NETWORK DESIGN

FOR

STOCHASTIC TRAFFIC

by

David W. McLimont

A thesis submitted to the Faculty of Science
in partial fulfilment of the requirements for the
degree of Master of Science

Department of Electrical Engineering
Faculty of Pure and Applied Science
The University of Ottawa
Ottawa, Canada

1966

ABSTRACT

Network design, in this thesis, is the assignment of branch capacities for the case of stochastic traffic, exemplified by a telephone communication network. Blocking as defined by the Erlang loss formula is chosen as a measure of performance. The assignment is governed by doctrines which achieve either near-equal blocking, as low as possible, on all branches, or the lowest weighted average blocking in the network, subject to fixed network capacity. The assignment is also constrained by the requirement that the branch capacities be integers.

A partitioning process is developed which assigns branch capacities meeting these conditions. Some examples illustrate the application of the doctrines and partitioning to several species of network configurations. This method gives control over the performance of the network and avoids treating stochastic traffic as a finite set of simultaneous flows sampled at different instants in time.

ACKNOWLEDGEMENTS

It is a pleasure to express my gratitude to Professor G.S. Glinski, Chairman of the Department of Electrical Engineering, for his direction and encouragement in the research for this thesis. I wish also to acknowledge the many helpful discussions with memebers of the Faculty.

I am indebted to the Northern Electric Company for assistance, and to the members of the staff who offered valuable and helpful criticism. Finally, it is a privilege to have been able to study the writings and works of many authors who have explored the properties of communication networks and contributed to the literature on them.

TABLE OF CONTENTS

CHAPTER I

NETWORK PHILOSOPHY

1.1 Introduction

A communication network is a system of points at which messages are originated, relayed, or terminated, and of channels interconnecting the points and over which the messages travel. Many have been led to the study of communication networks by an intuitive feeling for their behaviour; others have been obliged to cope with them as a matter of expedience. As a result, some of the problem areas of communication networks have become apparent, and remarkable contributions towards the understanding of the behaviour of communication networks have been made in a relatively short period. Although the problems are yielding under the pressure of attacks from both theoretical and practical quarters, the behaviour is not yet completely understood; it is clear that the computational operations involved are often staggering and that the day of unification is not yet in sight.

The challenge of these problems has stimulated continuing activity in the investigation of the behaviour of networks. However, many of the steps towards the goal of understanding network operation have used models which are not realistically representative of communication networks. In the construction of actual communication networks, the final design is seldom, if ever, ideal because of the need to allow for safety factors, economic considerations, political complications, and use of mathematical approximations.

In the area of network design, one of the broad systems engineering objectives is to provide satisfactory service at lowest cost. The systems engineer is here concerned with the deployment of equipment rather than with the manner in which it operates, and he observes the functional performance of the network in order to evaluate his designs.

Practical solutions are desirable for engineering purposes; but one can find so many practical constraints that the general problem often loses its generality and becomes specific. It is our intention to examine some aspects of the behaviour of telephone communication networks with the objective of finding certain solutions to the problem of network design for networks carrying stochastic traffic, subject to fixed network capacity. The method can be applied equally to a network with linear costs, subject to fixed network cost. A solution will give us an assignment of branch capacities which we will call optimal under carefully defined conditions. In this work we are seeking general cases which resemble reality and which will enable comparisons of performance to be made rapidly and simply. If we can build a model which is general and which yields meaningful solutions, we will have made further headway into this complex subject.[1,2]

Network philosophy is discussed in Chapter 1, and the subject of the work is placed in a practical setting.

The design problem is discussed and defined in Chapter 2. Chapter 3 describes the matrix representation which will be used. The method for achieving optimal assignment of branch capacities is developed in Chapter 4, and the essential doctrines are laid down. The method and the doctrines are extended to networks in Chapter 5. Chapter 6 is devoted to some possible ways of handling networks in which tandem traffic is carried, while Chapter 7 illustrates some additional applications. Future lines of investigation are suggested in Chapter 8.

The crux of the work is to be found in Chapter 4, and it opens up some exciting possiblities in solving some of the problems of network synthesis.

1.2 The Setting

In the physical world, a communication network is an arrangement of various forms of transmitting, receiving and switching equipment, having mutual means of interconnection. We will deal in particular with a telephone communication network, and hence the equipment consists of telephones and telephone offices. Some types of equipment other than telephones do use the telephone network, of course. We will be concerned with those which make use of the network in the same way as telephones do; that is to say, which have characteristics that enable them to be equated with telephones in their use of the network; but we will refer to them collectively as "telephones" for simplicity. This equipment is situated at the nodes of the network.

The nodes are interconnected by branches consisting of paths. A path is the entity which can carry a single message. Messages are the network commodity. A path can carry only one message at a time. It is occupied when it is being used to carry a message, otherwise it is idle. The path, as we have defined it, is the unit of branch capacity. The capacity of a branch is, then, the number of paths that are in it, which is the same as the largest number of messages which can be carried simultaneously in that branch. Messages are commonly transmitted over a variety of wire and wireless channels, which we shall call "paths" without distinction as to type. In carrier and time division channels, for example, the frequency band or time slot which carries a single message counts as one path.

A telephone message is known as a call. The informational content of the call is of no immediate concern. A telephone call is characterized by the instant in time when the system is engaged to carry it, and by the duration of the call. It is also characterized by its points of origin and termination.

Network theory gives the impression that a commodity flows along the branches from node to node. We should be quite clear at the outset that telephone calls do not flow. The call either exists or does not exist; either the path is occupied or it is not; but the call does not flow. It is convenient, nevertheless, to take advantage of the sense of directionality which is inherent in network theory, because every call has to be originated somewhere, and requires connection with a point somewhere else. We can give a direction to an occupied path, from the point of origin to the point of termination of the call. The important thing to note is that a path for a call is secured at the point of origin. A directed branch, therefore, signifies that paths in it can be secured by calls arising only at the origin of the branch, and nowhere else.

A path, by implication so far, is a direct connection between two nodes. When a call originates and terminates at two nodes which are not directly connected by a path, it has to be relayed via one or more different nodes which can offer it a set of contiguous paths which it can occupy simultaneously. Nodes used to relay a call in this way are called tandem nodes. We will distinguish between a direct call, which occupies one direct path between the points of origin and termination, and a tandem call, which occupies two or more consecutive paths for its duration.

Tandem calls, like direct calls, do not flow. However, the directed branches of the network are useful in indicating the sequence in which the successive paths of the tandem call are secured, starting from the origin. If we liken this concept of securing paths to a form of control, then we could say that the control flows through the network, although the calls do not.

This control embodies the very function of the telephone communication network, which is to select and provide idle paths for the transmission of telephone calls offered to the system. The control elements of the telephone system are contained in everything from the telephone and the switching office to the electronic data processor and the microwave carrier system. We will think of control activity as residing in the nodes, where paths are secured for the calls.

1.3 Network Function

Something of the control task, and of the network problem we are going to tackle, can be appreciated if one imagines for a moment a small telephone network in which no relaying of calls is permitted, Figure 1(a). Clearly the total number of calls which can be handled simultaneously in the system is equal to the total number of paths in the system. If, in a stated interval of time, all the calls arising could just be packed end to end in time, and exactly fully occupy each network path for the time interval, then the system would be occupied to capacity. We observe that calls arising in excess of that number simply cannot be accpeted.

Let us create a tandem node in this imaginary network, Figure 1(b). We will designate three of the interconnected nodes A, B, and C. If we cut one, and only one, path between A and C, and set up a new path between A and B, we now have exactly the same number of paths in the network as before. At B, let us join the new path between A and B with any one of the existing paths between B and C, and let us send the same calls over the two joined paths which had previously been sent over the path now cut.

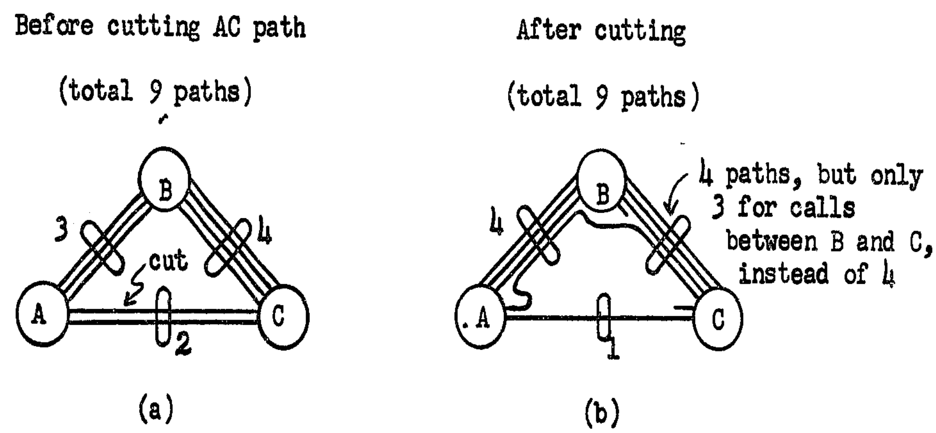


Figure 1

The result of this maneuver is that we have deprived the system of its ability to carry the calls which were formerly carried on the path which we have just expropriated between B and C. This illustrates the fact that relaying calls reduces the capacity of the system; or, alternatively, when the same number of calls as before is offered in the given time interval, relaying causes some of them not to be accepted.

So far, in these two examples, we have assumed that all the calls will pack perfectly. If, however, any one of the calls in our examples should arrive a moment sooner, it would not find a space, that is, an idle path, and it would not be accepted. And if it should arrive a moment later, then its successor, upon arrival, would find the path occupied, and so it would not be accepted. In general, if the calls were to arrive at random in the stated interval of time, although at the same rate of arrival, it would be unlikely that they would adopt an arrangement whereby they would pack perfectly, and so unaccepted calls are bound to arise on this account.

Here we have the situation where a system whose sole purpose in life is to provide paths for telephone calls apparently cannot do so. Naturally we are bound to take steps to put this situation right. A deliberate reduction in the call arrival rate would probably find fewer calls unaccepted; it follows that any attempt to lower the number this way effectively reduces the capacity of the system. Conversely, by starting out with more paths we might be able to handle the original calling rate with fewer calls unaccepted, but it would be at the price of the larger number of paths instead of the reduced capacity.

A communication network is, in fact, usually branch capacity limited. It has the characteristic that when a message is offered to it, it may or may not have an idle path to provide at that instant. The statistics of offering messages to a group of paths have long been known;[3] the best that can be done in order to handle a reasonable calling rate on an economical number of paths is to suffer a small quantity of unaccepted calls. To eliminate this quantity would be, in a commercial system, intolerably expensive.

The telephone system, by nature, operates in real time. If the message is a telephone call, either it is accepted, or it finds no idle path available and is not accepted. Calls not accepted when they are offered are called lost. If the lost call is cleared from the system instantly, the call is said to be blocked.[4] In this paper we deal with just such a lost calls cleared system, because it is, by and large, fairly straightforward.

Here we have the situation where a system whose sole purpose in life is to provide paths for telephone calls apparently cannot do so. Naturally we are bound to take steps to put this situation right. A deliberate reduction in the call arrival rate would probably find fewer calls unaccepted; it follows that any attempt to lower the number this way effectively reduces the capacity of the system. Conversely, by starting out with more paths we might be able to handle the original calling rate with fewer calls unaccepted, but it would be at the price of the larger number of paths instead of the reduced capacity.

A communication network is, in fact, usually branch capacity limited. It has the characteristic that when a message is offered to it, it may or may not have an idle path to provide at that instant. The statistics of offering messages to a group of paths have long been known;[3] the best that can be done in order to handle a reasonable calling rate on an economical number of paths is to suffer a small quantity of unaccepted calls. To eliminate this quantity would be, in a commercial system, intolerably expensive.

The telephone system, by nature, operates in real time. If the message is a telephone call, either it is accepted, or it finds no idle path available and is not accepted. Calls not accepted when they are offered are called lost. If the lost call is cleared from the system instantly, the call is said to be blocked.[4] In this paper we deal with just such a lost calls cleared system, because it is, by and large, fairly straightforward.

Lost calls may, alternatively, be held for a certain length of time, or delayed until an idle path is obtained, but consideration can be given to these at another time.

It is this essential characteristic of the network, whether or not it can provide a required idle path, which furnishes us with a measure of performance. Blocked calls occur when more simultaneous calls are offered to the network than there are paths available to carry the calls. In our telephone network, the measure of performance is defined as the probability that a call will be blocked, which is to say, the probability of not finding a path for a call at the instant that it is offered to the system. The probability of not finding a path is termed blocking.

The measure of performance enables us not only to evaluate network design, but to go further and seek optimal solutions to the problems of network design. When blocking is used in this connection, minimum blocking is the tacit objective, subject to whatever economic or physical constraints may be applied. As we have implied earlier, blocking is undesirable because it is inherently inconsistent with the functional objective of the system.

1.4 Network Commodity

Let us turn our attention to calls, for a moment. It is a characteristic of human beings that, over any short interval of time, they appear collectively to generate calls at random.

The qualification of the time interval is necessary because humans are creatures of habit, and generate their calls also with marked diurnal and other periodic and long-term variations in rate. In practice we are chiefly concerned with the busiest time interval, or busy period, which is when calls are offered at the highest rate, because this period, (which recurs daily with remarkably consistent calling rates), produces the highest blocking experienced by the system. At any other time, the blocking is less. During the busy period, the arrivals of calls approach a condition of statistical equilibrium; hence the random process is approximately stationary. If we assume that it is stationary, then we can represent the arrivals of the offered calls by the Poisson distribution, having for its parameter the average rate at which calls are generated in that period. Of course, when we know that we are dealing with a system in which the calls are not offered at random, then we must use some other appropriate distribution; but otherwise, the assumption of randomness is realistic, if not true.

Now, while it is possible to imagine that independent individuals would request service in an unrelated fashion, it is rather more difficult to make an intuitive estimate of the durations of the calls they make. One might guess that short calls are more frequent than long calls. It happens that call durations, or call holding times, as they are called, are approximated by a negative exponential distribution. This distribution, which embodies our guess, is convenient, because it requires only one parameter. The simplest thing we can measure about calls is their length, and hence their average length, which gives us the required parameter.

The negative exponential distribution applies to ordinary telephone call holding times. Other holding time distributions can occur, as, for example, when time limits are applied, when a non-human device controls the length of the call, or when some other bias exists. If we know these distributions we can use them, but for the general case of the telephone system, with its variety of uses, the holding times of all calls, taken collectively, tend to follow a negative exponential distribution.

We can now define the load on the telephone system, or on any part of it. The holding time of a random call is not in any obvious way related to the interval between it and any other call offered to the system, and hence the distributions of call arrivals and of holding times are assumed to be independent. The product of the parameters of these two distributions, therefore, gives the average amount of traffic offered to the system, or part of it, and this we call the load. Since the call arrivals and holding times are random quantities, telephone traffic is stochastic.

(Average call holding time is such a relatively constant quantity that variations in traffic quantities can usually be construed as variations in average calling rates of the various traffic sources, or nodes. In fact, the calling rate may almost be regarded as the functional parameter of the network.)

We shall use traffic as defined above for the functional parameter, for reasons of accuracy and convenience, in our lost calls cleared system. The statistics of traffic in such a system have long been known, and are expressed in the well-known Erlang loss formula,[3,5] as follows:

$$B(c,a) = \frac{\frac{a^c}{c!}}{\sum_{x=0}^{c} \frac{a^x}{x!}}$$

where: a = traffic offered, in Erlangs,

c = number of paths to which the traffic a is offered,

B(c,a) = B = blocking when a is offered to c; or probability that when the traffic is a, when one more call is offered it will find all the c paths occupied and so will not be accepted.

This Erlang loss formula is also called the Erlang B relationship. A portion of a table of the Erlang loss formula is given in Appendix I.

1.5 Routing Doctrine

We have already referred to relaying calls via tandem nodes; in real life this may be due to physical difficulty in placing the paths, or due to unfavourable economics, and it is customary to use paths via another node or nodes to provide the required connection. When, like this, a network is not fully connected, a routing doctrine is drawn up to specify the sequence of nodes between which paths will be used for tandem calls.

A routing doctrine is simply a set of rules which defines the relaying procedure.

The routing doctrine has another function. When calls between a pair of nodes are permitted to travel over more than one route, the routing doctrine specifies the tandem nodes for each route and also the order in which the several routes are to be tried when a call is offered.

When more than one route between a pair of nodes is possible, the doctrine is called alternate routing; if only one route is possible, it is called fixed routing. Fixed routing is the special case of alternate routing when only one of the possible routes is allowed. Both fixed and alternate routing may include routing direct calls over direct paths. Also, part of the alternate routing doctrine may include fixed routing.

1.6 Summary

Now we are ready to review our telephone communication network model and the assumptions used in it. The network is a set of nodes inter-connected by branches consisting of paths which permit calls to be made between node pairs. The function of the network is to provide idle paths for calls. The path or paths with which a call is provided are governed by a routing doctrine, which specifies also the choices of routes if more than one is possible. A call is characterized by the instant of its arrival into the network, its holding time, and the nodes where it originates and terminates. Branches in the network are directed to show which node has control over the paths to handle offered calls.

The term network is used in the broadest sense, so as to include any possible topological structure and nodal representation. A communication network can range from fully connected, to a single path structure; and nodes can range from a single telephone on a single path, to a switching center which concentrates the network for a large number of telephones, and to a switching center which serves solely to relay calls and neither originates nor terminates any.

The problem we have set out to solve deals with that important duty of interconnecting the switching centres of a nation (or the like) into a national telephone communication network. Accordingly, we will assume that our nodes consist of either switching centres which are points of origin and termination for the telephones of the community which they serve, or tandem nodes, or both.

Assumptions used in our model are:

(a) calls are handled in real time,

(b) network is branch capacity limited,

(c) directed branches,

(d) lost calls cleared system,

(e) no switching delay,

(f) single destination for calls.

These assumptions give a reasonably realistic representation of a working telephone communication system. The first four have been thoroughly discussed, and enable us to relate the function of the system with its characteristic in performing the function.

Although the establishment of a connection from origin to termination is not achieved instantaneously, it must, in effect, be established in an instant of time when the system displays a certain state. And since that instant is random with respect to the system state, the assumption of no switching delay is acceptable. To be sure, the processing of a call to make the desired connection does take perceptible time, and this time does take up some of the capacity of the system, but it should be, and is, relatively small, and it is getting less as techniques for processing calls become faster. Again, if the connection is a tandem one and needs several steps to set it up, the assumption of randomness of call arrivals means that we can regard the several successive instants of decision on availability of idle paths as coincident, from a stochastic point of view.

The lost calls cleared system simply means that an offered call which is not accepted does not become a responsibility of the system, and that the person whose call is lost must, if he wishes, make a new attempt. In practice, the lost calls cleared system is found to be a very good approximation for alternate routing.

Three more assumptions describe the load on our network model, inasmuch as they relate to time:

(a). Poisson call arrivals, i.e., negative exponential distribution of time intervals between call arrivals, have been thoroughly discussed, and enable us to relate the function of the system with its characteristic in performing the function.

