (A.22).

It is easily verified that (A.15) is satisfied by the prescribed solution. To see that (A.16) is satisfied it must be noticed that the X_i^*'s are linear. Introduce X_0^* and let $X_i^* - X_{i+1}^* = \epsilon$ $i = 0, ...n-1$ where $\epsilon = (1 - e^{-n/a})/n$ is positive. Now

$$X_i^* - X_j^* = (j - i)\epsilon.$$

Lemma 10.

$$\mathcal{C}_i(\Phi^*, j) = \frac{P_n^*(-1)^j (j-1)! \epsilon^{j-1}}{\prod_{k=i}^{i+j-1} X_k^*} \tag{A.24}$$

for $j = 1, ..., n-1, \quad i+j \leq n$ and for $j = n, i = 0$.

Proof.

Now

$$\frac{1}{X_i^*} - \frac{1}{X_j^*} = \frac{-(j-i)\epsilon}{X_i^* X_j^*} \qquad i = 0, ..., n, \quad j = 0, ..., n$$

$$\mathcal{C}_i(\Phi^*, 1) = \phi_i^* - \phi_{i+1}^* = \frac{-P_n^*}{X_i^*} \qquad i = 0, ..., n-1$$

so (A.24) is satisfied for $j = 1$. Now providing that $n > 2$

$$\begin{aligned}\mathcal{C}_i(\Phi^*, 2) = \phi_i^* - 2\phi_{i+1}^* + \phi_{i+2}^* &= \frac{-P_n^*}{X_i^*} - \frac{-P_n^*}{X_{i+1}^*} \\ &= P_n^* \frac{\epsilon}{X_i^* X_{i+1}^*}\end{aligned}$$

for $i = 0, ..., n-1$. Suppose

$$\mathcal{C}_i(\Phi^*, j) = \frac{P_n^*(-1)^j (j-1)! \epsilon^{j-1}}{\prod_{k=i}^{i+j-1} X_k^*} \quad i = 0, ..., n-j, \quad j = 1, ..., m$$

then

$$\begin{aligned}\mathcal{C}_i(\Phi^*, m+1) &= \mathcal{C}_i(\Phi^*, m) - \mathcal{C}_{i+1}(\Phi^*, m) \\ &= \frac{P_n^*(-1)^m (m-1)! \epsilon^{m-1}}{\prod_{k=i}^{i+m-1} X_k^*} - \frac{P_n^*(-1)^m (m-1)! \epsilon^{m-1}}{\prod_{k=i+1}^{i+m} X_k^*} \\ &= \frac{P_n^*(-1)^m (m-1)! \epsilon^{m-1}}{\prod_{k=i+1}^{i+m-1} X_k^*} \left[\frac{1}{X_i^*} - \frac{1}{X_{i+m}^*} \right] \\ &= \frac{P_n^*(-1)^m (m-1)! \epsilon^{m-1}}{\prod_{k=i+1}^{i+m-1} X_k^*} \left[\frac{(-1)m\epsilon}{X_i^* X_{i+m}^*} \right] \\ &= \frac{P_n^*(-1)^{m+1} (m)! \epsilon^m}{\prod_{k=i}^{i+m} X_k^*}\end{aligned}$$

and so

$$\mathcal{C}_i(\Phi^*, j) = \frac{P_n^*(-1)^j (j-1)! \epsilon^{j-1}}{\prod_{k=i}^{i+j-1} X_k^*}$$

for $j = 1, ..., n-1$, $i + j \leq n$ and for $j = n, i = 0$. □

From Lemma 10

$$\frac{\mathcal{C}_0(\Phi^*,i)}{\mathcal{C}_1(\Phi^*,i)} = \frac{P_n^*(-1)^i(i-1)!\epsilon^{i-1}}{\prod_{k=0}^{i-1} X_k^*} \frac{\prod_{k=1}^{i} X_k^*}{P_n^*(-1)^i(i-1)!\epsilon^{i-1}}$$
$$= \frac{X_i^*}{X_0^*} = X_i^*$$

and therefore (A.16) is satisfied.

Note that by (A.11)

$$P_n^* = \frac{(1-X_n^*)}{n/a}$$

so

$$\phi_n^* + \frac{X_n^*-1}{n/aX_n^*} = -\frac{P_n^*}{X_n^*}$$

therefore the above process will also work to show that (A.17) is satisfied.

To show that (A.18) to (A.21) are satisfied we should firstly note that many of the λ^*'s are zero. Once we notice this, then by rearranging (A.21), we find we must satisfy

$$\lambda_{n-1}^* + \lambda_{3n-1}^* = -A_n^* \left\{ \mathcal{C}_1(\Phi^*,n) + (-1)^{n+1}\frac{1}{n/a} \right\}.$$

Since $\lambda_{3n-1}^* = (-1)^n \frac{A_n^*}{n/a}$ it is found that for (A.21) to be satisfied we need

$$\lambda_{n-1}^* = -A_n^* \mathcal{C}_1(\Phi^*,n)$$

which is satisfied by the proposed solution so (A.21) is satisfied.