Although the establishment of a connection from origin to termination is not achieved instantaneously, it must, in effect, be established in an instant of time when the system displays a certain state. And since that instant is random with respect to the system state, the assumption of no switching delay is acceptable. To be sure, the processing of a call to make the desired connection does take perceptible time, and this time does take up some of the capacity of the system, but it should be, and is, relatively small, and it is getting less as techniques for processing calls become faster. Again, if the connection is a tandem one and needs several steps to set it up, the assumption of randomness of call arrivals means that we can regard the several successive instants of decision on availability of idle paths as coincident, from a stochastic point of view.

The lost calls cleared system simply means that an offered call which is not accepted does not become a responsibility of the system, and that the person whose call is lost must, if he wishes, make a new attempt. In practice, the lost calls cleared system is found to be a very good approximation for alternate routing.

Three more assumptions describe the load on our network model, inasmuch as they relate to time:

(a). Poisson call arrivals, i.e., negative exponential distribution of time intervals between call arrivals,

(b) negative exponential distribution of call holding times,

(c) stationarity.

The two assumed distributions conform reasonably well to observation and experience, although some departure from reality at the extremes is undeniable. For instance, if the near-infinite time between arrivals ever did chance to occur, a node might cease to be an originator of calls for years! Nevertheless, these assumptions give us a way of creating a dynamic and workable model. The use of these assumptions, naturally, does not preclude the use of other distributions where they are applicable, and so does not detract from the generality of this approach, except that the mathematics would become much more complex.

The condition of stationarity is essential to the application of these assumptions, and, as has been stated, this is an acceptable condition for short periods of operation. When we are concerned with the design of a system to carry a range of loads, we normally design it to handle the maximum expected load and call this its load capacity. Therefore, of all the possible values of traffic which the telephone network might experience for short intervals, we choose as the design parameter the highest average traffic with which we are concerned, and associate it with the condition of stationarity.

CHAPTER II

THE DESIGN PROBLEM

2.1 Considerations

The design element in the network, as we have described it, is the branch. The branch capacity is a variable, and it is the magnitude of this variable which it is the designer's task to specify. The unit of branch capacity, the path, is discrete, and we know that we are faced with obtaining integral values for any solutions which we get.

The network commodity is traffic. Traffic and branch capacity are functionally related by blocking. Blocking is a measure of performance, and it is a very significant measure of service. It can be related in practice very closely to the telephone subscribers' opinions of the grade of service the network is rendering them.

The subscribers offer traffic; the telephone companies furnish the service of carrying the traffic; the service has to be satisfactory. What is to prevent us from providing an unlimited number of paths? Surely the problem is an economic one, one of how much we can afford to spend on paths weighed against the cost of giving unsatisfactory service. Of course, the cost of complaints is usually very difficult to assess, so the customary procedure is to engineer the system to meet but not exceed some specified standard "grade of service", or blocking, at minimum cost.

The whole subject of network costs is a complicated one, since it is dependent on branch lengths, transmission methods, node switching equipment, cost of laying cable, relay towers, and so on, and none of these are necessarily constant throughout an entire system. We can make two assumptions which are helpful:

(i) the cost of switching equipment prorated at the terminals of a path is proportional to the cost of the path,

(ii) the cost of a branch is directly proportional of the number of paths in the branch.

These enable us to formulate the cost function W, defined as:

$$W = \sum_k y_k c_k$$

where c_k is the number of paths in the kth branch, $k = 1, 2, \ldots, m$, in a network with m branches, and y_k is the cost of one path and associated terminal switching equipment in the kth branch.

There is a further simplification that we can make in order to demonstrate the methods of solution which follow: we can set the y_k's equal to unity, and hence

$$\underset{(y_k=1)}{W} = \sum_k c_k = C$$

where C is defined as the total branch capacity, or network capacity.

This simplication allows us to avoid setting up models for path costs; in effect, we continue to operate with the network parameters only, without losing any of the features of the model. (When the network capacity methods are worked out, we can apply weighted branch capacities, where the weights are the path costs, for solutions with more economic realism.)

The assumption that all paths in the network cost the same has some real counterpart in life. It is approached in certain cases of microwave carrier equipment, where the bulk of the channel and switching equipment is concentrated at the switching points, and the cost of a path is largely independent of length and route, provided that no relaying is required or that the cost of relay equipment is small compared to the cost of the terminal equipment.

2.2 Problem Definition

The network design problem which we consider here is the problem of assigning branch capacities so as to minimize blocking, with given traffic, in a given topological structure, under control of a routing doctrine, subject to a fixed network capacity. That is,

Minimize $\bar{\bar{B}} = f(B_k)$, with respect to traffic, branch capacity assignment, topological structure, and routing doctrine, subject to $C = \sum_k c_k$,

where $f(B_k)$ is a function of blocking defined by a particular doctrine.

We state the problem this way because if we were simply to minimize total branch capacity we would have no network; and if we were to minimize blocking we would only need to provide such a large number of paths that one would be available for every pair of telephones in the network, if all were simultaneously occupied in every possible combination. We therefore relate blocking and total branch capacity for given traffic; and minimize one (blocking) while holding the other (capacity) fixed, where we use the term minimize to mean as low as possible, subject to the availability of a finite number of paths. We call the resulting branch capacity assignment the optimal solution.

Much of the work on the movement of commodities in networks has dealt with steady flow. Static maximal flow problems and minimal cost flow problems in capacity limited networks have been investigated by Ford and Fulkerson, Dantzig, and others.[6,7,8] Multiterminal network flows have been described by Gomory, Hu, and others.[9,10] The telephone communication network, however, falls into the class of multicommodity network flow problems. The term flow is a mathematical one, and occupancy would be a better term for telephone communication networks. However, there are many features of network theory which make it useful to retain the association with flow networks while we are investigating telephone network behaviour.

Works in the area of multicommodity network flow are not plentiful,[11,12] and still fewer when it comes to stochastic flows.[13] One which appeared while this paper was being written was the subject of a doctoral thesis and dealt with telegraph networks.[14] The stochastic nature of telegraph messages and telephone calls make steady flow treatment inapplicable. The telegraph system, however, differs from the telephone system in that it is characterized by the queuing of messages at the switching points. The performance of the telephone system would be quite unsatisfactory if blocking of calls occurred to an extent equivalent to the queuing of the telegraph system, even if the blocked calls were put through when paths became available after a delay. Wing and Kleinrock give excellent assessments and bibliographies.[1,14]

2.3 Routing Doctrine

The routing doctrine has not been clearly defined in many single and multiterminal pair network solutions. Usually the flow between terminal pairs has been assumed to use any nodes as tandem nodes at the convenience of the solution. In the telephone network, relaying can only be done by certain specified nodes, and often only according to a preference scheme. By contriving a suitable model, we will be in a position to apply and evaluate routing doctrines. The mechanics of routing will be discussed, as required, later.

2.4 Changes in Load

One of the usual assumptions made in network problems is that the busy load is coincident; that is to say, all the indicated loads at each node apply at the same time, and each indicated load is the maximum one. A considerable increase in complexity is involved if we imagine an over-riding time function relating all the load levels at each node, such that some are increasing, some decreasing, some are at maximum, and some at minimum. Nevertheless, non-coincidence of busiest traffic loads in telephone networks is commonly experienced. It may happen in small scale networks, where the busiest period for metropolitan type nodes is during the morning while the busiest time for nodes in the residential suburbs is often during the evening.

The situation is of greater consequence in national networks, on account of the time zone effect; as the earth rotates, busy periods are created across the length and breath of the network which are anything but coincident. Now, we do not propose to solve this problem, but we do want to show that the task of accommodating the time zone effect is not as formidable as it might seem, if we have a reasonable and easily workable model.

Herein lies one of the important reasons for using directed branches in our model: the traffic into and out of a node might not even have coincident busy periods.

From the engineering point of view, it is interesting to observe the sensitivity of the system performance, that is, blocking, to changes in load, and also to changes in branch capacity, so that various network designs and routing doctrines can be compared. Specific cases can be, and have been, calculated with greater or lesser degrees of accuracy, but our objective is to be able to test out broad design philosophy particularly, and specific cases only incidentally.

CHAPTER III

OPERATION OF THE NETWORK

3.1 Matrix Representation

A convenient way of expressing the telephone network as a mathematical entity is to organize its parameters in the form of matrices.[1] In this Chapter we will establish the notations that will be used. We will thus be enabled to study the effect of various operating doctrines on the behaviour of the network.

The node and branch structure of the network is given by the topological matrix of its nodes and branches. This matrix, like those which follow, is a square matrix, n x n, in which each row corresponds to a node i, $i = 1, 2, \ldots, n$, which originates calls, and each column corresponds to a node j, $j = 1, 2, \ldots, n$, which terminates calls. The entries t_{ij} in the topological matrix $[T]$ are unity if there is a branch, (appropriately directed, as described earlier), between node i and node j, and zero otherwise. Since we are not concerned with traffic within a telephone office, entries in the main diagonal of this and the other matrices below are zero.

The branch capacities of the branches in $[T]$ are given by the entries c_{ij} in the branch capacity matrix $[C]$, where c_{ij} is the number of paths in the branch between node i and node j.

The traffic offered between nodes is given by the entries a_{ij} in the traffic matrix $[A]$. Traffic between nodes is called the "community of interest". Even where no branch exists in $[T]$, there may still be a community of interest in $[A]$; it follows from previous descriptions of the network that calls between nodes not directly connected must be relayed via tandem nodes.

Tandem nodes are specified in the routing doctrine. In dealing with general cases where there is no guiding policy or physical reason for selecting tandem nodes, the essential tandem nodes can be obtained by manipulation of the topological matrix. It is helpful to insert the letter H in the main diagonal of the topological matrix to identify each of the tandem nodes. The routing doctrine can be written in the cells of a routing matrix $\left[R\right]$; but it is often evident from the description of the network, and then $\left[R\right]$ is ommitted.

The measure of performance, or blocking, is given by the blocking matrix $\left[B\right]$, in which each entry B_{ij} is the blocking experienced by the community of interest a_{ij}.

We will be manipulating the $\left[A\right]$ and $\left[B\right]$ matrices to obtain solutions, and the significance of the entries will be made clear as we go along.

Finally, if we describe a cost matrix $\left[Y\right]$, the entries y_{ij} would represent the unit path costs between nodes i and j. We feel that the matrix $\left[T\right]$ would give us more control over the structure of the network than prohibitive or infinite costs in the $\left[Y\right]$ matrix. However, we could, if we wished, use a fully connected network and let the costs determine the structure. Since we intend to proceed with $y_{ij} = 1$, all i and all j, as stated in Section 2.1, we will not involve $\left[Y\right]$ in what follows.

previous descriptions of the network that calls between nodes not directly connected must be relayed via tandem nodes.

Tandem nodes are specified in the routing doctrine. In dealing with general cases where there is no guiding policy or physical reason for selecting tandem nodes, the essential tandem nodes can be obtained by manipulation of the topological matrix. It is helpful to insert the letter H in the main diagonal of the topological matrix to identify each of the tandem nodes. The routing doctrine can be written in the cells of a routing matrix $\left[R\right]$; but it is often evident from the description of the network, and then $\left[R\right]$ is omitted.

The measure of performance, or blocking, is given by the blocking matrix $\left[B\right]$, in which each entry B_{ij} is the blocking experienced by the community of interest a_{ij}.

We will be manipulating the $\left[A\right]$ and $\left[B\right]$ matrices to obtain solutions, and the significance of the entries will be made clear as we go along.

Finally, if we describe a cost matrix $\left[Y\right]$, the entries y_{ij} would represent the unit path costs between nodes i and j. We feel that the matrix $\left[T\right]$ would give us more control over the structure of the network than prohibitive or infinite costs in the $\left[Y\right]$ matrix. However, we could, if we wished, use a fully connected network and let the costs determine the structure. Since we intend to proceed with $y_{ij} = 1$, all i and all j, as stated in Section 2.1, we will not involve $\left[Y\right]$ in what follows.

A feasible design is always given by a consistent set of $[T]$, $[C]$, $[A]$, and $[B]$ matrices and routing doctrine. A solution to the design problem is given by a consistent set in which, (with $[T]$ and $[A]$ given, the total branch capacity C given, and the routing doctrine followed), the assignment of C into $[C]$ has been made in such a way as to minimize $\overline{B}$ as derived from $[B]$ by a particular, defined, method or doctrine.

The design problem is not really confined to a one-shot effort to find the minimum $\overline{B}$ for a given C, but in addition it involves the broad relationship between blocking and paths for a wide range of branch capacities; really, the relationship between service and cost of paths. Normally, the prerogative of putting a value on service intangibles and deciding on capital expenditures belongs to management. Placing the necessary information at the disposal of management is a responsiblity of the systems engineer. It is not enough to achieve minimum blocking if that minimum level is not satisfactory; what we really want to achieve is a high expectation of connection, with quality that the customers think is good, at the lowest possible cost. Accordingly, we will solve the problem for minimum $\overline{B}$ for each value of C in a range of network capacities representing a range of interest.

Matrix representation, as above, gives us a medium for "operating" the network with complete freedom, or for imposing constraints or patterns on certain parameters. Also, with the usual linear cost assumption, it enables us to weight the branches suitably, and obtain a solution in terms of fixed network cost in a manner analogous to the solution we will arrive at for fixed network capacity.

CHAPTER IV

SINGLE NODE DOCTRINES

4.1 Single Node Operation

Let us first consider the situation as viewed from a node which has to communicate with some surrounding nodes. We are really concerned, of course, with securing a path over which to communicate, since, once the connection between the nodes is established, the intelligence of the communication may flow in either direction. Having to communicate therefore means having to take the initiative in securing a path. Let us suppose that we have at our disposal a finite number of paths to distribute between our node and the surrounding nodes, and that these paths can only be secured from the end at our node. Let us assume that the number of paths is in excess of the number of surrounding nodes, so that we have a choice in the way we distribute the paths. However, the number is not as large as the number of subscribers at our node who can initiate calls, and hence there is a likelihood of blocking occurring.

Our problem, therefore, is to distribute the given number of paths in such a way as to minimize the blocking that our subscribers will experience.

Knowing only that for a given amount of offered traffic, blocking increases as the number of paths to which it is offered decreases, we see that there are two reasonable node doctrines for distributing the paths. We might adopt the doctrine that the blocking to each one of the surrounding nodes is to be equal, so that each subscriber in our node would experience the same blocking, irrespective of whether he directed his calls to all of the surrounding nodes, or only to one of them; indeed, irrespective of how he distributed his calls.

On the other hand, we might adopt the doctrine that we want the weighted average blocking encountered to be a minimum, even though the blocking to some of the nodes might be higher and to others lower.

We might think that the second doctrine is less palatable than the first, but actually it is not unrealistic, since it is commonly observed that the grade of service to some places is poorer than others. We might suspect that the two doctrines are equivalent under certain conditions. In any event, we will apply both doctrines, and examine the consequences.

4.2 General Representation of the Single Node

Let us draw upon the matrix description of the network given in the previous chapter, Section 3.1. A single node i is represented by the ith row in each of the matrices - topological, branch capacity, traffic, blocking. Let us stipulate that we are not concerned with intranode traffic; and that our cost function is in the form of the total branch capacity.

The row vector $\left[a_{ij}\right]$ represents the given traffic originating from our node i to each of the n nodes. For the general case we will assume that all the a_{ij}'s are non-negative. By the stipulation made above, $a_{ii} = 0$. We will define $a_i = \sum_j a_{ij}$.

For the present, we will assume that all traffic is direct;

that is, in the row vector $\left[t_{ij}\right]$, $t_{ij} = 1$ where a_{ij} is positive.

The number of paths which we can secure to each node where $t_{ij} = 1$ is represented by the row vector $\left[c_{ij}\right]$. However, the c_{ij}'s are not given; only the total number of paths $c_i = \sum_j c_{ij}$ at our disposal is given, and c_i is an interger.

The row vector $\left[B_{ij}\right]$ represents the blocking experienced by node i in securing paths to the connecting nodes.

Our problem can be expressed as that of distributing a given number of paths c_i into the row vector $\left[c_{ij}\right]$, such that $c_i = \sum_j c_{ij}$, in order to accommodate the given community of interest $\left[a_{ij}\right]$, according to the conditions of the Erlang B relationship $B_{ij}\ (c_{ij}, a_{ij})$, as previously discussed, such that the blocking $\bar{\bar{B}}$ in terms of a particular doctrine is minimized.

4.3 First Node Doctrine

The doctrine in which we aim to make the blocking to each surrounding node equal will be called the first node doctrine. The required solution is the assignement of c_i paths to (n-1) branches, such that all the B_{ij}'s are equal and as small as possible. Let us see whether both of these conditions can be satisfied simultaneously.

Consider a simple case where the traffic to each node is identical. In order to have identical blocking on every branch, it is obvious that every branch should have the same number of paths. However, unless c_i is a multiple of (n-1), the assignment of $\frac{c_i}{(n-1)}$ paths to each branch will result in fractional paths in each branch. Since fractional paths are not permissible - we can only have a path or no path, but not part of one - the first problem that arises is that of achieving integer values for the branch capacities.

If c_i is not a multiple of $(n-1)$, and we let S equal the integer part of $\frac{c_i}{(n-1)}$, then the best that we can do to achieve near-equal blocking on every path is to assign S paths to every branch, and to distribute the remainder at one to each branch, so that some of the branches will have $(S+1)$ paths. Since our problem is not in the form of a linear program, a method of obtaining integer solutions in linear programming is not applicable.[15]

The second problem is how to designate the blocking result, since we cannot now say that the blocking is equal on all branches; it is higher on the S path branches than it is on the $(S+1)$ path branches. The situation is like that of the man who could not swim, and was drowned while crossing a river whose average depth, he was told, was three feet. The important measure to him would have been the maximum depth, since there was a shipping channel in the middle of the river where the depth was over his head. So, if the telephone systems engineer is to reflect fairly the performance of his system, he must quote the maximum blocking encountered in the near-equal blocking design.

In general, when the traffic to each directly connected node is not the same and when the total branch capacity, c_i, can take on any value in the range of interest, it is not likely that every one of the branch capacities, c_{ij}, in a set will turn out to be an interger and as well produce all blockings B_{ij} equal. Evidently when we require integer solutions, the first node doctrine needs some modification.

If it were possible to permit the paths to take on non-integer values, then there would be no difficulty. We can, in fact, use a continuous form of the Erlang B relationship, instead of the discrete form, and obtain branch capacities which are not restricted to integers. By applying this to the problem of the first node doctrine we can always obtain the unique solution for the assignment of branch capacities, $\sum_j c_{ij}' = c_i$, for equal blocking on all branches, and, concomitantly, minimum blocking, by virtue of the fact that, for the continuous Erlang B function, B is a monotonically decreasing function of c, for constant a.

The path vector $\left[c_{ij}'\right]$ calculated this way is not, in general, a practical solution, because it contains non-integer values. Since we need integral branch capacities, c_{ij}'', in a practical solution, and yet we need them to be close to the exact branch capacities, c_{ij}', calculated by the continuous method, it seems reasonable to increase or decrease each c_{ij}', (assumed to be non-integral, in general), to the integer above or below it, such that $|c_{ij}'' - c_{ij}'|<1$, and such that $\sum_j c_{ij}'' = \sum_j c_{ij}' = c_i$. We will call $(c_{ij}'' - c_{ij}') = \Delta c_{ij}$, and represent the set of Δc_{ij}'s by $\left[\Delta c_{ij}\right]$. The blocking vector $\left[B_{ij}''\right]$ which corresponds to $\left[c_{ij}''\right]$ will no longer have equal blocking on all branches.

We will define a path vector $\left[c_{ij}''\right]$ as one that gives near-equal blocking if and only if there exists a $\left[\Delta c_{ij}\right]$, $|\Delta c_{ij}|<1$, over all j, such that $\left[c_{ij}''\right] + \left[\Delta c_{ij}\right]$ gives equal blocking on all branches, for a given c_i.

There will, in general, be a number of ways in which the shuffling of the paths can be accomplished, for a given c_i, and each way will produce a $\left[c_{ij}''\right]$ that gives near-equal blocking. Let us denote by S_{ij} the largest integer $\leq c_{ij}'$. Let us call $S_i = \sum_j S_{ij}$. Then the maximum number of different $\left[c_{ij}''\right]$'s giving near-equal blocking is $\binom{n-1}{c_i - S_i}$. Now we need a criterion to determine which one of of them is the best.

Let us consider two of these near-equal blocking path vectors $\left[\overset{(1)}{c_{ij}}\right]$ and $\left[\overset{(2)}{c_{ij}}\right]$, and their respective blocking vectors $\left[\overset{(1)}{B_{ij}}\right]$ and $\left[\overset{(2)}{B_{ij}}\right]$. Let us call the maximum blocking in a blocking vector B_{ijmax}. We will define that $\left[\overset{(1)}{c_{ij}}\right]$ is better than $\left[\overset{(2)}{c_{ij}}\right]$ if and only if $\overset{(1)}{B_{ijmax}} < \overset{(2)}{B_{ijmax}}$.

Hence we can say that the best of the possible $\left[c_{ij}''\right]$'s giving near-equal blocking, for a given c_i, is that one which has the least maximum branch blocking B_{ijmax}'' in its blocking vector $\left[B_{ij}''\right]$.

If we want to test an arrangement $\left[c_{ij}\right]$, which we claim has near-equal blocking on all branches, to see whether that blocking is in fact minimum, we will take a path from any branch and try to put it somewhere else so that the blocking on the branch we take it from does not rise as high as the maximum was on any branch before the attempted rearrangement. If we cannot do this, then we say that the blocking is minimum.

If we apply this same test to an arrangement $\left[c_{ij}''\right]$ giving near-equal blocking, and which we have determined is best by the definition above, because its B_{ijmax}'' is the least of all near-equal blocking arrangements, we will find that we cannot take one path from any branch which does not have the highest blocking, without the blocking on that branch rising to equal or exceed $B_{ij}''_{max}$.

If this test is applied to any of the other $\left[c_{ij}''\right]$ giving near-equal blocking, we will find that rearrangement is possible.

We can say, therefore, that our criterion for the best arrangement of $\left[c_{ij}''\right]$ enables us to meet this test for minimum blocking.

We are now in a position to modify the first node doctrine, on account of the integer constraint on the solution, as follows:

The first node doctrine is one in which we aim to distribute a given number of paths so as to achieve near-equal blocking, such that the maximum of the blockings on all branches has the least possible value; and that value of blocking will be used to designate the blocking of the path arrangement.

Although this criterion we have chosen for best path assignment, subject to integer values, is not the only one possible, it does have the virtue of giving subscribers using the branch with the worst grade of service (highest blocking), the "least worst" that it is possible to arrange.

We could have chosen, as an alternative, that $\left[c_{ij}''\right]$ which gave the smallest range between maximum and minimum blockings, so that the branches would seem to be more similar in blocking. However, given a choice, the systems engineer is not likely to assign a given number of paths in such a way as to give every branch a worse grade of service than necessary, just so that they can be more similar. Such a criterion would give no regard to the requirement that the blocking function is to be made as low as possible.

4.4 Method of Solution for the First Node Doctrine

The use of the continuous form of the Erlang B blocking function[13] may have certain advantages, for example in analytical methods of solution and in finding approximations of the blocking to be expected when c_i is given for making comparisons. However, since non-integral results have no practical meaning, if we use it, we are faced with the subsequent problem of adjusting the results in order to get whole numbers of paths.

We will therefore use a partitioning process which will give the required path assignment directly in integers.

Consider a set of monotonically decreasing functions $f_k(n)$, $k = 1, 2, \ldots, m$, defined on the finite set of integers $n \in \{0, 1, 2, \ldots, N_k\}$.

Consider a new set of functions $f_k^*(y)$, defined on the domain $f_k(N_k) \leq y \leq f_k(0)$, and defined as follows:

$$f_k^*(y) = \begin{cases} N_k & \text{if} \quad f_k(N_k) \leq y < f_k(N_k - 1) \\ N_k - 1 & \text{if} \quad f_k(N_k - 1) \leq y < f_k(N_k - 2) \\ \vdots & \\ 1 & \text{if} \quad f_k(1) \leq y < f_k(0) \\ 0 & \text{if} \quad f_k(0) = y \end{cases}$$

and where $f_k^*(y)$ corresponds to $f_k(n)$ for each value of the index k.

Let us put $F(y) = \sum_{k=1}^{m} f_k^*(y)$

where $y \in \bigcap_{k=1}^{m} \left[f_k(N_k), f_k(0)\right] = \left[u, v\right]$.

Let $F^{-1}(y)$ be the inverse mapping of $F(y)$. In the inverse mapping, to every point $n_j \varepsilon \left\{0, 1, 2, \ldots, \sum_{k=1}^{m} N_k \right\}$ there will correspond an interval of the form $\left[u_{n_j}, v_{n_j}\right) \subset \left[u, v\right]$, such that $u_{n_j} \leq y < v_{n_j}$, and where $u_{n_j} = \underset{k}{\text{Max}} \ (f_k(n))$, for $n = f_k^*(y)$, $k = 1, 2, \ldots, m$.

Let us designate $y_{n_j} = u_{n_j}$ for n_j.

Now, if we call $f_k(n)$ the blocking function B (c, a), where n corresponds to the variable c, and k is the index for the values of the parameter a, then we see that the blocking value which corresponds to $n_j = C$ is u_{n_j}, the maximum of the blocking values of the component branches for the component n = c.

The partitioning process implied by the foregoing corresponds to the choice of the best of the several possible near-equal blocking solutions, and it does, in fact, lead to the computational process for the optimal solution which is easily handled by the electronic computer. That process is the tabulation of the m sets of values of $B_k(c, a)$, $1 \leq c \leq N_k$, followed by ordering, in descending order of B, the quantities B, c, and k, with the initial value of the ordering index equal to m. If we enter the ordered table where the ordering index = C, we find the corresponding B, which represents the blocking value by which we wish to designate the near-equal blocking solution. In a more sophisticated tabulation, the tabulated values of c and k would be used to obtain, for each ordering index, the B and c for each of the k branches, and the B, c would be listed in proper tabular columns.

The partitioning process can be demonstrated graphically in the following way. Let us plot on one graph of B vs c the (n-1) relationships for $B_{ij}(c,a_{ij})$, $0 \leq c \leq c_i$, for traffic originating at node i, using the usual discrete Erlang B equation. Let us erect verticals on each point $(c, B_{ij}(c, a_{ij}))$, up to but not including the point $(c, B_{ij}(c-1, a_{ij}))$, for the same j.

Now let us sum the abscissae of the (n-1) resulting discontinuous functions in order to produce another function which will consist of a set of ordinates at integer values of c, and whose lowest point we will designate (c_k, B_k).

The graph of the points (c_k, B_k) now represents the relationship between the node blocking as defined by the first node doctrine for integer solutions, and the total number of paths available. For a given number of paths, c_i, the blocking B_k at the point $(c_k = c_i, B_k)$ is the minimized blocking for near-equal blocking to surrounding directly-connected nodes.

Let us call the minimized blocking $\overline{B}_i$. Then the corresponding path distribution can be obtained by reading off the values of c where the (n-1) discontinuous functions for the branches cut the line $B = \overline{B_i}$, and putting $c_{ij} = c$ for the corresponding j. These c_{ij}'s are the required branch capacities for the vector $\left[c_{ij}\right]$.

This method is analogous to the graphical method which would produce the required result if the continuous Erlang B relationship were used for the first node doctrine. This gives us some further justification for using the least maximum blocking criterion.

The theroretical best result, i.e. lowest blocking and equal blocking on all branches, is obtained by using the continuous

Erlang B relationship, and so we have an absolute measure of the departure from the theoretical best blocking which the branch with our least worst blocking suffers because of the integer constraint.

<u>Example</u>

Let us consider the row vector (of the representative matrices) for one node, say node 1, of a 5-node network. Thus:

$$i = 1, \qquad n = 5,$$

$$\begin{bmatrix}T\end{bmatrix}_{\text{row } 1} = \begin{bmatrix}0, 1, 1, 1, 1\end{bmatrix}$$

and let $\begin{bmatrix}A\end{bmatrix}_{\text{row } 1} = \begin{bmatrix}0, 4, 2, 3, 5\end{bmatrix}$

and $c_1 = 20$

The objective is to assign the 20 paths to the 4 branches indicated in the topological row vector, such that $\overline{B}_1$ as defined by the first node doctrine will be minimized. We will, then, produce

$$\begin{bmatrix}B\end{bmatrix}_{\text{row } 1} = \begin{bmatrix}0, B_{12}, B_{13}, B_{14}, B_{15}\end{bmatrix} \quad \text{whence } \overline{B}_1 = \underset{j}{\text{Max}}\,(B_{ij})$$

and $\begin{bmatrix}C\end{bmatrix}_{\text{row } 1} = \begin{bmatrix}0, c_{12}, c_{13}, c_{14}, c_{15}\end{bmatrix}$

First we will calculate the blocking relationships for each of the four branches, as shown in the table below, and plot them, as in Figure 2:

c_{ij}	⊙ B12 for $a_{12} = 4$	△ B13 for $a_{13} = 2$	× B14 for $a_{14} = 3$	◻ B15 for $a_{15} = 5$
1	0.800	0.667	0.750	0.833
2	0.615	0.400	0.529	0.676
3	0.451	0.211	0.346	0.530
4	0.311	0.095	0.206	0.398
5	0.199	0.037	0.110	0.285
6	0.117	0.012	0.052	0.192
7	0.063		0.022	0.121
8	0.030		0.008	0.070

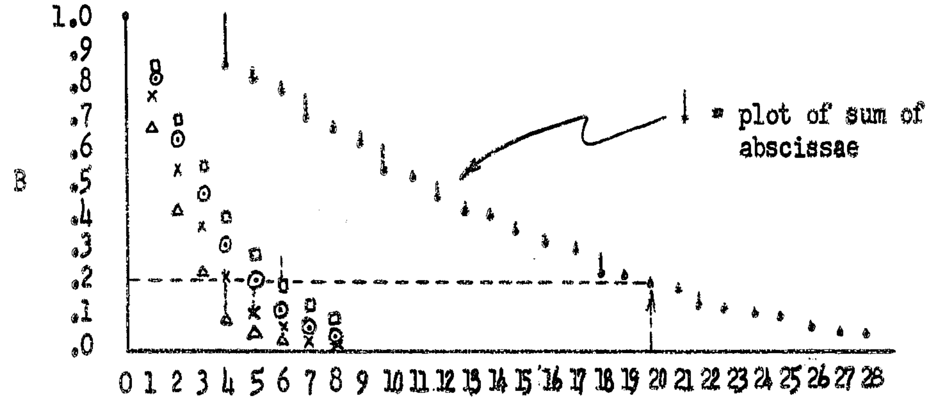


Figure 2

Of all the possible combinations of the paths which add up to 20, the one combination shown is the optimal distribution of paths. This distribution,

$$[C]_{\text{row } 1} = [0, 5, 4, 5, 6]$$

gives $[B]_{\text{row } 1} = [0, 0.199, 0.095, 0.110, 0.192]$

and $\bar{B}_1 = 0.199$

No other combination of paths totalling 20 will produce a lower maximum blocking in the set $[B]$ than 0.199.

(Incidentally, the weighted average blocking at the node 1 actually works out to be 0.163 with this path assignment.)

This example shows how easily the solution can be produced by hand or by computer. All that it is necessary to do is to produce the $(n-1) = 4$, in this case, blocking relationships shown in the table, and to merge them in descending order of blocking, starting with 1 path assigned to each branch, and adding 1 path for each successive blocking.

c_1	$\overline{B}_1$
4	0.833
5	0.800
6	0.750
7	0.676
8	0.667
9	0.615
10	0.530
11	0.529
12	0.451
13	0.400
14	0.398
15	0.346
16	0.311
17	0.285
18	0.211
19	0.206
20	0.199 - Hence $\overline{B}_1$ =
21	0.192 0.199 for
22	0.121 c_1 = 20
23	0.117
24	0.110
25	0.095
26	0.070
27	0.063
28	0.052

If, while making the new table in descending order of blocking, we tabulate also the identity of the added branch (by its termination) and the number of paths accumulated in that branch, the $[B]$ and $[C]$ can be obtained easily. For example:

c_1	$\overline{B}_1$	Terminating node	Paths in Branch
4	0.833	5	1
.	.	.	.
.	.	.	.
.	.	.	.
15	0.346	4	3
16	0.311	2	4
17	0.285	5	5
18	0.211	3	3
19	0.206	4	4
20	0.199 = $\overline{B}_1$ = B_{12}	2	5 = c_{12} for c_1 = 20
21	B_{15} = 0.192	5	6 = c_{15}

22	0.121		5	7
23	0.117		2	6
24	$B_{14} = 0.110$	$\longleftarrow$	4 $\longrightarrow$	$5 = c_{14}$
25	$B_{13} = 0.095$	$\longleftarrow$	3 $\longrightarrow$	$4 = c_{13}$
.	.		.	.
.	.		.	.
.	.		.	.

Having determined for 20 paths that $\bar{B}_1 = 0.199$, we see that it occurs on the branch to node 2, with 5 paths in that branch. Now we simply continue to search down the table for the first occurrence of each terminating node, other than this one just determined. We see that the first occurrence of the branch to node 5 has a blocking of 0.192 with 6 paths; to node 4, a blocking of 0.110 with 5 paths; and to node 3, 0.095 with 4 paths. Hence $\left[B\right]_{\text{row } 1}$ and $\left[C\right]_{\text{row } 1}$ are also obtained by ordering.

Before leaving this example, let us examine the result of taking one path from any one of the other three branches in order to give it to the branch to node 2 and reduce its blocking to 0.117:

Thus, we want $\left[B\right] = \left[0,\ 0.117,\ \ldots\ldots\right]$

so if we take it

from the branch to

node 3, we get: $\left[B\right] = \left[0,\ 0.117,\ 0.211,\ \ldots\ldots\right]$

to node 4: $\left[B\right] = \left[0,\ 0.117,\ \ldots,\ 0.206,\ \ldots\ldots\right]$

and to node 5: $\left[B\right] = \left[0,\ 0.117,\ \ldots\ldots,\ 0.285\right]$

Hence, we find that the alternative assignments all have a branch with blocking higher than 0.199, so our previously determined assignment meets the objective.

4.5 Second Node Doctrine

If the designer wants a feeling for the effect of system performance on the majority of subscribers, it seems reasonable that he would weight the blocking on each branch in proportion to the fraction of the total node traffic offered on each branch, in order to arrive at an average. The weighted average blocking defined this way is, of course, the lost fraction of the total traffic offered at the node.

The doctrine in which we seek to minimize the weighted average blocking will be called the second node doctrine. The required solution is the assignment of c_i paths to (n-1) branches such that $\frac{\sum_j a_{ij} B_{ij}}{\sum_j a_{ij}}$ is minimum. We shall call $\bar{B}_i$ the minimum weighted average blocking which corresponds to the solution.

In effect, we are seeking a solution such that, having distributed the given number of paths, c_i, there is no change which we can make in any of the c_{ij}'s which will reduce the weighted average blocking at the node.

Again we have the obvious problem of obtaining an integer solution, as in the first node doctrine. However, there is no problem in designating the minimum weighted average blocking, because weighted average blocking is clearly defined above.

Let us consider a single node, say node i, at which we let $B_i = \frac{\sum_j a_{ij} B_{ij}}{\sum_j a_{ij}}$, for $j = 1, 2, \ldots, n$ but $j \neq i$. We seek the solution $\bar{B}_i$ by minimizing B_i. Let us put $\sum_j a_{ij} = a_i$. Let us re-index j over $m = (n-1)$ branches, so we will say $j = 1, 2, \ldots m$. We shall take partial derivatives of B_i with respect to $c_{i1}, c_{i2}, \ldots, c_{i(m-1)}$, remembering that $c_m = c_i - c_{i1} - c_{i2} - \ldots - c_{i(m-1)}$. Applying these operations, we obtain

$$\frac{a_{ik}}{a_i} \cdot \frac{d\,B_{ik}}{dc_{ik}} + \frac{a_{im}}{a_i} \cdot \frac{d\,B_{im}}{dc_{im}} (-1) = 0$$

for all k, $k = 1, 2, \ldots, m$. Since this is true for any k, we have

$$\begin{aligned} \frac{a_{i1}}{a_i} \cdot \frac{dB_{i1}}{dc_{i1}} &= \frac{a_{i2}}{a_i} \cdot \frac{d\,B_{i2}}{dc_{i2}} \\ &= \frac{a_{i3}}{a_i} \cdot \frac{dB_{i3}}{dc_{i3}} \\ &\;\;\vdots \\ &= \frac{a_{im}}{a_i} \cdot \frac{dB_{im}}{dc_{im}} \end{aligned}$$

Now we have the second node doctrine problem in a form which is similar to that of the first node doctrine, except that we want the (n-1) weighted derivatives of blocking, $\frac{a_{ij}}{a_{ij}} \cdot \frac{dB_{ij}}{dc_{ij}}$, to be equal, instead of the actual blocking B_{ij}. We will call $D_j =$ $\frac{a_{ij}}{a_i} \cdot \frac{dB_{ij}}{dc_{ij}}$.

We understand in the above that, while $B_{ij} = f(c_{ij}, a_{ij})$, we have arbitrarily fixed the value of a_{ij}, so that $B_{ij} = f(c_{ij})$ for

a_{ij} fixed. Hence we have expressed the derivative of blocking with respect to paths as $\frac{dB_{ij}}{dc_{ij}}$ instead of $\frac{\partial B_{ij}}{\partial c_{ij}}$.