For (A.20) to be satisfied we need

$$\lambda_{n-2}^* = 2\lambda_{n-1}^* - A_{n-1}^*\mathcal{C}_1(\Phi^*,n-1)$$
$$= -2A_n^*\mathcal{C}_1(\Phi^*,n) - A_{n-1}^*\mathcal{C}_1(\Phi^*,n-1)$$
$$= -\sum_{k=n-1}^{n} (k-(n-2))A_k^*\mathcal{C}_1(\Phi^*,k)$$

which is true by (A.22) so (A.20) is satisfied. For (A.18) to be true for the case $i = n-2$ we need

$$
\begin{aligned}
\lambda_{n-3}^* &= 2\lambda_{n-2}^* - \lambda_{n-1}^* - A_{n-2}^*\mathcal{C}_1(\Phi^*, n-2) \\
&= -4A_n^*\mathcal{C}_1(\Phi^*, n) - 2A_{n-1}^*\mathcal{C}_1(\Phi^*, n-1) + A_n^*\mathcal{C}_1(\Phi^*, n) - A_{n-2}^*\mathcal{C}_1(\Phi^*, n-2) \\
&= -3A_n^*\mathcal{C}_1(\Phi^*, n) - 2A_{n-1}^*\mathcal{C}_1(\Phi^*, n-1) - A_{n-2}^*\mathcal{C}_1(\Phi^*, n-2)
\end{aligned}
$$

and continuing in this fashion it is found that for (A.18) to be satisfied we need

$$
\lambda_j^* = -\sum_{k=j+1}^{n} (k-j)A_k^*\mathcal{C}_1(\Phi^*, k)
$$

for $j = 2, ..., n-2$ which is also true by (A.22) so (A.18) is satisfied.

For (A.19) to be satisfied we must have

$$
-2\lambda_1^* + \lambda_2^* = -A_1^*\mathcal{C}_1(\Phi^*, 1)
$$

or equivalently

$$
\begin{aligned}
0 &= 2\lambda_1^* - \lambda_2^* - A_1^*\mathcal{C}_1(\Phi^*, 1) \\
&= -2\sum_{k=2}^{n}(k-1)A_k^*\mathcal{C}_1(\Phi^*, k) + \sum_{k=3}^{n}(k-2)A_k^*\mathcal{C}_1(\Phi^*, k) - A_1^*\mathcal{C}_1(\Phi^*, 1) \\
&= -\sum_{k=2}^{n}(2(k-1) - k + 2)A_k^*\mathcal{C}_1(\Phi^*, k) - A_1^*\mathcal{C}_1(\Phi^*, 1) \\
&= -\sum_{k=1}^{n} kA_k^*\mathcal{C}_1(\Phi^*, k)
\end{aligned}
$$

Now using the fact that $\phi_n^* = 0$

$$
\begin{aligned}
-\sum_{k=1}^{n} kA_k^* C_1(\Phi^*, k) &= -\sum_{k=1}^{n} k(-1)^k \binom{n}{k} C_1(\Phi^*, k) \\
&= -\sum_{k=0}^{n} k(-1)^k \binom{n}{k} \sum_{j=0}^{min(k,n-1)} \binom{k}{j} (-1)^j \phi_{j+1}^* \\
&= -\sum_{k=0}^{n} \sum_{j=0}^{min(k,n-1)} k(-1)^{j+k} \binom{n}{k} \binom{k}{j} \phi_{j+1}^* \\
&= -\sum_{j=0}^{n-1} \sum_{k=j}^{n} k(-1)^{j+k} \binom{n}{k} \binom{k}{j} \phi_{j+1}^* \\
&= -\sum_{j=0}^{n-1} \phi_{j+1}^* (-1)^j \sum_{k=j}^{n} k(-1)^k \binom{n}{k} \binom{k}{j} \\
&= 0
\end{aligned}
$$

by (L.7) and so (A.19) is satisfied. The proposed solution satisfies (A.18) to (A.21).

Note that for every inequality constraint that is not equal to zero the corresponding Lagrange Multiplier is equal to zero and so condition (7.10) is satisfied. It must also be shown that condition (7.11) is satisfied for the proposed solution to be a local maximum, that is all the Lagrange Multipliers corresponding to inequality constraints must be non negative .

The Lagrange Multiplier $\lambda_{3n-1}^* = (-1)^n (aA_n^*/n) = a/n$ is positive so it remains to show that λ_i is positive for $i = 1, ..., n-1$.

Now by (A.22)

$$\begin{aligned}
\lambda_1^* &= -\sum_{k=1}^{n}(k-1)A_k^*\mathcal{C}_1(\Phi^*,k) \\
&= -\sum_{k=1}^{n} kA_k^*\mathcal{C}_1(\Phi^*,k) + \sum_{k=1}^{n} A_k^*\mathcal{C}_1(\Phi^*,k) \\
&= -0 + \sum_{k=1}^{n} A_k^*\mathcal{C}_1(\Phi^*,k) \\
&= \sum_{k=1}^{n}(-1)^k\binom{n}{k}\sum_{j=0}^{k}\binom{k}{j}(-1)^j\phi_{j+1}^* \\
&= \sum_{k=1}^{n}(-1)^k\binom{n}{k}\left(\sum_{j=1}^{k}\binom{k}{j}(-1)^j\phi_{j+1}^* + \binom{k}{0}\phi_1^*\right) \\
&= \sum_{j=1}^{n}\sum_{k=j}^{n}(-1)^k\binom{n}{k}\binom{k}{j}(-1)^j\phi_{j+1}^* + \sum_{k=1}^{n}(-1)^k\binom{n}{k}\phi_1^* \\
&= \sum_{j=1}^{n}(-1)^j\phi_{j+1}^*\sum_{k=j}^{n}(-1)^k\binom{n}{k}\binom{k}{j} + \phi_1^*\sum_{k=1}^{n}(-1)^k\binom{n}{k} \\
&= -\phi_1^*
\end{aligned}$$

using (L.1) and (L.5) and since ϕ_1^* is obviously negative $\lambda_1^* > 0$.

We will introduce a dummy variable ϕ_{n+1}^* such that

$$\phi_n^* - \phi_{n+1}^* = \frac{-P_n^*}{X_n^*}$$

which means that using the same steps as described in the section "A local maximum"

$$\mathcal{C}_1(\Phi^*,n) + (-1)^{n-1}(\phi_n^* - \phi_{n+1}^*) = \phi_1^* - n\phi_2^* + \ldots + (-1)^{n-1}n\phi_n^* + (-1)^n\phi_{n+1}^*$$

$$= \frac{P_n^*(-1)^n(n-1)!\epsilon^{n-1}}{\prod_{k=1}^{n} X_k^*}.$$

Now using the facts that $0 < X_i^* < 1$, $i = 1, ..., n$ and $\epsilon < 1/n$

$$
\begin{aligned}
\lambda_{n-1}^* &= -A_n^* C_1(\Phi^*, n) \\
&= -A_n^* \left\{ C_1(\Phi^*, n) + (-1)^{n-1}(\phi_n^* - \phi_{n+1}^*) + (-1)^n(\phi_n^* - \phi_{n+1}^*) \right\} \\
&= -(-1)^n \left\{ \frac{P_n^*(-1)^n(n-1)!\epsilon^{n-1}}{\prod_{k=1}^n X_k^*} - (-1)^n \frac{P_n^*}{X_n^*} \right\} \\
&= -P_n^* \left\{ \frac{(n-1)!\epsilon^{n-1}}{\prod_{k=1}^n X_k^*} - \frac{1}{X_n^*} \right\} \\
&= P_n^* \left\{ \frac{1}{X_n^*} - \frac{(n-1)!\epsilon^{n-1}}{\prod_{k=1}^n X_k^*} \right\} \\
&= \frac{P_n^*}{X_n^*} \left\{ 1 - \prod_{k=1}^{n-1} \frac{(n-k)\epsilon}{X_k^*} \right\} \\
&= \frac{P_n^*}{X_n^*} \left\{ 1 - \prod_{k=1}^{n-1} \frac{(n-k)\epsilon}{1-k\epsilon} \right\} \\
&> 0
\end{aligned}
$$

and so λ_{n-1}^* is positive.

Also since

$$
\begin{aligned}
\lambda_{i-1}^* - 2\lambda_i^* + \lambda_{i+1}^* &= -A_i^* C_1(\Phi^*, i) \\
&= -(-1)^i \binom{n}{i} \frac{P_n^*(-1)^i(i-1)!\epsilon^{i-1}}{\prod_{k=1}^i X_k^*} \\
&= -\binom{n}{i} \frac{P_n^*(i-1)!\epsilon^{i-1}}{\prod_{k=1}^i X_k^*}
\end{aligned}
$$

is always negative

$$\lambda_{i-1}^* - 2\lambda_i^* + \lambda_{i+1}^* < 0$$

and the λ's form a convex set. Since λ_{n-1} is positive and λ_1 is also positive all the λ's are positive. So the proposed set of Lagrange Multipliers satisfy all the constraints on these multipliers, (7.9) to (7.11), and so the proposed maximum has satisfied the necessary constraints. However, to show that this solution is a strict local maximum of the optimisation problem it is still necessary to show the final conditions (7.12) and (7.13).

Final conditions for local maximum.

It remains to show that (7.12) and (7.13) are satisfied to show that this solution is a local maximum of $f(\mathbf{x})$. The possible values of $\mathbf{z}$ must first be determined. The elements of $\mathbf{z}$ correspond to the variables in the problem so $\mathbf{z}^T = (z_{A_0}, z_{A_1}, ..., z_{A_n}, z_{X_1}, ..., z_{X_n})$. The valid values of z must satisfy (7.13). So if $g_i(\mathbf{x}^*) = 0$ and $\lambda_i^* > 0$ then $\mathbf{z}^T \nabla_x g_i(\mathbf{x}^*) = 0$. This is true for the inequality constraints $g_i(\mathbf{x}), \quad i = 1, ..., n$. So $\mathbf{z}^T \nabla_x g_i(\mathbf{x}^*) = 0, \quad i = 1, ..., n-1$. For the case when $i = 1$ we get

$$\mathbf{z}^T(0, ..., 0, -2, 1, 0, ..., 0)^T = 0 \Leftrightarrow -2z_{X_1} + z_{X_2} = 0.$$

Similarly for the cases $i = 2, ..., n-1$ we find

$$\mathbf{z}^T(0, ..., 0, 1, -2, 1, 0, ..., 0)^T = 0 \Leftrightarrow z_{X_{i-1}} - 2z_{X_i} + z_{X_{i+1}} = 0$$

Also for the case when $i = n$

$$\mathbf{z}^T(0, ..., 0, -1, 0, ..., 0)^T = 0 \Leftrightarrow -z_{X_1} = 0.$$

For all of these conditions to be satisfied we must have $z_{X_i} = 0, \quad i = 1, ..., n$.