4.6 Method of Solution for the Second Node Doctrine

If we want the results in whole numbers of paths, the partitioning process used is the same as described in Section 4.4. If we call $f_k(n)$ the weighted derivative of the blocking function with respect to paths, then we can use the same method to identify the set of n's, $\sum_k n = n_j$, which produce near-equal weighted derivatives. Since n corresponds to c, $n_j = c_i$, and a's are given, it is simple matter to compute B (c,a), and $\bar{B}_i = \frac{\sum a_{ij} B_{ij}}{\sum a_{ij}}$. Appendix II gives examples of the first derivatives of the Erlang B function;[16] this is taken from a much more extensive table.[17] In lieu of exact values of $\frac{\partial B}{\partial c}$, Appendix III gives approximate $\frac{\partial B}{\partial c}$ values over a small range of interest, calculated from Appendix I, using tangents to a fitted curve.

To apply the graphical counterpart of this method, we will have to evaluate the partial derivatives at integral values of c, and then proceed to the solution by the same method as the first node doctrine, plotting instead $\frac{a_{ij}}{a_i} \cdot \frac{dB_{ij}}{dc_{ij}}$, or D_j, vs c for $0 \leq c \leq c_i$, for each of the (n-1) branches, on a D vs c graph. In erecting the verticals, we will direct them away from the c-axis, as before, and after summing the ordinates to produce the total branch capacity function, we will designate the points on the vertical segments of it nearest to the c-axis as (c_k, D_k).

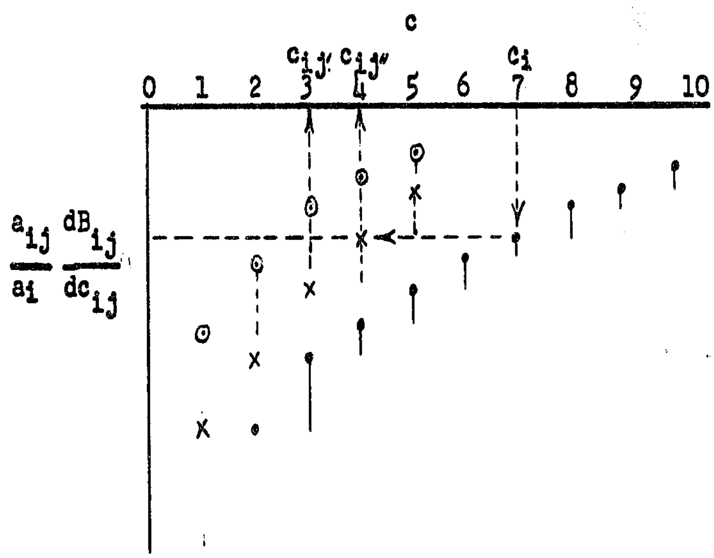
By running a horizontal line $D = D_k$ at the point $(c_k = c_i, D_k)$ across the ordinates for the (n-1) branches, we can read off the various c_{ij} at the intersections. These c_{ij}'s are the required branch capacities for the vector $\left[c_{ij}\right]$.

To complete the solution, the several values of B_{ij} to each of the surrounding nodes are found for corresponding pairs of c_{ij} and a_{ij}, and are used in calculating the weighted average blocking as defined. This average is the minimized blocking function required and is designated $\bar{B}_i$.

While solutions can be obtained using continuous functions here too, the remarks made earlier about non-integer results also apply. The immediate purpose of this paper is to explore behaviour and indicate feasible solutions, rather than to elaborate on methods of solution. Undoubtedly there is room in this field for seeking and applying techniques for determining the minimum value of the maximum of sets of minima of a set of functions subject to mutual constraints.

Example

For the second node doctrine, we obtain the value of $\frac{dB_{ij}}{dc_{ij}}$ for integer values of c, $0 \leq c \leq c_i$, for each a_{ij}. Each set of $\frac{dB_{ij}}{dc_{ij}}$ values is weighted by the fraction of its traffic to the total traffic. The weighted values are plotted and verticals erected, as in Figure 3. The sum of the (j-1) functions is plotted, with respect to the domain f(B). For a sum equal to c_i given, the values of c_{ij} for near-equal values of $\frac{a_{ij}}{a_i} \frac{dB_{ij}}{dc_{ij}}$ can be found, and these values of c_{ij}, along with their corresponding a_{ij}, give us B_{ij} directly.



Functions shown by ⊙ and by × are for two different levels of traffic from node i to 2 other nodes; functions shown by ⊙ and by × are constructed as per method; function shown by | is sum of the two functions. When we enter at integer value of c_i we find corresponding integer values of c_{ij} for the two component functions as shown. Example $c_i = 7$, values of c_{ij}'s are 3 and 4.

Figure 3

Since the calculation of B from the Erlang B formula is cumbersome; of the traffic or the paths when B is an argument, even worse; and of the partial derivative by differentiating a continuous form of the Erlang B function, very formidable, it is customary to use tables with appropriate arguments. For that very reason, network calculations using traffic and blocking relationships tend to be approximations at best; a lot depends on the fineness of the tables.

4.7 Comments on the Node Doctrines

It is evident from the procedure used in the second node doctrine that the distribution of paths has been made in such a way as to achieve the minimum possible weighted average blocking. Assignment according to the first node doctrine will therefore result in an equal or larger value of weighted average blocking, although it will be minimum subject ot the constraint that blocking to all nodes will be equal (continuous) or near-equal (integer), or that no set of blockings can be found that has a lower value of the maximum in the set. (The first node doctrine will usually result in a larger value of weighted average blocking than the second node; but, on account of the integer constraint, it sometimes happens that both doctrines will give the same assignment). It can be seen, when the blocking is near-equal, and when the derivatives of blocking with respect to paths, to each of the other nodes, are not equal, and when the assignment for near-equal blocking is not identical to the assignment for minimum weighted average blocking, that it is possible to move a path from one branch to another in such a way as reduce the blocked traffic more to one node more than it is increased to the other node, and consequently, the weighted average blocking can be reduced. Because the derivative of blocking with respect to traffic is not independent of traffic, apparently the only condition under which minimum blocking by both doctrines will be equal will be when the traffic to all nodes is equal, i.e. all a_{ij}'s are equal.

It is also evident that both of these doctrines have distributed the paths as an increasing function of the amount of traffic offered to them. This is certainly a reasonable sort of expectation. Let us ask: What is the effect on the traffic offered to the various nodes, assuming traffic not all equal, when we go from the first node doctrine to the second in order to get smaller average blocking with the same given total number of paths? We tend to make the number of paths to each node become less nearly alike, because, starting from the condition when we have $c_{ij} > a_{ij}$ and near-equal blocking on the B vs c functions for all branches, we have to decrease the number of paths to nodes to which the traffic is small and increase the number of paths to nodes to which the traffic is large in order to move along the several respective B vs c curves until we are at points on them where the slopes are all inversely proportional to the fraction of the traffic offered. The effect on the traffic is, therefore, to increase the blocking encountered to nodes where the traffic offered is small, and to reduce blocking where traffic is heavy.

Conversely, it stands to reason that, if we fashion a third doctrine in which we distribute the paths uniformly to all nodes, independently of the traffic to any node, then we produce relatively high blocking where the traffic is heavy and relatively little blocking where the traffic is light. It can be shown that the average blocking for the uniform distribution is higher than it is for the first node doctrine, because when we arrange for near-equal blocking to all nodes, it is also possible to move a path from one branch to another in such a way as to increase the blocked traffic more to one node than it is decreased to the other. This occurs when a path is taken from a larger group than the group it is moved to; thus we tend to increase the weighted average blocking when we tend to make the distribution of paths more equal.

It should be pointed out here that an advantage of using matrices and graphical methods for obtaining solutions is that constraints can be applied quite easily to elements of the blocking or path vectors. Such methods are also valuable in obtaining results in cases which are too complicated for analytical solution. Analytical methods, in the regions for which they exist, are most useful for avoiding the more tedious graphical operations for large problems, and for testing various configurations and comparing various conditions. However, suitable analytical methods are difficult to find. The method we have described can be programmed for the electronic computer.

4.8 Electronic Computer

The electronic computer can be used to make optimal assignments, comparisons, etc., by this method. It may not be the best way; but if it is, it still may not be easy if calculations for Erlang B and its derivatives are required. We will simply indicate, in a block diagram, Figure 4, the major steps which a computer program might follow. The time spent in writing a program would depend on how sophisticated one chose to make it, but it may be worthwhile for a large volume of design work.

COMPUTER FLOW DIAGRAM FOR NETWORK DESIGN FOR STOCHASTIC TRAFFIC

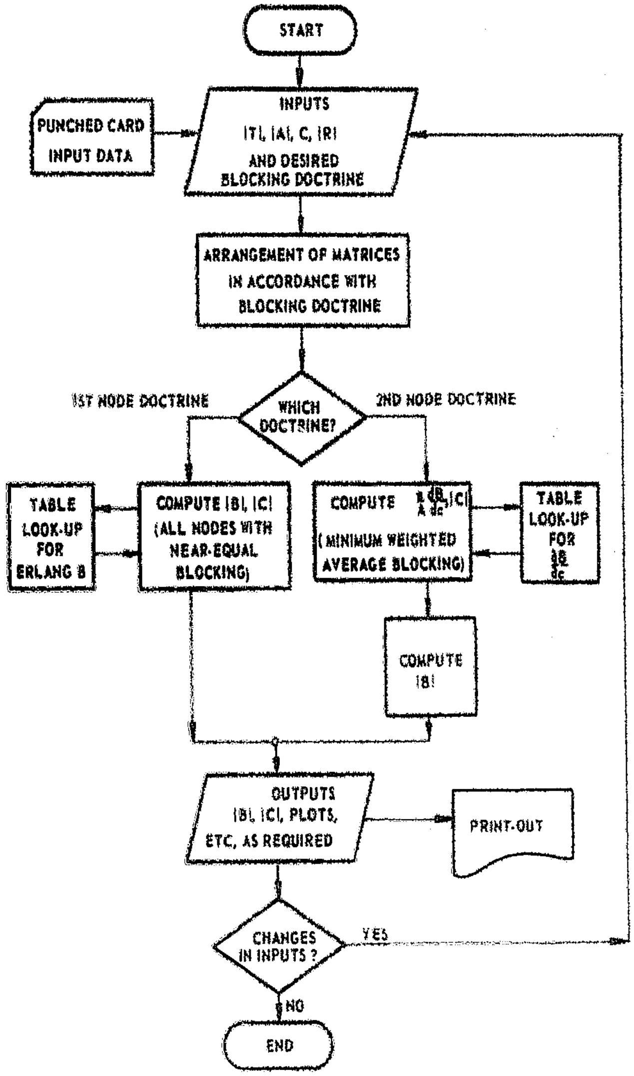


FIGURE 4

CHAPTER V

NETWORK DOCTRINES

5.1 Network Composed of Several Nodes

When we come to consider a network composed of several interconnected nodes, each like the single node just discussed, we have the same problem of distributing a given number of paths so as to minimize the blocking which our subscribers will encounter. The directed branches of our model, and their relationship to the nodes, defined earlier, enable us to treat the network as though each node were independent.

5.2 Path Assignment by Nodes

If we assume that each node has been assigned a specific number of paths on which it may initiate calls, then no further problem exists. The solution is found by application of one of the previously described node doctrines to each node in turn.

This case should arise if the nodes belonged to autonomous operating companies, each providing service according to its means and its particular policy. Although we cannot discount this possibility, it is not likely to arise because of the degree of association among telephone companies, essential to their well-being and to the satisfactory performance of the communication network. Nevertheless, our model permits us to accommodate this case, if we are dealing with a network in which we find regional differences.

5.3 Path Assignment to the Whole Network

We shall consider in detail applying two reasonable network doctrines. In one, we can take the view that an average subscriber should experience the same blocking in any node in which he might be, and we can distribute the available paths accordingly. No node, then, is given preferential treatment.

We might, on the other hand, wish to distribute the paths so that the whole network suffers as small a weighted average blocking as possible, without any concern for whether an average subscriber at one node gets more blocking than another at another node.

The second network doctrine seems unfair, compared with the first; and in fact it would give rise to unfavourable reaction in in practice if the blocking at some nodes were noticeably worse than at others. Again we will solve the problem first, then compare the doctrines.

5.4 General Representation of the Network

The matrix representation of the network, as described in the chapter on network operation, will be used in all that follows. The routing doctrine, and suitable routing matrices, will be developed as we proceed.

Our problem is now the originally stated one of distributing C paths into the matrix $\left[C\right]$, such that $c = \sum_i \sum_j c_{ij}$, given the community of interest $\left[A\right]$, and the topological structure $\left[T\right]$, conditional upon the Erlang B relationship between paths and traffic, in such a way as to minimize a function of blocking which we will carefully define.

5.5 First Network Doctrine

The doctrine with the objective of making the blocking at all nodes in the network near-equal will be called the first network doctrine.

The first network doctrine can be met in two ways, depending upon whether we apply the first or second node doctrine. When we stipulate that the blocking from one node to all other nodes must be near-equal and that all nodes themselves must have near-equal blocking, it follows logically that we are confined to making the blocking on all groups of paths in the network near-equal.

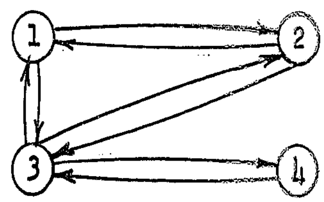
5.6 Method of Solution for First Network Doctrine with First Node Doctrine

The solution for the whole network can be arrived at graphically in the same way as for the first node doctrine, Section 4.4, by plotting the n(n-1) curves of B_{ij} (c, a_{ij}), for $0 \leq c < C$, on one graph of B vs c. By striking a position with a horizontal where the abscissae sum to C, we can read off the required values of the c_{ij}'s. We can designate the value of the blocking which applies to the network as the blocking on that branch which has the highest of the near-equal blockings. We can then determine the blocking on each branch in the network. (If we wish, we can designate the blocking at each node by the maximum value of blocking on the branches originating at each node.)

This will give the best near-equal blocking possible, on account of the integer constraint on the paths.

<u>Example</u>

Consider a simple example of a directly connected network which is not necessarily fully connected. Suppose we are given:



that is, $\begin{bmatrix} T \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

and $\begin{bmatrix} A \end{bmatrix} = \begin{bmatrix} 0 & a_{12} & a_{13} & 0 \\ a_{21} & 0 & a_{23} & 0 \\ a_{31} & a_{32} & 0 & a_{34} \\ 0 & 0 & a_{43} & 0 \end{bmatrix}$

We are required to determine $\begin{bmatrix} B \end{bmatrix}$ and $\begin{bmatrix} C \end{bmatrix}$ such that:

$$\begin{bmatrix} B \end{bmatrix} = \begin{bmatrix} 0 & B & B & B \\ B & 0 & B & 0 \\ B & B & 0 & B \\ 0 & 0 & B & 0 \end{bmatrix}$$

where B represents the

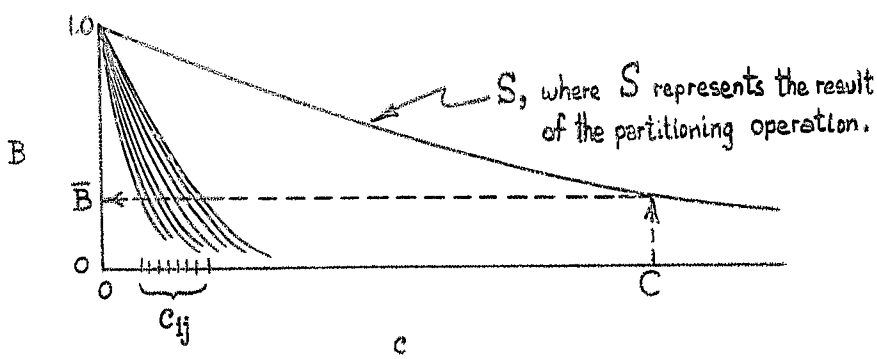
near-equal values of B, and where the B_{max} of the set is the least B_{max} in all possible sets of near-equal blocking; and such that the sum of the elements of:

$$[c] = \begin{bmatrix} 0 & c_{12} & c_{13} & 0 \\ c_{21} & 0 & c_{23} & 0 \\ c_{31} & c_{32} & 0 & c_{34} \\ 0 & 0 & c_{43} & 0 \end{bmatrix}$$

will equal a given value which we shall denote by C, i.e.,

$$\sum_{i=1}^{4} \sum_{j=1}^{4} c_{ij} = C$$

In every case, of course, $B = B\ (c_{ij}, a_{ij})$ over all i, j. Also, $\bar{B} = B_{max}$. We will take the liberty of representing the relationships in continuous form in the graph below, because it is a satisfactory visual analogy, although we will actually proceed with the partitioning form of solution described earlier, and used in Section 4.4



Obviously the blocking relationships for the zero traffic elements can be ignored, since no paths are required on their account. Proceeding as in Section 4.4, we can obtain the $\bar{B}$ corresponding to C; and $\begin{bmatrix} B \end{bmatrix}$ and $\begin{bmatrix} C \end{bmatrix}$ as well.

5.7 Method of Solution for First Network Doctrine with Second Node Doctrine

However, if we simply require that the nodes have the same weighted average blocking, irrespective of the magnitude of the blocking on any individual group of paths, then the problem is more difficult to solve. We proceed in two steps. First we set up the relationships at one node according to the second node doctrine, Section 4.6, and obtain values of $\bar{B}_i$ at every integral value of c_i, $0 \le c_i \le C$; and plot $\bar{B}_i$ vs c_i on a new B vs c graph for the purpose. Then we do likewise for every node in the network. This gives us on one graph the whole picture of blocking as far as the nodes are concerned, on the basis of minimum weighted average blocking.

For the second step we strike a horizontal on this node blocking graph, as though we were applying the first node doctrine, in order to find the level at which the several c_i add up to C. Thus we obtain the value of the blocking for the network, and also the number of paths allocated to each node. The assignment of paths at each node is read from the corresponding node graph at the level which agrees with c_i. The minimum weighted average blocking at each node will be near-equal, rather than equal, of course.