The valid values of $\mathbf{z}$ must also satisfy $\mathbf{z}^T \nabla_x h_j(\mathbf{x}^*) = 0, \quad j = 0, ..., n$. Since the elements of $\mathbf{z}$ corresponding to the X_i's are all zero we need only look at the elements of $\mathbf{z}$ corresponding to the A_i's. The case $j = 0$ leads to

$$\mathbf{z}^T(1, -1, ..., -1, 0, ..., 0)^T = 0$$

$$\Leftrightarrow z_{A_0} - z_{A_1} - z_{A_2} - ... - z_{A_n} = 0. \tag{A.25}$$

The case $j = 1$ leads to

$$\mathbf{z}^T(-1, X_1^* + 1, 2X_2^* + 3, ..., X_n^* + n, 0, ..., 0)^T = 0$$

$$\Leftrightarrow -z_{A_0} + (X_1^* + 1)z_{A_1} + (2X_2^* + 3)z_{A_2} + ... + (X_n^* + n)z_{A_n} = 0. \qquad (A.26)$$

The cases $j = 2, .., n-1$ lead to

$$\mathbf{z}^T(0, ., 0, (-1)^{j+1}X_{j-1}^*, .., (-1)^{j+1}\binom{n-1}{j-1}X_{n-1}^* + (-1)^{j+1}\binom{n-1}{j},$$

$$(-1)^{j+1}\binom{n}{j-1}X_n^* + (-1)^{j+1}\binom{n}{j}, 0, ..., 0) = 0$$

$$\Leftrightarrow (-1)^{j+1}X_{j-1}^* z_{A_{j-1}} + ... + (-1)^{j+1}\left\{\binom{n-1}{j-1}X_{n-1}^* + \binom{n-1}{j}\right\} z_{A_{n-1}}$$

$$+(-1)^{j+1}\left\{\binom{n}{j-1}X_n^* + \binom{n}{j}\right\} z_{A_n} = 0. \qquad (A.27)$$

The case $j = n$ leads to

$$\mathbf{z}^T(0, ..., (-1)^{n+1}X_{n-1}^*, (-1)^{n+1}(n-1)X_n^* + (-1)^{n+1}, 0, ..., 0) = 0$$

$$\Leftrightarrow (-1)^{n+1}X_{n-1}^* z_{A_{n-1}} + (-1)^{n+1}((n-1)X_n^* + 1)z_{A_n} = 0. \qquad (A.28)$$

The case $j = n+1$ leads to

$$\mathbf{z}^T(1, 0, ..., 0)^T = 0 \Leftrightarrow z_{A_0} = 0. \qquad (A.29)$$

By adding all the equations given by (A.25) to (A.29) it can be found that $X_n^* z_{A_n} = 0$ which means that $z_{A_n} = 0$ since $X_n^* > 0$. So the only possible way that all of the above equations can be satisfied is if $z_{A_i} = 0, \quad i = 0, ..., n$. Therefore the only $\mathbf{z}$ that satisfies (7.13) is the trivial case $\mathbf{z} = \mathbf{0}$. The condition (7.12) is therefore satisfied for the proposed solution.

Since (7.7) to (7.13) are all satisfied by the proposed solution it has been shown that this solution is a strict local maximum of $f(\mathbf{x})$.

A global maximum?

To show that this local maximum is in fact the global maximum it must be shown that no other local maximum to this problem greater than the local maximum presented in this appendix exists.

Equation (A.16) has been derived without any reference to the possible final solution and so (A.16) must be satisfied by all local maxima. Since all the X_i's are positive we can find some necessary conditions on the $\mathcal{C}_0(\Phi, i)$'s and the $\mathcal{C}_1(\Phi, i)$'s which must be satisfied by any local maximum.

We start by showing that $\mathcal{C}_0(\Phi, 1) = \phi_0 - \phi_1$ cannot be positive. Assume $\mathcal{C}_0(\Phi, 1) > 0$ then by (A.16) for the case when $i = 1$ it is found that $\mathcal{C}_1(\Phi, 1) > 0$ since $X_1 > 0$. Since it is also known that $X_1 < 1$ and using (A.16) when $i = 1$ again, we find $\mathcal{C}_0(\Phi, 1) < \mathcal{C}_1(\Phi, 1)$ which means that $\mathcal{C}_0(\Phi, 2) = \mathcal{C}_0(\Phi, 1) - \mathcal{C}_1(\Phi, 1) < 0$. Continuing in this fashion it can be seen that the $\mathcal{C}_0(\Phi, i)$'s and the $\mathcal{C}_1(\Phi, i)$'s must alternate in sign. Note that this is true if $\mathcal{C}_0(\Phi, 1)$ is positive or negative. Notice also that since

$$\mathcal{C}_j(\Phi, i+1) = \mathcal{C}_j(\Phi, i) - \mathcal{C}_{j+1}(\Phi, i)$$

it is easy to see that the $\mathcal{C}_j(\Phi, i)$'s are all either positive or are all negative for the same j.

For the case when $\mathcal{C}_0(\Phi, 1)$ is positive we must have $(-1)^{i+1}\mathcal{C}_0(\Phi, i)$ also being positive for $i = 1, ..., n$. Also from (A.17)

$$\frac{\mathcal{C}_0(\Phi, n)}{\mathcal{C}_1(\Phi, n) + (-1)^{n+1}\frac{X_n - 1}{n/aX_n}} > 0$$

and so

$$(-1)^{n+1}(\mathcal{C}_1(\Phi, n) + (-1)^{n+1}\frac{X_n - 1}{n/aX_n}) > 0$$

or

$$(-1)^{n+1}\mathcal{C}_1(\Phi, n) > \frac{1-X_n}{n/aX_n}$$

which leads to

$$(-1)^{n+1}\mathcal{C}_1(\Phi, n) > 0.$$

But also notice that

$$\mathcal{C}_{n-1}(\Phi, 1) = \phi_{n-1} - \phi_n > 0$$

and

$$\mathcal{C}_{n-2}(\Phi, 2) = \phi_{n-2} - 2\phi_{n-1} + \phi_n < 0$$

and so

$$\phi_{n-2} - 3\phi_{n-1} + 2\phi_n < 0.$$

Continuing in this fashion we find

$$((-1)^{n+1}\phi_1 + (-1)^n \binom{n}{1} \phi_2 + ... + (n-1)\phi_n) < 0.$$

$$\equiv (-1)^{n+1}\mathcal{C}_1(\Phi, n) < 0$$

and so there is a contradiction. Therefore $\mathcal{C}_1(\Phi, i)$ cannot be positive but must be negative and so $(-1)^i\mathcal{C}_0(\Phi, i)$ must be positive for $i = 1, ..., n$. Using this fact the above contradiction does not exist since we find using the above techniques that

$$(-1)^n\mathcal{C}_1(\Phi, n) > \frac{X_n - 1}{n/aX_n}$$

and

$$(-1)^n\mathcal{C}_1(\Phi, n) < 0 \qquad (A.29a)$$

which are not contradictory conditions since $X_n - 1 < 0$.

We must now investigate whether the A_i's are positive or negative. Firstly note that for a maximum solution $(-1)^n A_n$ must be positive. It has already been shown that a positive local maximum exists and so any local maximum that is negative can be dismissed as it is obviously not a global maximum. From equation (A.14)

$$A_n\left[(1-n)X_n - 1\right] = A_{n-1}X_{n-1}$$

and so since the X_i's are positive and $(-1)^n A_n$ is positive $(-1)^{n-1}A_{n-1}$ must also be positive. Now from (A.13)

$$\sum_{i=1}^{n}\left\{A_i\left[X_i - 1\right] - A_{i-1}X_{i-1}\right\}(\mathbf{w}_i)_j = A_nX_n(\widehat{\mathbf{w}})_j$$

$$\Leftrightarrow \sum_{i=j}^{n}\left\{A_i\left[X_i - 1\right] - A_{i-1}X_{i-1}\right\}(-1)^j\binom{i}{j} = A_nX_n(-1)^j\binom{n+1}{j}$$

so

$$\begin{aligned}
&- A_{j-1}X_{j-1} \\
&= \left[1 - X_j\right]A_j + \sum_{i=j+1}^{n}\left\{A_i\left[1 - X_i\right] + A_{i-1}X_{i-1}\right\}\binom{i}{j} + A_nX_n\binom{n+1}{j} \\
&= \sum_{i=j}^{n} A_i\left[1 - X_i\right]\binom{i}{j} + A_iX_i\binom{i+1}{j} \\
&= \sum_{i=j}^{n} A_iX_i\left\{\binom{i+1}{j} - \binom{i}{j}\right\} + A_i\binom{i}{j} \\
&= \sum_{i=j}^{n} A_iX_i\binom{i}{j-1} + A_i\binom{i}{j}
\end{aligned} \tag{A.30}$$

for $1 \le j < n$. For the case when $j = n$ we get

$$-A_{n-1}X_{n-1} = A_nX_n(n-1) + A_n. \tag{A.31}$$

It can be shown that equations (A.30) and (A.31) are equivalent to

$$A_jX_j = A_{j+1}(X_{j+1} - 1) + (-1)^{n-j}\binom{n}{j}A_nX_n \quad 0 \le j \le n \tag{A.32}$$

which will yield the result we are looking for. To prove this first note that for $j < n$

$$\begin{aligned}
&A_j X_j \\
=&A_{j+1}X_{j+1} - A_{j+1} + (-1)^{n-j}\binom{n}{j} A_n X_n \\
=&A_{j+2}X_{j+2} - A_{j+1} - A_{j+2} \\
&+ (-1)^{n-j}\binom{n}{j} A_n X_n + (-1)^{n-j+1}\binom{n}{j+1} A_n X_n \\
=&A_n X_n - \sum_{k=j+1}^{n} A_k + \sum_{k=j+1}^{n} (-1)^{n-k+1}\binom{n}{k-1} A_n X_n \\
=&-\sum_{k=j+1}^{n} A_k + (-1)^n A_n X_n \sum_{k=j+1}^{n} (-1)^{k+1}\binom{n}{k-1} + A_n X_n \\
=&-\sum_{k=j+1}^{n} A_k + (-1)^n A_n X_n \left\{(-1)^j \binom{n-1}{j-1} - (-1)^n\right\} + A_n X_n
\end{aligned}$$

using (L.3) which means that

$$A_j X_j = -\sum_{k=j+1}^{n} A_k + (-1)^n A_n X_n (-1)^j \binom{n-1}{j-1} \quad j < n \tag{A.33}$$

which will now be shown to be equivalent to the above equations (A.30) and (A.31). In other words if we want to show that equations (A.30) and (A.31) are satisfied then it is sufficient to show that (A.33) is satisfied.