Example

Using the same example as in Section 5.6, this time we are required to determine $\left[B\right]$ and $\left[C\right]$ such that:

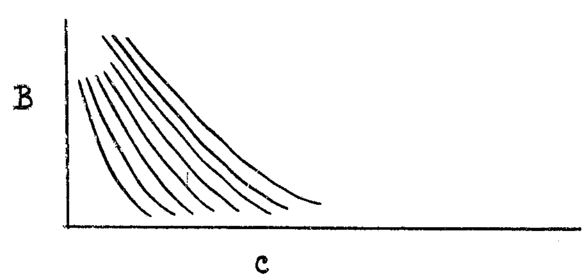
$$\left[B\right] = \begin{bmatrix} 0 & B_{12} & B_{13} & 0 \\ B_{21} & 0 & B_{23} & 0 \\ B_{31} & B_{32} & 0 & B_{34} \\ 0 & 0 & B_{43} & 0 \end{bmatrix} \quad \text{where} \left\{ \begin{aligned} &\frac{a_{12} B_{12} + a_{13} B_{13}}{a_{12} + a_{13}} = \bar{\bar{B}}_1 \\ &\frac{a_{21} B_{21} + a_{23} B_{23}}{a_{21} + a_{23}} = \bar{\bar{B}}_2 \\ &\frac{a_{31} B_{31} + a_{32} B_{32} + a_{34} B_{34}}{a_{31} + a_{32} + a_{34}} = \bar{\bar{B}}_3 \\ &\frac{a_{43} B_{43}}{a_{43}} = \bar{\bar{B}}_4 \end{aligned} \right.$$

and where the minimum weighted average blockings $\bar{\bar{B}}_i$ represent a set of near-equal values of $\bar{\bar{B}}_i$; and where the largest $\bar{\bar{B}}_i$ is the least value of $\bar{\bar{B}}_i$ in all possible near-equal sets of $\bar{\bar{B}}_i$; and such that

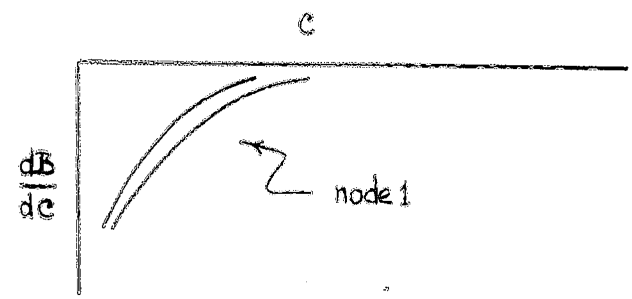
$\sum_{i=1}^{4} \sum_{j=1}^{4} c_{ij} = C$, as before.

Again, in every case $B_{ij} = B(c_{ij}, a_{ij})$ for all i, j. (The method will be demonstrated with sketches of continuous functions.)

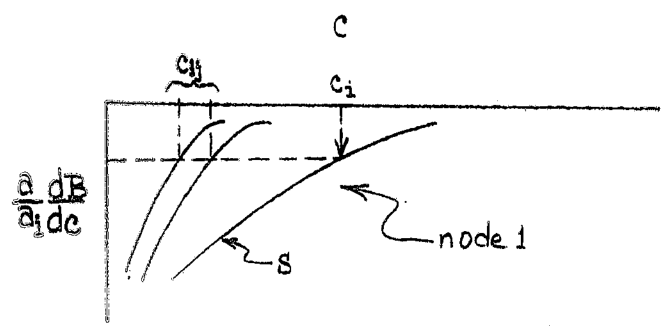
From the same blocking relationships:



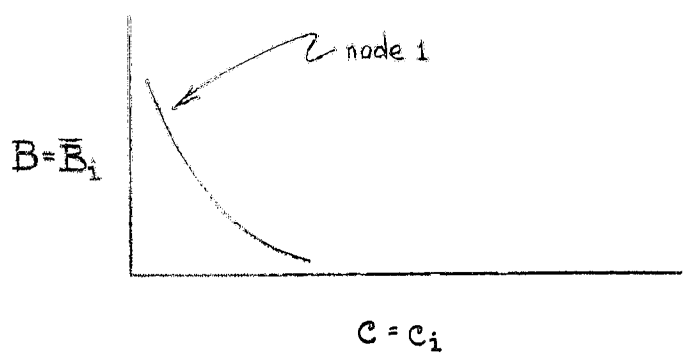
we derive the first derivative relationships, for each node. Starting with node 1, for example, we get these relationships:



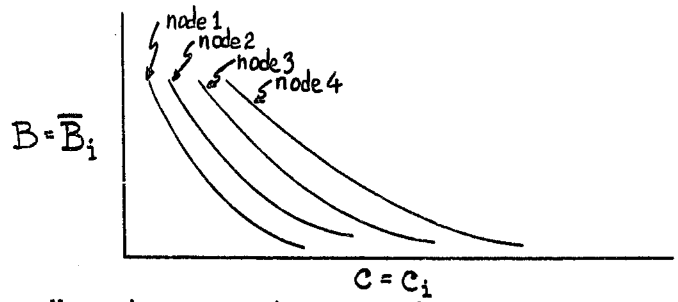
and weight them according to the traffic:



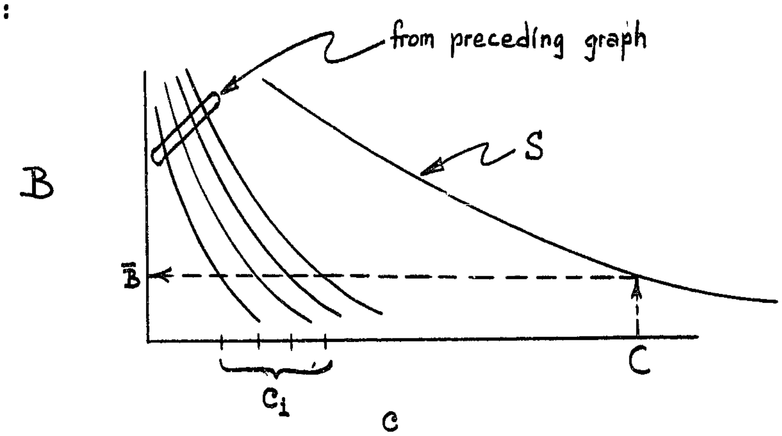
We allow c_i to cover a suitable range of values, and read off the c_{ij} for each c_i, and calculate the weighted average blocking $\overline{B}_i$ for each set of c_{ij}. We get the following result:



This is repeated for each node:



Now, since we want near-equal blocking for these four nodes with respect to each other, we apply the method of Section 4.4 for near-equal blocking of the node traffic as a whole:



This gives us the near-equal blockings for each node, and the total path assignment to each node, c_i. Going back to the weighted derivative graphs for each node, and reading into them at c_i, we find the c_{ij} assigned paths on each branch from the node. Then, the blocking on each branch, B_{ij}'s is found by reading into the original blocking relationships at each c_{ij}.

5.8 Second Network Doctrine

The second network doctrine will refer to the one that requires that the weighted average blocking in the whole network be as small as possible, with no requirement that the blocking at the nodes be equal, nor that the blocking on groups of paths be equal.

5.9 Method of Solution for the Second Network Doctrine

The solution is found by extending the method of the second node doctrine, Section 4.6, to the entire network, because here, too, we want to allocate the paths in such a way that there is no change which can be made in any of the c_{ij}'s which will produce a lower weighted average blocking. $\overline{B} = \dfrac{\sum\sum_{ij} a_{ij} B_{ij}}{\sum\sum a_{ij}}$, and we will take partial derivatives of this with respect to each of the c_{ij}'s and equate to zero.

Proceeding as we did for the second node doctrine, we find for our solution the required minimum value of $\overline{B}$, and the B_{ij} and c_{ij} values for the blocking and path matrices.

Example

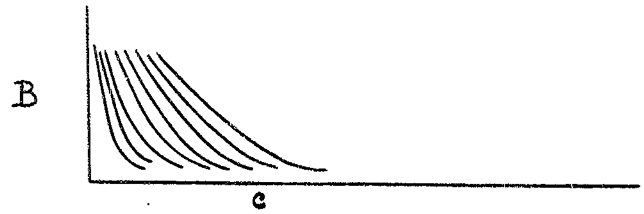
Now, for the same example, the objective is to determine $[B]$ and $[C]$ such that:

$$[B] = \begin{bmatrix} 0 & B_{12} & B_{13} & 0 \\ B_{21} & 0 & B_{23} & 0 \\ B_{31} & B_{32} & 0 & B_{34} \\ 0 & 0 & B_{43} & 0 \end{bmatrix}$$

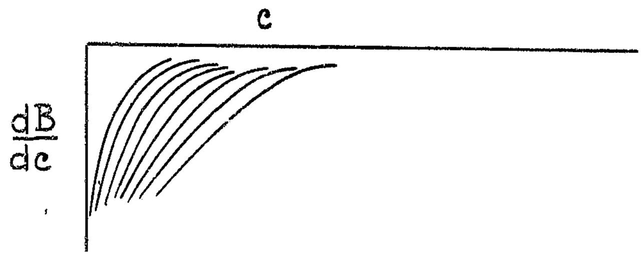
where $\dfrac{\sum\sum_{ij} a_{ij} B_{ij}}{\sum\sum_{ij} a_{ij}} = \overline{B}$, and where $\overline{B}$ is to be minimum; and such that

$\sum_{i=1}^{4} \sum_{j=1}^{4} c_{ij} = C$, as before. Let $\sum\sum_{ij} a_{ij} = \sum_i a_i = A$. (The method will again be shown by sketches of continuous functions.)

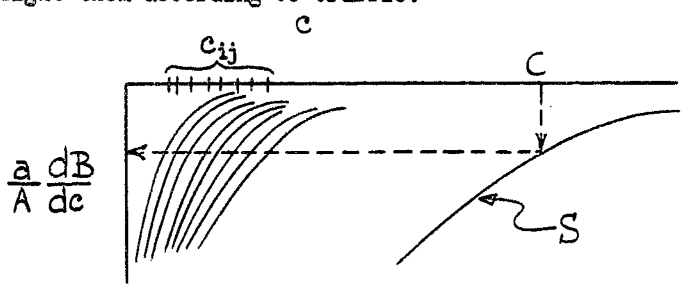
From the same blocking relationships:



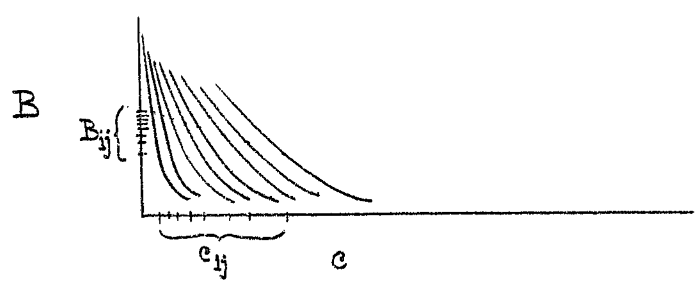
we derive the first derivative relationships for all nodes, all branches:



and weight them according to traffic:



For the given C, we can read off the c_{ij} for each branch. The blocking on each branch is obtained from the original blocking relationships:



and the weighted average blocking which actually applies is now easily calculated from:

$$\bar{B} = \frac{\sum\sum a_{ij} B_{ij}}{\sum_i\sum_j a_{ij}}$$

5.10 Comments on the Network Doctrines

By employing the above methods, we have been able to achieve the objective of minimizing the blocking of networks without any condition on the amount of interconnection, (that is, independently of the topological structure), for direct routes.

When the a_{ij}'s are not all equal, the second network doctrine, like the second node doctrine, produces higher blocking on the branches carrying light traffic than on branches carrying heavy traffic in the network. This is because the paths are less uniformly distributed than they are by the first network doctrine which produces equal blocking at all nodes, (when $c_{ij} > a_{ij}$).

A third network doctrine suggests itself, to round out the picture: uniform distribution of the given number of paths among all the groups. Obviously this will result in higher blocking on the branches carrying high traffic than on those with low traffic; and, by reasoning analogous to that for the single node, the weighted average blocking will be higher than that for either of the other two doctrines.

The three doctrines will, of course, result in equal minimum values of blocking when the offered traffic is uniformly distributed, i.e. when the traffic on every branch is equal.

5.11 Choice of Node and Network Doctrines

The node and network doctrine offer a wide choice of approaches to the problem. The first type doctrines have the advantage of giving approximately the same blocking (near-equal) to all persons offering traffic at any and all nodes, regardless of the distribution of his own community of interest. It is patently realistic to assume near-equal blocking in the absence of any rationale for unequal blocking. We have arbitrarily chosen to specify the resultant situation by the worst, or highest, of the near-equal blockings. This might be regarded as a disadvantage, and in order to get around this we could calculate and quote the corresponding weighted average; but then we would have subordinated the objective of achieving equality of blocking.

The second type doctrines, on the other hand, yield the lowest weighted average blocking right away. (The weighted average blocking of the near-equal solution is not necessarily the lowest.) They are useful when we wish to favour the grade of service on branches which are offered the largest amounts of traffic. They do, however, present the possible disadvantage that individuals offering traffic will, in general, experience widely differing levels of blocking.

The first network second node doctrine is an example of combining doctrines to suit a particular purpose.

The third type doctrines may also have their place, but are not of immediate interest. In general, we do not want unequal blocking at the nodes, so we propose the use of the first type doctrines preferentially, and the second type only where advantageous, in telephone network applications.

5.12 Structure of the Network

At this point we should discuss fully the relationship between the topological structure of the network and the community of interest. When we are dealing with direct traffic only, as we have been until now, it generally makes no difference whether the topology is defined or not. The Erlang B formula shows that when $a = 0$, then for $c = 0$, we have $0 < B \leq 1$. Thus when the topology is not defined and when an element in the community of interest is zero, the number of paths which will be allocated by the solution to that element is also zero, for any positive value of blocking which gives us a solution. This result would be in agreement with, and indeed, would be identical with, the fact that, if the topology were defined, the corresponding element in the topological matrix would be zero, because no traffic implies no paths.

The only cases so far where it would be useful to define the topology would be for the uniform, (third type), node and network doctrines. The blocking solution would be affected by whether the network were fully or only partially connected if there were some zero elements in the community of interest. Although we would probably mean that the given paths are to be distributed uniformly only where there is traffic to be carried, nevertheless some ambiguity could exist unless the structure were defined.

The practice of specifying the topological structure always makes it clear just where paths are to be provided, and is consistent with the methods for distributing paths under the various doctrines. The topological matrix, is, in fact, the conventional means of representing the network for mathematical operations, and from it conventional flow graphs can be constructed. It is important for these, if for no other, reasons.

The topological matrix is essential, however, when it comes to dealing with traffic which is not direct, and with various possible routing doctrines, as well as with the problem of merging traffic. In these applications, the topological structure dictates the assignment of paths and the calculation of blocking, and there may be positive traffic elements for which the corresponding topological elements are zero.

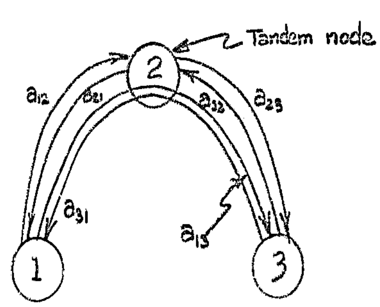
CHAPTER VI

ROUTING DOCTRINES

6.1 Fixed Routing

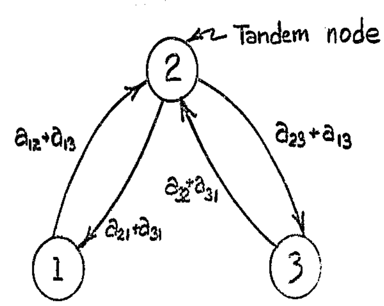
The next step in developing the general communication network for telephone traffic is to consider fixed routing. In many networks we find that there is a community of interest between nodes which are not connected with each other. A network need not be fully connected, but it must, in general, consist of nodes which are connected directly with at least one other node for originating traffic and likewise for terminating traffic; isolated nodes are not part of the communication network.

There are two ways of giving paths to tandem traffic. One way is to assign paths along each indirect route, dedicated to the exclusive use of the community of interest involved, Figure 5. The other way is for it to share the paths between nodes along the route with any other traffic that happens also to need paths between the same nodes, Figure 6. Although both of the modes could be applied at once, not only within the network, but also within a single route, we will assume for clarity that the modes are not so mixed.



Dedicated Paths

Figure 5



Shared Paths

Figure 6

6.2 First Tandem Doctrine

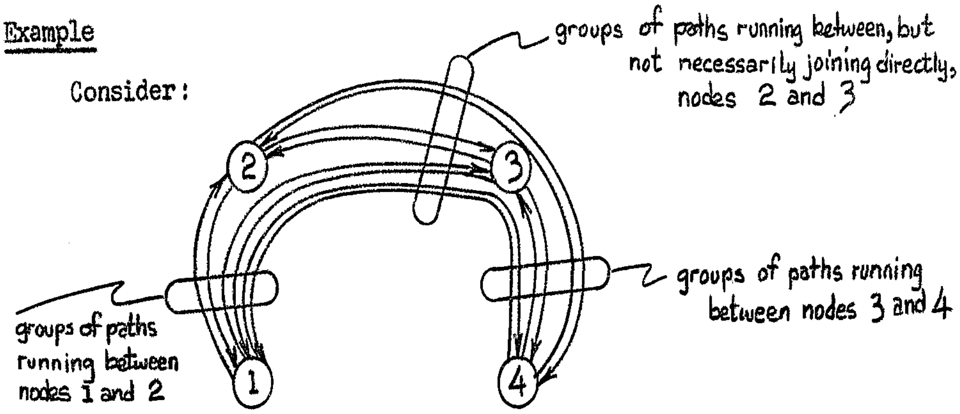
If we dedicate paths for indirect traffic between two given nodes, it is obvious that every group of paths between nodes on the route should have the same number of paths, since, if the numbers of paths varied, the traffic would be limited by the group having the smallest number of paths. It is also clear that there is no blocking problem at the intervening nodes, because the traffic does not have to compete with other traffic at the intervening nodes to secure the dedicated paths. In actual practice this does not mean that switching is obviated, but the effect is the same as if it were.

Hence the dedicated paths can be derived by taking the group of direct paths which, if allowed, would be required between the given pair of nodes, and stringing it via the specified intervening node or nodes.

The distribution of paths for this case and the calculation of minimum blocking is therefore carried out as already described in Section 5.6, with the exception that for each indirect route the curve for its community of interest is repeated once for every intervening node, in order to account properly for the total number of paths as defined. This is equivalent to weighting the paths corresponding to the direct route by a factor of $(m_{ij} + 1)$ where m_{ij} is the number of intervening nodes from node i to node j. (Note: this does not mean that the blocking B_{ij} is based on $(m_{ij} + 1) \cdot c_{ij}$ paths.) Let us define a matrix $[M]$ as consisting of the elements m_{ij} for all i, j, (for i = 1, 2, ..., n and j = 1, 2, ..., n); and also a path weighting factor matrix $[F]$, whose elements $f_{ij} = m_{ij} + 1$, (for all i, j, for i = 1, 2, ..., n and j = 1, 2, ..., n).

Example

Consider:



Given $[T] = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ and $[A] = \begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} \\ a_{21} & 0 & a_{23} & a_{24} \\ a_{31} & a_{32} & 0 & a_{34} \\ a_{41} & a_{42} & a_{43} & 0 \end{bmatrix}$

and C, where C is the total number of paths to be assigned;

we derive $[M] = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix}$ and $[F] = \begin{bmatrix} 1 & 1 & 2 & 3 \\ 1 & 1 & 1 & 2 \\ 2 & 1 & 1 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}$

We want $[B] = \begin{bmatrix} 0 & B_{12} & B_{13} & B_{14} \\ B_{21} & 0 & B_{23} & B_{24} \\ B_{31} & B_{32} & 0 & B_{34} \\ B_{41} & B_{42} & B_{43} & 0 \end{bmatrix}$ such that $B_{ij} \leq \overline{B}$ over all i,j and $\overline{B}$ is near-equal blocking, subject to C given.