When $j = n-1$ in (A.33) we get

$$A_{n-1}X_{n-1} = -A_n - A_n X_n (n-1)$$

which corresponds to (A.31). Now using (A.30) and (A.31) from $n-1$ down to j

and using (L.4) and (L.9) we get

$$
\begin{aligned}
&A_{j-1}X_{j-1}\\
&= -\left\{\sum_{i=j}^{n} A_iX_i \binom{i}{j-1} + A_i\binom{i}{j}\right\}\\
&= -\sum_{i=j}^{n-1}\left\{\binom{i}{j-1}\left[-\sum_{k=i+1}^{n} A_k + A_nX_n(-1)^n(-1)^i\binom{n-1}{i-1}\right]\right\}\\
&\quad -\sum_{i=j}^{n} A_i\binom{i}{j} - A_nX_n\binom{n}{j-1}\\
&= \sum_{k=j+1}^{n} A_k \sum_{i=j}^{k-1}\binom{i}{j-1} - A_nX_n(-1)^n\sum_{i=j}^{n-1}(-1)^i\binom{n-1}{i-1}\binom{i}{j-1}\\
&\quad -\sum_{i=j}^{n} A_i\binom{i}{j} - A_nX_n\binom{n}{j-1}\\
&= \sum_{k=j+1}^{n} A_k\left\{\sum_{i=j-1}^{k-1}\binom{i}{j-1} - 1 - \binom{k}{j}\right\} - A_j - A_nX_n\binom{n}{j-1}\\
&\quad - A_nX_n(-1)^n\left\{\sum_{i=j-1}^{n-1}\left[(-1)^i\binom{n-1}{i-1}\binom{i}{j-1}\right] - (-1)^{j-1}\binom{n-1}{j-2}\right\}\\
&= \sum_{k=j+1}^{n} A_k\left\{\binom{k}{j} - 1 - \binom{k}{j}\right\} - A_j - A_nX_n\binom{n}{j-1}\\
&\quad - A_nX_n(-1)^n\left\{(-1)^{n-1}\binom{n}{j-1} - (-1)^{j-1}\binom{n-1}{j-2}\right\}\\
&= -\sum_{k=j}^{n} A_k + A_nX_n(-1)^n(-1)^{j-1}\binom{n-1}{j-2}
\end{aligned}
$$

and so (A.33) is equivalent to the equations (A.30) and (A.31). The equations (A.32) must be satisfied for a local maximum and from this result we know that the A_j's must alternate in sign with $(-1)^nA_n$ being positive.

So, in general, $(-1)^iA_i$ must always be positive for the global maximum of this problem. From (A.18) we have

$$\lambda_{i-1} - 2\lambda_i + \lambda_{i+1} - \lambda_{i+n-1} + \lambda_{i+n} + \lambda_{i+2n-1} = -A_i\mathcal{C}_1(\Phi, i)$$

and so

$$\lambda_{i-1} - 2\lambda_i + \lambda_{i+1} - \lambda_{i+n-1} + \lambda_{i+n} + \lambda_{i+2n-1} = -(-1)^i A_i (-1)^i C_1(\Phi, i).$$

Since the right hand side must be negative

$$\lambda_{i-1} - 2\lambda_i + \lambda_{i+1} - \lambda_{i+n-1} + \lambda_{i+n} + \lambda_{i+2n-1} < 0. \tag{A.34}$$

In a similar fashion we find from (A.19) and (A.20)

$$-2\lambda_1 + \lambda_2 - \lambda_n + \lambda_{n+1} + \lambda_{2n} < 0 \tag{A.35}$$

$$\lambda_{n-2} - 2\lambda_{n-1} - \lambda_{2n-2} + \lambda_{2n-1} + \lambda_{3n-2} < 0 \tag{A.36}$$

and from (A.21) and (A.29a) we find that

$$A_n \left\{ C_1(\Phi, n) + (-1)^{n+1} \frac{1}{n/a} \right\} + \lambda_{n-1} - \lambda_{2n-1} + \lambda_{3n-1} = 0$$

$$\Rightarrow \lambda_{n-1} - \lambda_{2n-1} + \lambda_{3n-1} > 0 \tag{A.37}$$

Firstly we notice that either λ_1 or λ_n or both must be positive from (A.35). If λ_n is positive then $g_n(\mathbf{x})$ must be equal to zero or $X_1 = 1$. This means that all the X_i's must be equal to 1 since $g_i(\mathbf{x})$ must be satisfied for $i = 1, ..., n-1$. If $X_n = 1$ then the time congestion will be zero and since we have already shown that a positive feasible solution exists this cannot be a global maximum to the problem. We cannot have $\lambda_n > 0$ therefore $\lambda_n = 0$ and $\lambda_1 > 0$. So for a global maximum to this problem we must have $X_2 - 2X_1 + 1 = 0$.

We must also have by (A.34) with $i = 2$

$$-\lambda_1 + 2\lambda_2 - \lambda_3 + \lambda_{n+1} - \lambda_{n+2} - \lambda_{2n+1} > 0$$

and so either λ_2 or λ_{n+1} must be positive. If λ_{n+1} is positive then $X_1 = X_2$ which is impossible as it will contradict the fact that $X_2 - 2X_1 + 1 = 0$ and $X_1 \neq 1$. Hence we must have $\lambda_2 > 0$ and so $X_1 - 2X_2 + X_3 = 0$. Continuing in this way we find that the inequalities $g_1(\mathbf{x})$ to $g_{n-1}(\mathbf{x})$ must all be satisfied for a local maximum that is not zero or negative.

For $g_1(\mathbf{x})$ to $g_{n-1}(\mathbf{x})$ to all be satisfied the X_i's must all lie on a line which goes through 1 for $i = 0$. The value of X_n must also lie between $e^{-n/a}$ and 1. So the X_i's must be of the form $(X_n - 1)i/n + 1$.