Now, in the partitioning or plotting process, we consider the relationship $B_{ij} = B(c_{ij}, a_{ij})$ vs $f_{ij}\, c_{ij}$, over all i, j which appear in $\left[A\right]$, and then treat the problem as in Section 5.6 for near-equal blocking, $\bar{\bar{B}}$. Let us call the path axis c'. Then the partitioned values of c'_{ij} corresponding to B_{ij} (at the $\bar{\bar{B}}$ solution) are weighted according to their length. Let us therefore produce a matrix $\left[U\right]$ whose elements $u_{ij} = \frac{c'_{ij}}{f_{ij}}$, over all i,j in the example.

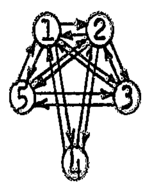
Finally, we sum the appropriate u_{ij}'s to form the elements in $\left[C\right]$ which correspond to the given branches in $\left[T\right]$, and hence we obtain optimal assignment of branch capacity.

6.3 Determination of Tandem Nodes

At this point it is convenient to introduce a method of determining the essential intervening nodes, or tandem nodes, if they have not been specified. In working with hypothetical network examples, the tandem nodes will not usually be specified. The method is to raise the topological matrix to sucessive powers until all the off-main diagonal elements are non-zero. Nodes corresponding to the element or elements having the maximum value along the main diagonal are the essential tandem nodes; although essential, some may be alternative, and can be eliminated by trial. This solution is not necessarily the only one, but it is a definite one in the absence of any other specified route; and it has the virtue of giving the minimum number of nodes to do the job, since indirect traffic uses paths which increase in quantity as a function of the number of intervening nodes.

This method also tends to concentrate the indirect traffic; the effect of this on the individual and overall community of interest will be discussed later.

Let us consider, for example, the following network:



$$[T] = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \text{ and } [A] = \begin{bmatrix} 0 & a & a & a & a \\ a & 0 & a & a & a \\ a & a & 0 & a & a \\ a & a & a & 0 & a \\ a & a & a & a & 0 \end{bmatrix}$$

Since it is clear that certain branches to carry direct traffic are missing, we fill first of all square the topological matrix:

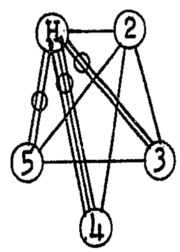
$$[T]^2 = \begin{bmatrix} 4 & 3 & 2 & 1 & 2 \\ 3 & 4 & 2 & 1 & 2 \\ 2 & 2 & 3 & 2 & 2 \\ 1 & 1 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 3 \end{bmatrix}$$

Since t_{11} and t_{22} are maximum in the diagonal, and no elements in this matrix are zero, this tells us that either node 1 or node 2 can serve as tandem nodes. Hence we can write:

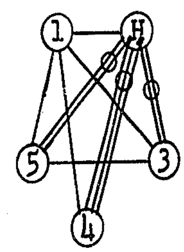
$$[T] = \begin{bmatrix} H & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad \text{or} \quad [T] = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & H & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

where H signifies the tandem node;

that is:



or



where the components of a branch are circled to show that they belong to one branch; and where each line represents paths in both directions, for clarity, in these diagrams only.

Of course, this method can be varied to suit particular cases: if $a_{ij} = 0$, then we do not need to eliminate the corresponding zero element in $[T]^n$. Other matrix operations with topological matrices are discussed in the literature, and may be applicable in future work.[18]

6.4 Second Tandem Doctrine

We now approach the question of merging traffic, that is, indirect traffic sharing paths with other traffic between nodes. The simplest case is that of one tandem node in a network.

In a three-node network, the simplest one for which the tandem possibility arises, Figure 7, all the traffic from node B to node C has to compete with all the other traffic from node A to node C on the paths from node A to node C:

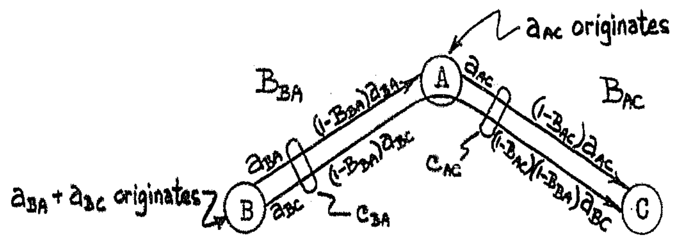


Figure 7

Naturally, all the traffic a_{BA} and a_{BC} originating at node B has to use the same paths, c_{BA}, to get a node A, regardless of whether it is destined for node A or node C. We will define $B_{BA} = B(c_{BA}, a_{BA} + a_{BC})$. As a result of blocking at node B, only a fraction $(1 - B_{BA})$ of the node B to node C calls reach node A; when this traffic is offered to the paths from node A to node C along with the direct traffic a_{AC}, further blocking is encountered. We will define $B_{AC} = B(c_{AC}, a_{AC} + (1-B_{BA})a_{BC})$. Hence only $(1-B_{AC})$ of the node B to node C traffic gets through to node C.

Now we can write:

Blocking effect felt at node B = B_B =

$$\frac{\text{Traffic offered - Traffic carried}}{\text{Traffic offered}}$$

$$= \frac{(a_{BC}+a_{BC}) - ((1-B_{BA})\, a_{BA} + (1-B_{AC})\,(1-B_{BA})a_{BC})}{a_{BA} + a_{BC}}$$

$$= B_{BA} + B_{BC}\left(\frac{(1-B_{BA})a_{BC}}{a_{BA} + a_{BC}}\right)$$

and:

Blocking effect felt at node A = $B_A = B_{AC}$

Under the first network doctrine we want $B_B = B_A$, hence by substituting and rearranging the blocking effect felt at node B we get:

$$B_B = B_{BA} \left(\frac{a_{BA} + a_{BC}}{a_{BA} + B_{BA}\, a_{BC}} \right)$$

Now B_B can be evaluated over all c_{BA}, $0 \leq c_{BA} \leq C$.

Since $B_A = B(c_{AC}, a_{AC} + (1-B_{BA})a_{BC}) = B_B$, and since we have the corresponding values of B_B and B_{BA}, then we can evaluate the corresponding values of c_{AC}. All this can be put in tabular form, after this fashion, Figure 8:

c_{BA}	$B_{BA} =$ $B(c_{BA}, a_{BA} + a_{BA})$	$B_B =$ $B_{BA}\left(\frac{a_{BA} + a_{BC}}{a_{BA} + B_{BA} a_{BC}}\right)$	$a^* =$ $a_{AC} + (1-B_{BA})a_{BC}$	c_{AC} from $B_B = B(c_{AC}, a^*)$	$C^* =$ $c_{BA} + c_{AC}$
0					
1					
2					
.					
.					
.					
C					

Figure 8

In this table, C* merely signifies the running total of paths in the two branches; C* will equal C, the given number of paths to be assigned, before c_{BA} reaches a value as large as C. Where c_{AC} does not happen to be an integer, we may choose the next highest integer. The solution at any C*, is, of course, found from the corresponding c_{BA} and c_{AC}, the path assignment and the corresponding B_B, which equals the minimum blocking $\bar{B}$ for this case. (It would be more precise to apply linear interpolation or a curvilinear fit to the (B_{AC}, c_{AC}) points, and so obtain more C* values.)

By extending the foregoing approach to a star network, where node A is the central tandem node of the star, it can be shown that for node i, $i \neq A$,

$$B_i = B_{iA} \left(\frac{\sum_j a_{ij}}{a_{iA} + B_{iA} \sum\limits_{\substack{j \\ (j \neq A)}} a_{ij}} \right)$$

and for node A, $j \neq A$,

$$B_{Aj} = B(c_{Aj}, a_{Aj} + \sum_{\substack{i \\ (i \neq A)}} (1-B_{iA})a_{ij})$$

Now it is simply a matter of applying the first network first node doctrine in order to relate the near-equal blocking to the given total branch capacity or to a range of total branch capacities. As before, we rate the performance of the system, $\bar{B}$, according to the maximum blocking in the set of blockings; but we can also give all the blockings in the set.

Unfortunately there is a drawback to this second tandem doctrine. For while, as $a_{ij} \rightarrow 0$, $B_i \rightarrow B_{iA}$, which is reasonable, we see that as $a_{iA} \rightarrow 0$, $B_i \rightarrow 1$, which is very bad. It would seem then, that this doctrine is most satisfactory when there is a preponderance of traffic terminating at the node at the centre of the star. But it is not likely to be at all suitable if the central node is serving chiefly as a tandem node. We will therefore devise another doctrine which will satisfy our definition of minimum blocking when there is no direct traffic between the points of the star network and the tandem node.

6.5 Third Tandem Doctrine

Let us suppose that we have a community of interest which is subject to two switchings in order to reach its destination: one at the origin and one at an intervening node. For three nodes, we have the following situation, Figure 9:

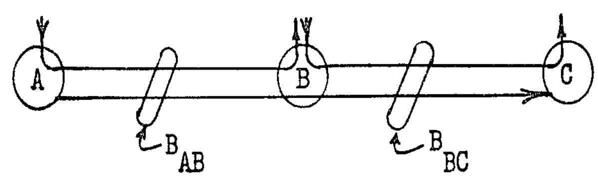


Figure 9

It is seen that:

$$a_{AC} \text{ becomes } a_{AC}(1-B_{AB}), \text{ becomes } a_{AC}(1-B_{AB})(1-B_{BC})$$

as we progress from origin to termination, and hence,

$$B_{AC} = \frac{a_{AC} - a_{AC}(1-B_{AB})(1-B_{BC})}{a_{AC}}$$

$$= \frac{a_{AC}B_{AB} + a_{AC}B_{BC} - a_{AC}\ B_{AB}\ B_{BC}}{a_{AC}}$$

$$= B_{AB} + B_{BC} - B_{AB}\ B_{BC}$$

$$\doteq B_{AB} + B_{BC},\ \text{for}\ B_{AB}\ B_{BC} \to 0$$

For reasonably small values of blocking, the approximation is acceptable. In fact, since $B_{AB} + B_{BC} > B_{AB} + B_{BC} - B_{AB}\ B_{BC}$, the approximation would result in conservative (safe) design. In tandem cases which follow, we will look for solutions chiefly at low levels of blocking, so that $a(1-B) \doteq a$, and $B.B \doteq 0$; at high levels, which are not of common interest anyway, accuracy is affected. (For accuracy at high levels of blocking, the model will need appropriate modification.)

Consider, then, a star network, in which all the traffic is pooled at the tandem node and is re-offered to the paths leading to the proper destination. This gives two stages of switching and two blockings. If we want the blocking (that is, resultant blocking) at the star points to be equal, then we can reason as follows.

Traffic originating at one point may leave the tandem node for several other points; therefore the blocking from the tandem node to all points must be equal. Also, traffic from several nodes may join at the tandem node so as to terminate at a single point; therefore the blocking from the points to the tandem node must be equal.

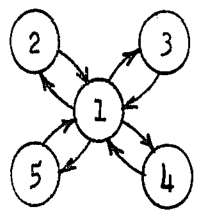
Using the first node doctrine, it is easy to distribute paths to the several branches originating at the tandem node so that the blocking on all branches is equal. Likewise, using the first network doctrine we can distribute paths amongst the star points, each having only one branch, so that all blockings are equal. Now we are faced with selecting one combination from each of those two sets of path assignments such that the sum of the corresponding blocking is minimum. However, this is familiar, because earlier, Section 4.7, we showed that, for equal traffic, equal blocking and minimum blocking are the same. Since the traffic in the point network approximately equals the traffic from the tandem node, all we need to do is to choose the two path distributions which give equal blocking and at the same time add up to the given number of paths to be assigned. The sum of the two sets of blockings which are equal and minimum is also minimum. The near-equal blocking conditions surrounding the integer constraint apply here as usual.

The solution is found, therefore, by assigning paths such that the blockings in every cell of the blocking matrix are near-equal, after the traffic has been pooled in the proper fashion. Then the blocking which applies to each community of interest is the sum of the appropriate pair of blockings.

More precisely, this is a first approximation, since the traffic offered at the tandem node is less than the traffic offered at the originating nodes by a fraction equal to the blocking. This solution will suffice for the small blocking values in the region of interest. The allocation of paths can, if desired, be adjusted by reducing the traffic offered at the tandem node by the blocking encountered at the originating nodes, and then applying the method of the second network doctrine to the traffic to, and and traffic from, the tandem node.

<u>Example</u>

Consider a star network:



$$[T] = \begin{bmatrix} H & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

for which the community of interest is:

$$[A] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{23} & a_{24} & a_{25} \\ 0 & a_{32} & 0 & a_{34} & a_{35} \\ 0 & a_{42} & a_{43} & 0 & a_{45} \\ 0 & a_{52} & a_{53} & a_{54} & 0 \end{bmatrix}$$

Since we can only have paths allowed by $[T]$, we must rearrange $[A]$ as follows:

$$[A^*] = \begin{bmatrix} 0 & \begin{matrix} a_{32}+ \\ a_{42}+ \\ a_{52} \end{matrix} & \begin{matrix} a_{23}+ \\ a_{43}+ \\ a_{53} \end{matrix} & \begin{matrix} a_{24}+ \\ a_{34}+ \\ a_{54} \end{matrix} & \begin{matrix} a_{25}+ \\ a_{35}+ \\ a_{45} \end{matrix} \\ \begin{matrix} a_{23}+ \\ a_{24}+ \\ a_{25} \end{matrix} & 0 & 0 & 0 & 0 \\ \begin{matrix} a_{32}+ \\ a_{34}+ \\ a_{35} \end{matrix} & 0 & 0 & 0 & 0 \\ \begin{matrix} a_{42}+ \\ a_{43}+ \\ a_{45} \end{matrix} & 0 & 0 & 0 & 0 \\ \begin{matrix} a_{52}+ \\ a_{53}+ \\ a_{54} \end{matrix} & 0 & 0 & 0 & 0 \end{bmatrix}$$

Now we know how to obtain, from Section 5.6:

$$[C] = \begin{bmatrix} 0 & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & 0 & 0 & 0 & 0 \\ c_{31} & 0 & 0 & 0 & 0 \\ c_{41} & 0 & 0 & 0 & 0 \\ c_{51} & 0 & 0 & 0 & 0 \end{bmatrix}$$

such that $\sum_i \sum_j c_{ij} = C$ given,

given that in:

$$[B^*] = \begin{bmatrix} 0 & B_{12} & B_{13} & B_{14} & B_{15} \\ B_{21} & 0 & 0 & 0 & 0 \\ B_{31} & 0 & 0 & 0 & 0 \\ B_{41} & 0 & 0 & 0 & 0 \\ B_{51} & 0 & 0 & 0 & 0 \end{bmatrix}$$

the B_{ij}'s indicated should be near-equal.

Having assigned the C_{given} accordingly, we can now rearrange $[B^*]$ to give the resultant B_{ij}'s for each community of interest, as follows:

$$[B] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & B_{13}+B_{21} & B_{14}+B_{21} & B_{15}+B_{21} \\ 0 & B_{12}+B_{31} & 0 & B_{14}+B_{31} & B_{15}+B_{31} \\ 0 & B_{12}+B_{41} & B_{13}+B_{41} & 0 & B_{15}+B_{41} \\ 0 & B_{12}+B_{51} & B_{13}+B_{51} & B_{14}+B_{51} & 0 \end{bmatrix}$$

Now we know how to obtain, from Section 5.6:

$$[C] = \begin{bmatrix} 0 & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & 0 & 0 & 0 & 0 \\ c_{31} & 0 & 0 & 0 & 0 \\ c_{41} & 0 & 0 & 0 & 0 \\ c_{51} & 0 & 0 & 0 & 0 \end{bmatrix}$$

such that $\sum_i \sum_j c_{ij} = C$ given,

given that in:

$$[B*] = \begin{bmatrix} 0 & B_{12} & B_{13} & B_{14} & B_{15} \\ B_{21} & 0 & 0 & 0 & 0 \\ B_{31} & 0 & 0 & 0 & 0 \\ B_{41} & 0 & 0 & 0 & 0 \\ B_{51} & 0 & 0 & 0 & 0 \end{bmatrix}$$

the B_{ij}'s indicated should be near-equal.

Having assigned the C_{given} accordingly, we can now rearrange $[B*]$ to give the resultant B_{ij}'s for each community of interest, as follows:

$$[B] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & B_{13}+B_{21} & B_{14}+B_{21} & B_{15}+B_{21} \\ 0 & B_{12}+B_{31} & 0 & B_{14}+B_{31} & B_{15}+B_{31} \\ 0 & B_{12}+B_{41} & B_{13}+B_{41} & 0 & B_{15}+B_{41} \\ 0 & B_{12}+B_{51} & B_{13}+B_{51} & B_{14}+B_{51} & 0 \end{bmatrix}$$

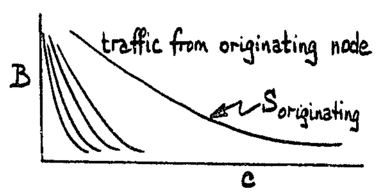
We can designate $\overline{B} = \underset{i,j}{\mathrm{Max}}(B_{1j}+B_{i1})$, $i \neq j$, $i \neq 1$, $j \neq 1$

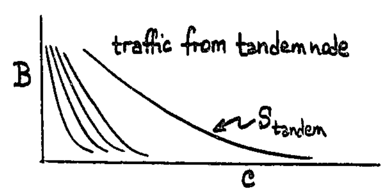
For the improved allocation, if required, we aim for:

$$\left[B^{**}\right] = \begin{bmatrix} 0 & b^* & b^* & b^* & b^* \\ b & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 0 \end{bmatrix}$$

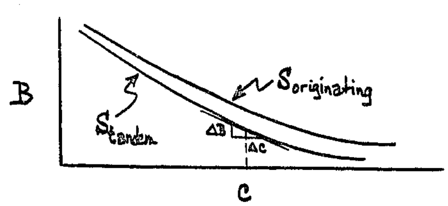
where b* and b are, severally, near-equal blockings, and we multiply the tandem row of $\left[A^*\right]$ by (1-b) in order to obtain traffic offered at the tandem node.

From the continuous Erlang B relationships:



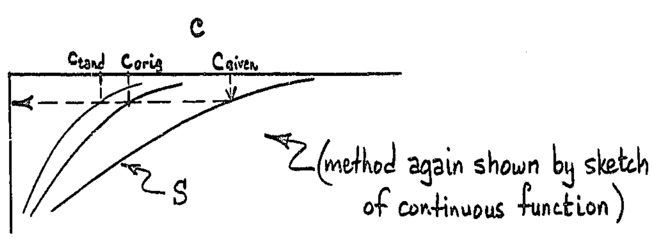


we can proceed to differentiate the two sum functions:



and then partition, by the method of Section 5.9:

$$\frac{\Sigma a}{\Sigma a_{orig} + \Sigma a_{tand}} \frac{dB}{dc}$$



using C_{given}. This leads back into the near-equal blocking with c_{tand} and c_{orig} for the two classes of traffic; and the required c_{ij}'s can be found by integer methods, or from the appropriate graphs of the originating node and tandem node B vs c relationships, by the method of Section 5.6.