It must now be shown that

$$A_k = (-1)^{n-k} \binom{n}{k} A_n. \tag{A.38}$$

For the case when $k = n$ this is obviously true. Let's also suppose that it is true for all $n \geq k > j$ then using (A.32)

$$\begin{aligned}
A_j X_j &= A_{j+1}(X_{j+1} - 1) + (-1)^{n-j} \binom{n}{j} A_n X_n \\
&= (-1)^{n-j+1} \binom{n}{j+1} A_n (X_n - 1)\frac{j+1}{n} + (-1)^{n-j} \binom{n}{j} A_n X_n \\
&= (-1)^{n-j+1} \binom{n}{j} A_n \left\{ (X_n - 1)\frac{j+1}{n}\frac{n-j}{j+1} - X_n \right\} \\
&= (-1)^{n-j+1} \binom{n}{j} A_n \left\{ (X_n - 1)\frac{n-j}{n} - X_n \right\} \\
&= (-1)^{n-j} \binom{n}{j} A_n \left\{ (X_n - 1)\frac{j}{n} + 1 \right\} \\
&= (-1)^{n-j} \binom{n}{j} A_n X_j
\end{aligned}$$

and so (A.38) has been shown to be true.

Now from (A.12) when $j = 0$

$$\begin{aligned}
1 &= -\sum_{i=1}^{n} A_i \\
&= -A_n \sum_{i=1}^{n} (-1)^{n-i} \binom{n}{i} \\
&= (-1)^{n+1} A_n \left\{ \sum_{i=0}^{n} (-1)^i \binom{n}{i} - 1 \right\} \\
&= (-1)^n A_n
\end{aligned}$$

and so the time congestion which is given by $(-1)^n \frac{a}{n} A_n (1 - X_n)$ is equivalent to $(1 - X_n)\frac{a}{n}$ which is maximised when X_n is minimised that is when $X_n = e^{-n/a}$ which is the postulated result. $\square$

REFERENCES

Ash G.R., Cardwell R.D. and Murray R.P. (1981) - *Design and optimisation of networks with dynamic routing*, B.S.T.J., **60**, 1787–1820.

Avriel M. (1976) - *Nonlinear programming. Analysis and Methods*, Prentice-Hall, New Jersey.

Baskett F., Chandy K., Muntz R. and Palacios J. (1975) - *Open, Closed and Mixed Networks of Queues with Different Classes of Customers*, J. Assoc. Comp. Mach., **22**, 248–260.

Bretshneider G. (1973) - *Extension of the equivalent random method for smooth traffics*, Proceedings, I.T.C. 7, Stockholm, paper no. 411.

Beneš V.E. (1961) - *The Covariance Function of a Simple Trunk Group, with applications to Traffic Measurement.*, B.S.T.J., **40**, 117–148.

Berry L.T.M. (1971) - *A mathematical model for optimizing telephone networks*, Ph.D thesis, University of Adelaide, Dec. 1971.

Berry L.T.M. and Coyle, A.J. (1984) - *Performance models for a class of non-hierarchical networks*, A.T.R., **18**, 33–38.

Brockmeyer E., Halström H.A. and Jensen A. (1948) - *The Life and Works of A.K. Erlang*, Trans. 2. Copenhagen:Danish Academy of Technical Sciences, 1–277.

Burman D. (1981) - *Insensitivity in Queueing Systems*, Adv. Appl. Prob., **13**, 846–859.

Cameron W.H. (1981) - *Simulation of dynamic routing : Critical path selection features for service and economy*, Proceedings, Int. Conf. on Commun., 55.5.1-55.5.6.

Chandy K., Howard J. and Towsley D. (1977) - *Product Form and Local Balance in Queueing Networks*, J. Assoc. Comp. Mach., **24**, 250–263.

Chandy K. and Martin J. (1983) - *A Characterisation of Product Form Queuing Networks*, J. Assoc. Comp. Mach., **30**, 286–299.

Coyle A.J. (1986) - *Dimensioning Non-Hierarchical Telephone Networks.*, Australian Telecom Reports.

Coyle A. (1989) - *Sensitivity Bounds in a $GI/M/n/n$ system*, to appear in the J. Austral. Math. Soc. Ser. B.

Eckberg A.E. Jr. (1977) -*Sharp bounds on Laplace-Stieltjes transforms, with applications to various queuing problems*, Mathematics of Operations Research, **2**, 135–142.

Feller W. (1966) - *An Introduction to Probability Theory and its Applications Vol II*, John Wiley, New York.

Franken P., Arndt U., König D. and Schmidt V. (1982) - *Queues and Point Processes*, John Wiley, London.

Garbin D.A. and Knepley, J.E. (1981) - *Marginal cost routing in non-hierarchical networks*, Proceedings, Int. Conf. on Commun., 55.1.1-55.1.4.

Gaudreau M. (1980) - *Recursive formulas for the calculation of point-to-point congestion*, IEEE Trans. Commun., **COM-28**, No. 3, 153–164.

Girard A. and Ouimet Y. (1983) - *End to end blocking for circuit-switched networks : Polynomial algorithms for some special cases*, IEEE Trans. Commun., **COM-31**, 1269–1272.

Harris R.J. and Helm A.V. (1976) - *Distribution of overflow traffics*, Traffic Engineering News Sheet, **12**, Australian Telecom.

Henderson W. (1983) - *Insensitivity and Reversed Markov Processes*, Adv. Appl. Prob., **15**, 752–768.