6.6 Mixed Direct and Tandem Network

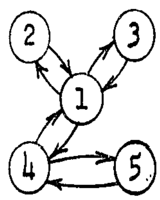
We are now in a position to take care of certain simple kinds of networks having both direct and tandem routes. We are still dealing with fixed routing, so that the traffic which can use a direct route does not have the option of using a tandem route, even if one exists; it simply cannot be switched onto the tandem paths.

If we consider a network in which the tandem routes are in star configurations, such that every community of interest which is not direct can be satisfied by using only one tandem node, and if we separate the network into a direct part and a tandem part, the method of solution becomes apparent.

We can treat the tandem part by the third tandem doctrine, and the direct part by the first network first node doctrine, simultaneously, all the while keeping the blocking on the direct part at twice the blocking on the tandem part. Thus, when the blocking on the tandem part is added, it will be nearly equal to the blocking on the direct part.

Example

Let us consider a mixed direct and tandem network as follows:



$$[T] = \begin{bmatrix} H & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad [A] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{23} & a_{24} & 0 \\ 0 & a_{32} & 0 & a_{34} & 0 \\ 0 & a_{42} & a_{43} & 0 & a_{45} \\ 0 & 0 & 0 & a_{54} & 0 \end{bmatrix}$$

From the previous example, we know that we can expect:

$$[A*] = \begin{bmatrix} 0 & a_{32}+a_{42} & a_{33}+a_{43} & a_{24}+a_{34} & 0 \\ a_{23}+a_{24} & 0 & 0 & 0 & 0 \\ a_{32}+a_{34} & 0 & 0 & 0 & 0 \\ a_{42}+a_{43} & 0 & 0 & 0 & a_{45} \\ 0 & 0 & 0 & a_{54} & 0 \end{bmatrix}$$

and that we want the form of $[B*]$ to be as follows:

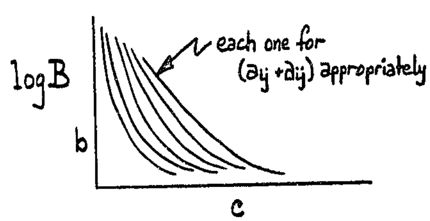
$$[B^*] = \begin{bmatrix} 0 & b & b & b & 0 \\ b & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 2b \\ 0 & 0 & 0 & 2b & 0 \end{bmatrix}$$

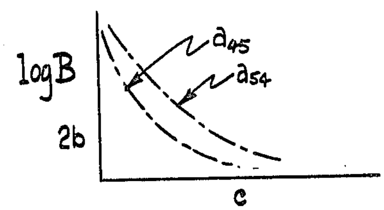
so that we will have a resultant:

$$[B] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2_b & 2_b & 0 \\ 0 & 2_b & 0 & 2_b & 0 \\ 0 & 2_b & 2_b & 0 & 2_b \\ 0 & 0 & 0 & 2_b & 0 \end{bmatrix}$$

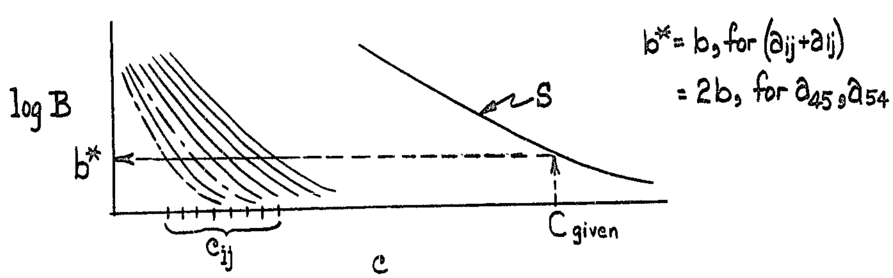
It is quite easy to operate with $[A^*]$ and $[B^*]$ to produce $[C]$ in the usual way; however, let us introduce a quick graphical method of handling this situation. Let us plot log $B(c, a_{ij})$ vs c for each a_{ij}. (Although the drawings below show the plot as a continuous function, we will understand that it represents the particular kind of step function we have described earlier).

On one plot, we can have the relationships for which we expect the blocking to be b, and on another, those for which the blocking is to be 2b:





Since these two plots are logarithmic with respect to B, we can superimpose them so as to maintain everywhere the required 1:2 relationship:



Thus we achieve the distribution of paths for near-equal blocking b*, (b* defined as above), and we can calculate the actual values of entries for the $\left[B^*\right]$ and hence $\left[B\right]$ matrices, as well as the value for $\overline{B} = \underset{i,j}{\mathrm{Max}}\ B_{ij}$ in $\left[B\right]$.

6.7 Fourth Tandem Doctrine

Let us now consider a network with two tandem nodes in series. Although we can work out patterns for equal blocking for each community of interest, we find that we encounter the sort of defect described for star networks, and that equality of blocking is often obtained only if we have high levels of blocking. It seems impractical to create bottlenecks in traffic just for the sake of equality. Thus, unless we dedicate paths, the best that we can do is to have the blocking additive.

One possible way of giving fair treatment to subscribers at all nodes is to make the blocking equal between all nodes attached to the same tandem node, and to have that blocking the same for both tandem nodes. Also, we can aim to have the blocking equal for traffic between any node on one tandem node and any node on the other tandem node.

Having already taken care of the case for a single tandem node, we know that the blocking on all branches between star points and their tandem should be equal. Now, between points on two different tandem nodes there is an extra branch; and it is clear that the two branches between the tandem nodes (one each way) must have equal blocking, but not necessarily the same as the blocking between tandem and points. This gives us an opportunity to effect an optimum distribution of paths, because we are not bound to relate the blocking between the tandem nodes to any other blocking.

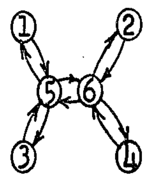
The solution lies, therefore, in producing two blocking vs path functions, each on an equal blocking doctrine - one for all the branches between star points and the tandem node to which they are directly connected, and one for the branches between the two tandem nodes.

Now, to these two functions we apply the sort of reasoning we used for the second node doctrine, where we sought the lowest weighted average blocking. We want to assign the paths so that there is no change to the number of paths in the tandem branches which we can make which will decrease the weighted average blocking of the network. We will, accordingly, produce the two corresponding weighted partial derivative functions, and add the abscissae to obtain the resultant relationship for total paths. For any given number of paths we can now work backwards and achieve the required path assignment. The blocking between nodes on two different tandem nodes is larger than the blocking between nodes on the same tandem node by an amount equal to the blocking between the two tandem nodes.

This principle can be extended to networks with many tandem nodes.

Example

Let us consider two tandem nodes in series:



$$[T] = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & H & 1 \\ 0 & 1 & 0 & 1 & 1 & H \end{bmatrix}$$

and let us assume a full community of interest, and pure tandem nodes,

$$[A] = \begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} & 0 & 0 \\ a_{21} & 0 & a_{23} & a_{24} & 0 & 0 \\ a_{31} & a_{32} & 0 & a_{34} & 0 & 0 \\ a_{41} & a_{42} & a_{43} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Now, when it comes to rearranging $[A]$, there are two situations to account for, and they correspond to whether the traffic passes one or both intervening tandem nodes. For instance, for one tandem node, such as a_{13} traffic, it is clear that it has to use paths from node 1 to node 5, and from node 5 to node 3:

$$[A^*] = \begin{bmatrix} 0 \longrightarrow a_{13} \\ \big\downarrow \\ a_{13} \end{bmatrix}$$

However, a_{12} traffic has to go via node 5 to get to node 6, and finally to node 2:

$$[A^*] = \begin{bmatrix} 0 \rightarrow a_{12} & \\ \downarrow & \\ a_{12} & \end{bmatrix}$$

and since $t_{52} = 0$, we have to distribute again:

$$[A^*] = \begin{bmatrix} 0 \rightarrow a_{12} \\ \downarrow \\ 0 \longrightarrow a_{12} \\ a_{12} \end{bmatrix}$$

Following this procedure, we get:

$$[A^*] = \begin{bmatrix} 0 & 0 & 0 & 0 & \Sigma a & 0 \\ 0 & 0 & 0 & 0 & 0 & \Sigma a \\ 0 & 0 & 0 & 0 & \Sigma a & 0 \\ 0 & 0 & 0 & 0 & 0 & \Sigma a \\ \Sigma a & 0 & \Sigma a & 0 & 0 & \Sigma a \\ 0 & \Sigma a & 0 & \Sigma a & \Sigma a & 0 \end{bmatrix}$$

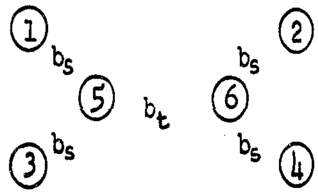
where each Σa represents the appropriately rearranged a_{ij}'s, (Σa's not necessarily equal).

The path matrix is:

$$[C] = \begin{bmatrix} 0 & 0 & 0 & 0 & c_{15} & 0 \\ 0 & 0 & 0 & 0 & 0 & c_{26} \\ 0 & 0 & 0 & 0 & c_{35} & 0 \\ 0 & 0 & 0 & 0 & 0 & c_{46} \\ c_{51} & 0 & c_{53} & 0 & 0 & c_{56} \\ 0 & c_{62} & 0 & c_{64} & c_{65} & 0 \end{bmatrix}$$

and we want $\sum_{ij}\sum c_{ij} = C$ given.

If we look at the network, we see that the blocking will be as follows:

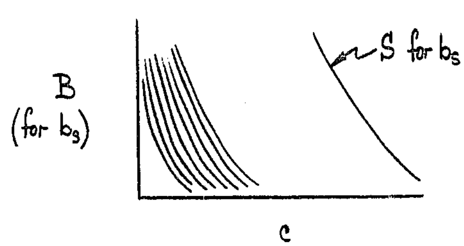


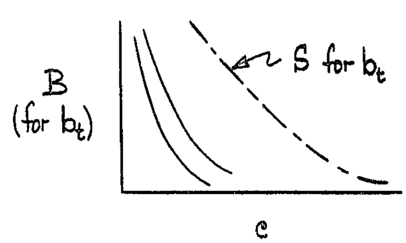
so that as we trace a route from node 1 to node 3, for example, the blocking B_{13} will be $b_s + b_s$; and from node 1 to node 2 the blocking B_{12} will be $b_s + b_t + b_s$. In terms of the blocking matrix, then:

$$\left[B*\right] = \begin{bmatrix} 0 & 0 & 0 & 0 & b_s & 0 \\ 0 & 0 & 0 & 0 & 0 & b_s \\ 0 & 0 & 0 & 0 & b_s & 0 \\ 0 & 0 & 0 & 0 & 0 & b_s \\ b_s & 0 & b_s & 0 & 0 & b_t \\ 0 & b_s & 0 & b_s & b_t & 0 \end{bmatrix}$$

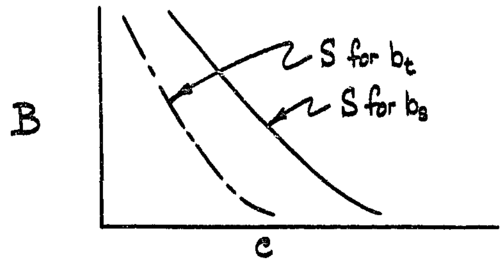
As usual, for integer values of paths, we will expect near-equal blocking.

We will begin with the continuous Erlang B relationships, and obtain sum functions:

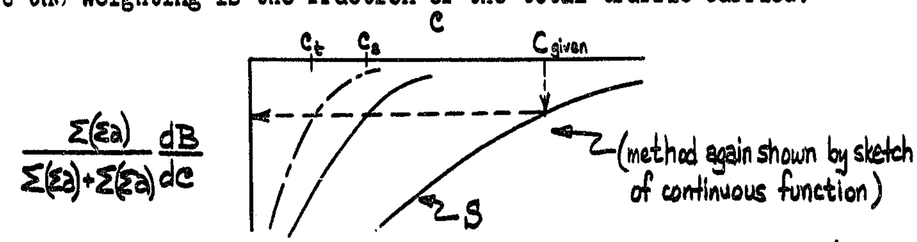




By combining these two graphs, we get the functions which we will differentiate with respect to c, at each integer value of c:



Next we plot the weighted partial derivatives of blocking against c, where the weighting is the fraction of the total traffic carried:



For our given C, we can obtain by the method of Section 5.9, the distribution of paths between the two tandem nodes, and between the ordinary nodes and the tandem nodes to which they are connected, c_t and c_s respectively. The c_t and c_s give us b_t and b_s, respectively, and the distribution of c_{ij} is found, by going back into the original B vs c graphs using integer methods and partitioning on the basis of near-equal blocking, Section 5.6.

The blocking at each originating node is found by reconstituting the $[B]$ matrix from the $[B^*]$ matrix just determined:

$$[B] = \begin{bmatrix} 0 & b_s+b_s+b_t & b_s+b_s & b_s+b_s+b_t & 0 & 0 \\ b_s+b_s+b_t & 0 & b_s+b_s+b_t & b_s+b_s & 0 & 0 \\ b_s+b_s & b_s+b_s+b_t & 0 & b_s+b_s+b_t & 0 & 0 \\ b_s+b_s+b_t & b_s+b_s & b_s+b_s+b_t & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where the b_s's and b_t's are the appropriate near-equal values. As usual, we designate $\bar{B} = \underset{ij}{\text{Max}}\ B_{ij}$, for all i,j in $\left[A \right]$.

(For a closer approximation, traffic offered at the tandem node can be reduced by the amount of blocking it suffered in getting to the tandem node, and the procedure can be repeated.)

6.8 Two Types of Tandem Nodes

We have said that tandem nodes could be of two types: those which serve to tandem traffic as well as to originate and terminate it; and those which only tandem it without originating or terminating any. So far the third and fourth tandem doctrines have assumed that the tandem nodes have served only the purely tandem function.

In order to accommodate the tandem node which originates and terminates traffic, we can simply split it into two parts, one of which is purely tandem, and the other, like an ordinary node with no tandem, connected to the purely tandem part by a pair of branches.

Now we obtain a solution for the assignment of a number of paths larger than the given number by a quantity equal to the number of paths on the said pair of branches. That is to say, we ignore the paths between the split parts of the tandem node, and solve for all the rest of the paths in the network to be equal to the given number to be assigned. However, we will assume that the blocking between the split parts applies in the usual way, due to switching within the tandem node.

6.9 Alternate Routing

Alternate routing is a doctrine for routing traffic which implies that calls from one point to another are offered to the routes between the two points such that if the first route to which the call is offered is busy, it is offered to another, and so on, until either a route is found, or a fixed number of unsuccessful tries has been made.

Usually the routes are tried in a strict order of preference; all calls are offered to the first route; calls not accepted on the first route are called overflow and are offered to the second route etc. The number of alternate routes varies considerably. Usually the first route is the direct one, or, if there is no direct one, then the shortest available tandem route, (i.e. fewest intervening nodes).

The nature of overflow traffic appears to be quite complicated; and to make matters more complicated, the second and subsequent routes are usually carrying the overflow traffic of many different first routes.

A simple example will serve to illustrate the problem. Suppose we offer a certain amount of traffic to a branch consisting of twleve paths, and that we observe the blocking to be 0.06, regardless of whether we offer it to the paths at random or whether we number the paths and always offer to the paths in order. If we offer it at random, we would expect all the paths to be equally busy, on the average; but if we offer it in order, we would expect the traffic to be packed in such a way that the first paths would be most busy and the last paths would be least busy, on the average.

Suppose we consider only the first four paths. If we offer the traffic in order, then by definition we are offering all the traffic to the first four paths. We would therefore find that the blocking on the first route consisting of the first four paths is 0.58, which means that 58% of the original offered traffic now has to be disposed of via an overflow route.

If we have 58% of the original traffic to offer, such that the blocking on it will be 0.103, (equivalent to 0.06 of the original traffic), and if we assume that the same relationship applies, we find that it will take only seven paths to carry it. But we know that it really takes eight more paths to carry the traffic, not seven; therefore the same relationship cannot apply. This means that we cannot make the same assumptions about the distribution of the overflow traffic that we make for the original traffic.

It is not the purpose of this paper to go into overflow traffic, nor to continue the studies made on it. We simply wish to adopt some reasonable hypothesis for dealing with traffic which is to be dispatched over more than one route, and which will be merging with other direct, tandem, and overflow traffic on some of the routes. Of course, purely original traffic is additive, so it would be convenient to divide the traffic and treat each part independently so that the parts are additive.

Let us assume that our objective remains the same in the alternate routing system as in the fixed routing system, and that we do not expect, just because we invoke alternate routing, that blocking will be reduced to nil. Let us consider a simple case of alternate routing between two nodes A and C, with node B serving as the tandem node. We know that for a certain blocking $B(c_{AC}, a_{AC})$, the offered traffic a_{AC} will require c_{AC} paths to be made available at node A. If we give the traffic a direct route, exactly c_{AC} paths between A and C will do. If we give the traffic a fixed tandem route via node C, then c_{AC} paths in each of the two branches will be needed, or $2c_{AC}$ paths altogether.

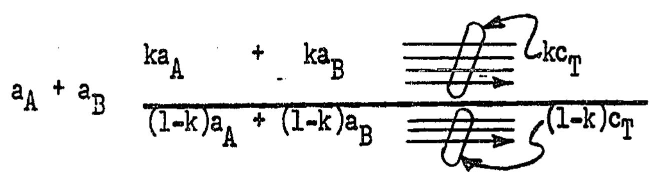
If we provide alternate routing, and allow both routes to be used, and distribute the traffic evenly on all paths but send a fraction k of it on the tandem route, then it is clear that we will have $(1-k)\ c_{AC}$ paths on the direct route and $2kc_{AC}$ paths on the two branches of the tandem route, or a total of $(1+k)\ c_{AC}$ paths. Since it is customary to pack the traffic on the direct route first, the actual blocking on the direct route can always be calculated later by offering the whole of the particular community of interest to the paths in the direct route only, that is, $B_{AC} = B((1-k)c_{AC}, a_{AC})$.

It is seen that alternate routing fits in between the two fixed routing cases, which are obtained when k is taken to its extremes $k = 0$ and $k = 1$.