Hordijk and Ridder (1988) - *Insensitive Bounds for the Stationary Distribution of Non-Reversible Markov Chains*, J. Appl. Prob., **25**, 9–20.

Hordijk and van Dijk (1983a) - *Networks of Queues*, Proc. International Seminar on Modelling and Performance Evaluation Methodology, INRIA, **1**, 79–135.

Hordijk and van Dijk (1983b) - *Adjoint Processes, Job Local Balance and Insensitivity for Stochastic Networks*, Bull. 44th Session Int. Stat. Inst., **50**, 776–788.

Jessop C.W.A. and McLeod N.W. (1983) - *Thoughts on a new telephone network structure for Australia*, Telecom. Journal of Aust., **33**, 67–71.

Kelly F.P. (1976) - *Networks of Queues*, Adv. Appl. Prob., **8**, 416–432.

Kelly F.P. (1979) - *Reversibility and Stochastic Networks*, John Wiley, London.

Kelly F.P. (1986) - *Blocking probability in Large Circuit-Switched networks*, Adv. Appl. Prob., **18**, 473–505.

Klincewicz and Whitt (1984) - *On approximations for Queues, II: Shape Constraints,* AT&T Bell Lab. Tech. J., **63**, 139–161.

König D. and Jansen U. (1974) - *Stochastic Processes and Properties of Invariance for Queueing Systems with Speeds and Temporary Interruptions*, Trans. 7th Prague Conference Inf. Th., Stat. Dec. Fns. and Rand. Proc., 335–343.

Kreĭn and Nudel'man. (1977) - *The Markov Moment Problem and Extremal Problems*, American Mathematical Society, Rhode Island.

Kuczura A. (1972) - *Queues with Mixed Renewal and Poisson Inputs*, Bell System Tech. J., **51**, 1305 – 1326.

Kuczura A. (1973) - *Piecewise Markov Processes*, SIAM J. Appl. Math., **24**, 2, 169 – 181.

Kuczura A. and Bajaj D. (1977) - *A method of moments for the analysis of a switched communication network's performance*, IEEE Trans. Commun., **COM-25**, 185–193.

Lin P.M., Leon B.J. and Stewart C.R. (1978) - *Analysis of circuit-switched networks employing originating-office control with spill-forward*, IEEE Trans. Commun., **COM-26**, 754–765.

Miyazawa M. (1988) - *The Characterization of the Stationary Distributions of the Supplemented Self-Clocking Jump Process*, Preprint.

Narendra K.S. and Mars P. (1981) - *A study of telephone traffic routing using learning algorithms*, Proceedings, Int. Conf. on Commun., 55.4.1–55.4.5.

Pearce C.E.M. (1978) - *On the Peakedness of Primary and Secondary Processes*, Australian Telecommunications Research, **12**, 2, 18–24.

Pioro M.P. (1981) - *A method for optimizing non-hierarchical circuit-switched communication networks with alternate routing*, Paper from Instytut Komunikacji.

Ridder A.A.N. (1987) - *Stochastic Inequalities for Queues*, Ph.D. Thesis, University of Leyden.

Schassberger R. (1977) - *Insensitivity of Steady-State Distributions of Generalized Semi-Markov Processes. Part I.*, Ann. Probability, **5**, 87–89.

Schassberger R. (1978a) - *Insensitivity of Steady-State Distributions of Generalized Semi-Markov Processes. Part II.*, Ann. Probability, **6**, 85–93.

Schassberger R. (1978b) - *Insensitivity of Steady-State Distributions of Generalised Semi-Markov Processes with Speeds*, Adv. Appl. Prob., **10**, 836–851.

Schassberger R. (1985) - *Two Remarks on Insensitive Stochastic Models*, Reihe Mathematik, Technische Universität Berlin.

Schramel F.J. (1982) - *Trends in digital switching and ISDN*, Telecommunication Journal, **49**, 421–429.

Tacács (1962) - *Introduction to the theory of Queues*, Oxford University Press, New York.

Taylor P.G. (1987) - *Aspects of Insensitivity in Stochastic Processes*, Ph.D. Thesis, The University of Adelaide.

Taylor P.G. (1988) - *Insensitivity in Processes with Zero Speeds*, to appear.

Taylor P.G. and Henderson W. (1988) - *Congestion in Models for Bursty Arrivals*, Teletraffic Research Centre, Dept. of Applied Mathematics, The University of Adelaide, Report No. 30/88.

Truitt C.J. (1954) - *Traffic engineering techniques for determining trunk requirements in alternate routing trunk networks*, BSTJ, **33**, 277–302.

van Dijk N.M. (1987) - *Simple and Insensitive Bounds for a Grading and an Overflow Model*, Ops. Res. Letters, **6**, 2, 73–76.

van Dijk N.M., Walrand J. and Tsoucas P. (1987) - *Simple Bounds for the Call Congestion of Finite Multi-Server Delay Systems*, Memorandum NR. 606, Department of Applied Mathematics, Twente University of Technology.

Whitt W. (1984) - *On approximations for Queues, I : Extremal Distributions*, AT&T Bell Lab. Tech. J., **63**, 115–138.

Whittle P. (1985) - *Partial Balance and Insensitivity*, J. Appl. Prob., **22**, 168–176.

Whittle P. (1986) - *Partial Balance, Insensitivity and Weak Coupling*, J. Appl. Prob., **18**, 706–723.

Wilkinson R.I. (1956) - *Theories for toll traffic engineering in the U.S.A.*, B.S.T.J., **35**, 421–514.