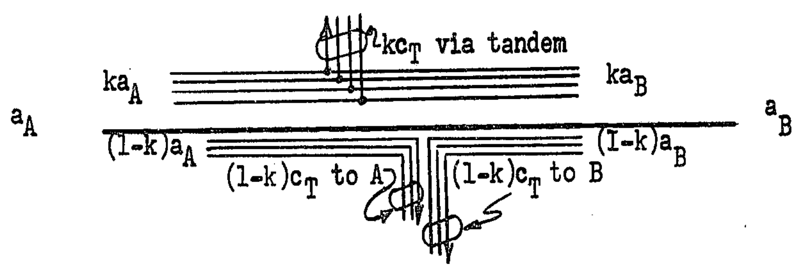
The above method will do for dedicated groups of paths very nicely. But, for merging traffic, the method of dealing with the traffic has to be carefully defined. Suppose we consider traffic originating at one node which is to go directly to two distant nodes, as well as via a common intervening node which provides an alternate route. At the originating node we have to adopt a distribution of traffic and paths somewhere between the extremes of direct routing and tandem routing.

Let us start as though we were offering all the traffic to the tandem route:

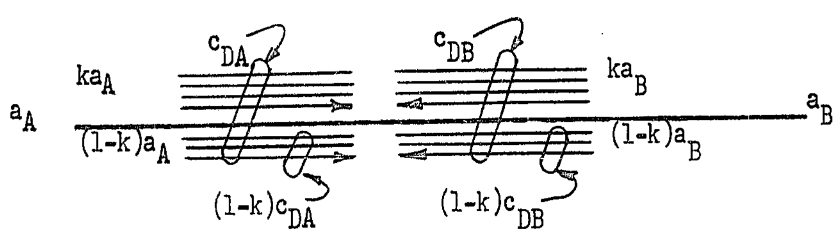
Then let us divide the traffic so that only a fraction k of it will go tandem, and the remainder will go direct:



Now we can revise the originating node as follows:

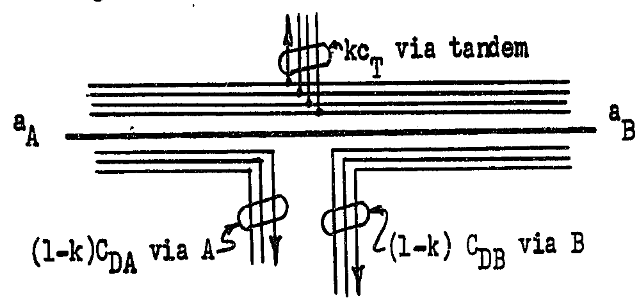


But the quantity $(1-k)\ c_T$ paths will be too large for each offering $(1-k)a_A$ or $(1-k)a_B$ of traffic alone, so we will reduce the number of paths from $(1-k)c_T$ to a quantity which will make each community of interest feel as if it were being offered to the same paths as before:



where: c_{DA} is the number of direct paths corresponding to a_A alone, and c_{DB} is the number corresponding to a_B.

Now we will put the appropriate parts together and arrive at the desired configuration



If we calculate the blocking on the direct paths as though all the traffic were offered to them first, we get $B_A((1-k)c_{DA}, a_A)$ and $B_B((1-k)c_{DB}, a_B)$, but still, on the overall system, both communities of interest experience the same blocking $B(c_T,(a_A + a_B))$.

This method allows us to control the fraction of traffic on the direct (or first) routes so that it is uniform throughout the system. To be more precise, we should say near-uniform, since the integer constraint will have some effect on the actual traffic distribution which finally results.

This is a safe and workable method which will suffice for the sort of broad comparisons of networks or doctrines which we wish to make, and it will lay the foundation for consideration of the much more complicated ways of distributing traffic among paths and branches which are actually used.

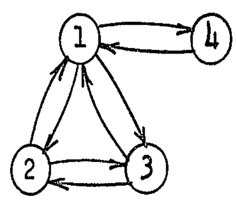
Next, we take this one more step, and consider merging traffic on the route from the tandem to the terminating node. Here we can adopt a method like that used for fixed routing in a network of mixed direct and tandem routes, say via one tandem node. We wish to achieve double the tandem blocking on the direct routes, so that the blocking on the two parts of the tandem route will add up to the same as the blocking on the direct routes. So we simply proceed to divide up the traffic as for alternate routing, and solve with appropriate blocking objectives as drawn from the patterns for fixed routing tandem doctrines.

Example

Alternate routing always implies tandem traffic, because it requires two or more possible routes for each community of interest, not more than one of which can be direct.

Let us consider a simple example of alternate routing:

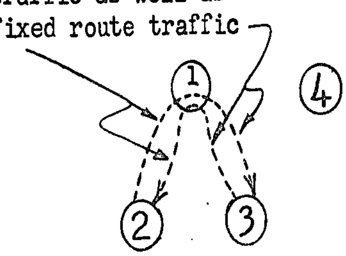
Node 1 is a pure tandem node



$$[T] = \begin{bmatrix} H & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

in which we will show the alternate routes with dotted lines:

Each branch carries alternate route traffic as well as fixed route traffic



$$[R] = \begin{bmatrix} H & 0 & 0 & 0 \\ 0 & 0 & \text{D; via 1} & \text{via 1} \\ 0 & \text{D; via 1} & 0 & \text{via 1} \\ 0 & \text{via 1} & \text{via 1} & 0 \end{bmatrix}$$

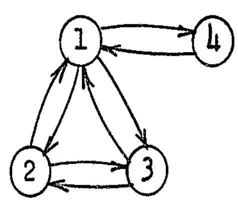
Next, we take this one more step, and consider merging traffic on the route from the tandem to the terminating node. Here we can adopt a method like that used for fixed routing in a network of mixed direct and tandem routes, say via one tandem node. We wish to achieve double the tandem blocking on the direct routes, so that the blocking on the two parts of the tandem route will add up to the same as the blocking on the direct routes. So we simply proceed to divide up the traffic as for alternate routing, and solve with appropriate blocking objectives as drawn from the patterns for fixed routing tandem doctrines.

<u>Example</u>

Alternate routing always implies tandem traffic, because it requires two or more possible routes for each community of interest, not more than one of which can be direct.

Let us consider a simple example of alternate routing:

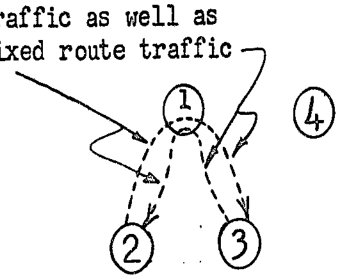
Node 1 is a pure tandem node



$$[T] = \begin{bmatrix} H & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

in which we will show the alternate routes with dotted lines:

Each branch carries alternate route traffic as well as fixed route traffic



$$[R] = \begin{bmatrix} H & 0 & 0 & 0 \\ 0 & 0 & \text{D; via 1} & \text{via 1} \\ 0 & \text{D; via 1} & 0 & \text{via 1} \\ 0 & \text{via 1} & \text{via 1} & 0 \end{bmatrix}$$

where: via 1: means tandem route only
D; via 1: means direct route first choice, tandem route via node 1 second choice.

Given the following traffic matrix:

$$[A] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & a_{23} & a_{24} \\ 0 & a_{32} & 0 & a_{34} \\ 0 & a_{42} & a_{43} & 0 \end{bmatrix}$$

and given that, if evenly distributed, half of the alternate route traffic should go direct, and half should go tandem, we can rearrange the traffic matrix as follows:

$$[A*] = \begin{bmatrix} 0 & \frac{a_{32}}{2} + a_{42} & \frac{a_{23}}{2} + a_{43} & a_{24} + a_{34} \\ \frac{a_{23}}{2} + a_{24} & 0 & \frac{a_{23}}{2} & 0 \\ \frac{a_{32}}{2} + a_{34} & \frac{a_{32}}{2} & 0 & 0 \\ a_{42} + a_{43} & 0 & 0 & 0 \end{bmatrix}$$

Our objective for $\left[B^*\right]$ must be of this sort:

$$\left[B^*\right] = \begin{bmatrix} 0 & b & b & b \\ b & 0 & 2b & 0 \\ b & 2b & 0 & 0 \\ b & 0 & 0 & 0 \end{bmatrix}$$

so that we will achieve equal blocking for the communities of interest:

$$\left[B\right] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 2b & 2b \\ 0 & 2b & 0 & 2b \\ 0 & 2b & 2b & 0 \end{bmatrix}$$

Now we have to assume that all the alternate route traffic is offered on all the alternate routes, and then allocate only the required fraction of paths needed:

$$\left[A^{**}\right] = \begin{bmatrix} 0 & a_{32}+a_{42} & a_{23}+a_{43} & a_{34}+a_{34} \\ a_{23}+a_{24} & 0 & a_{23} & 0 \\ a_{32}+a_{34} & a_{32} & 0 & 0 \\ a_{42}+a_{43} & 0 & 0 & 0 \end{bmatrix}$$

We now apply the method of solution for mixed direct and tandem networks, Section 6.6, but on account of the division of traffic on the alternate routes we must make a correction to the path matrix so obtained. Let us say that we can obtain an uncorrected path matrix $\left[C^{**}\right]$ from $\left[A^{*}\right]$, and that $\left[B^{*}\right]$ is the corresponding blocking matrix.

Let us organize the correction factors into a matrix $\left[M\right]$ in which $m_{ij} = \dfrac{a_{ij}^{*}}{a_{ij}^{**}}$. The corrected path matrix $\left[C\right]$ is obtained by multiplying each element of $\left[C^{**}\right]$, c_{ij}^{**}, by the corresponding element of $\left[M\right]$, m_{ij}. We calculate $C = \sum_i \sum_j c_{ij}$, and adjust b^* upwards or downwards until C = C given. When C = C given, then C represents the required path assignment, and we arrange $\left[B^{*}\right]$ into B to yield the actual blocking experienced by each community of interest. We designate $\overline{B} = \max_{ij} B_{ij}$. We also calculate, for the alternate route traffic only, the blocking on the direct route when all the traffic is offered to the direct route first:

$$B_{23} = B\ (c_{23},\ a_{23})$$

$$B_{32} = B\ (c_{32},\ a_{32})$$

It is common practice in telephony to engineer for a high value of blocking on the direct route, and low blocking on the final route. This is predicated on experience, economics, and non-coincident traffic. No more will be made of it here; it suffices to show that alternate routing can be dealt with.

CHAPTER VII

APPLICATIONS

This chapter will contain just a few examples which will show some useful applications and some interesting aspects of the method developed herein for finding optimal solutions for the design of communication networks for stochastic telephone traffic.

7.1 Economic Sensitivity

Broadly speaking, economic sensitivity is derived from the optimal blocking functions, $\overline{B}$ vs C, over all C in the range of interest for a given amount of traffic. This relationship is generally produced in the course of calculating the optimal blocking; and if it is not, it is obvious that it can be done without difficulty. From this we can find what reduction in blocking is obtained for (the cost of) an additional path.

7.2 Traffic Sensitivity

It is always of interest to see how a network responds to overload. We proceed by determining the optimal network for the given number of paths to be assigned, according to a selected doctrine. Then we can increase the traffic load, as represented by the traffic matrix, either uniformly or only locally, by a series of increments, such as 10%, 20%, 30%, etc., and simply calculate the blocking $B_{overload}$ using the original optimal path assignment.

Now we need some sort of reference against which to measure the effect of the overload on the network blocking. We can obtain this by determining the optimal path distribution for each of the overloads, on the basis of the same doctrine, and hence the minimum blocking $\bar{B}$.

By plotting $B_{overload}$ and $\bar{B}$ vs total traffic, we can assess the deterioration of the grade of service of the network due to overload. By doing this for several doctrines, we can see which doctrine results in the least deterioration for the given traffic requirement.

A similar technique can be used for severed branches, disabled tandem nodes, and other "catastrophies".

7.3 Cost

Having shown that we can assign paths in an optimal fashion, as though all paths were of equal cost, evidently we can also make the assignment to meet a fixed cost, if we apply the cost matrix.

7.4 Grid vs Hierarchical Alternate Routing

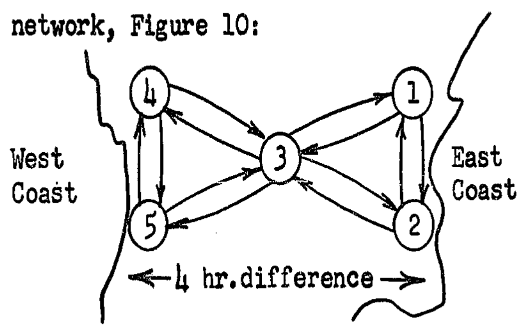
The methods in this paper enable grid and hierarchical methods of alternate routing to be compared. We are not immediately interested in random walk routing, since there are technical reasons for keeping the number of intervening nodes to a minimum. Also, we should tend to keep the alternate routing rules simple and general; if the decision-making at tandem nodes becomes too complicated, the cases become rather specific, and although they can be solved, some of the principles of network behaviour will become obscured.

7.5 Time Zone Effect

The rotation of the earth is one of the most obvious causes of load variation and of non-coincidence of busiest load periods in a network of continental proportions. The following illustration shows how the doctrines developed in Section 5.6 and 5.9 can be applied to the time zone problem.

Example

Let us consider an elementary transcontinental telephone network, Figure 10:



$$[T] = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & H & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Figure 10

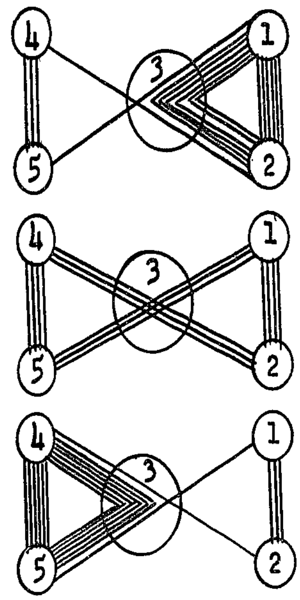
Let us assume that the traffic follows some sort of a business schedule like this:

West Coast Time	West Coast Traffic	Problem Period	East Coast Corresponding Time	East Coast Traffic
4 am	Not busy		8 am	Not busy
6	Not busy		10	Medium
8	Not busy	Early	12 noon	Very busy
10	Medium	Middle	2 pm	Medium
12 noon	Very busy	Late	4	Not busy
2 pm	Medium		6	Not busy
4	Not busy		8	Not busy
3 am	Not busy		7 am	Not busy

The problem period is boxed in, above; if we can accommodate this period, we can certainly accommodate the rest of the day.

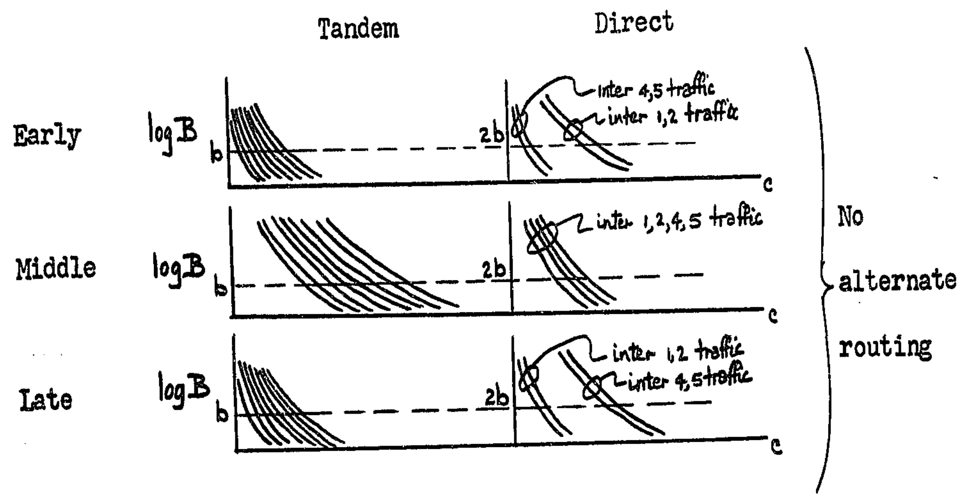
Graphically, the traffic flows in the problem period will be something like this, where the heavier lines indicate heavier traffic:

	West Coast	East Coast
Early	8 am	12 noon
Middle	10 am	2 pm
Late	12 noon	4 pm



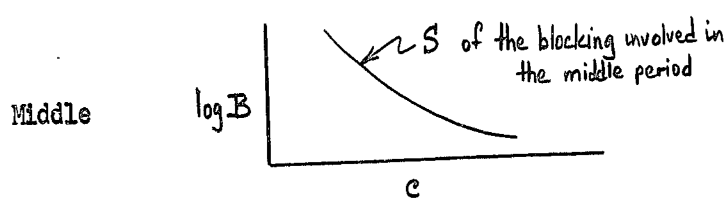
There is so much traffic between north and south nodes at the very busy time that it seems reasonable to pass a fraction of it via the tandem node, since, at that same time, the east-west traffic is extremely light, and the paths are well in excess of the number needed for the east-west traffic alone.

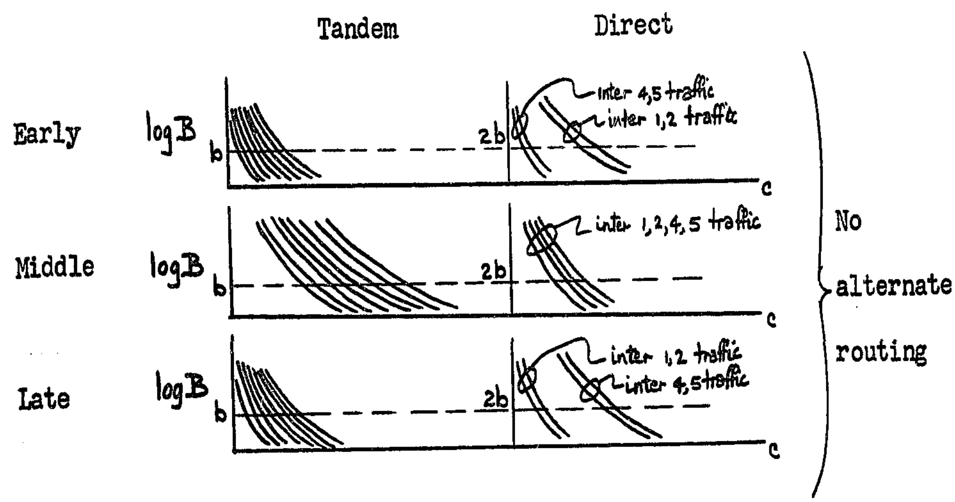
Now we can see a way of assigning the paths to this network. Let us try to do it in as few steps as possible first. Taking it for granted that we will use integer methods, mixed tandem and direct routing, and alternate routing doctrines, and aim for the least worst blocking possible (as well as near-equal blocking if possible), we can set up the following graphs:



We could adjust b, and 2b, up or down until the worst of the three cases was satisfied by the given number of paths, C. However, we have only achieved three separate solutions, not one, since there is not a correspondence between the c_{ij}'s of each of the three solutions. In order to bring about a correspondence, we will want to reduce the direct route traffic at the very busy times, by using alternate routing. This will increase the tandem traffic, naturally, and as a result the early and late cases will approach the middle case. What we are looking for is a correspondence such that the solution for one of the cases will also be a solution for the other two, and such that it is best solution of all possible solutions.

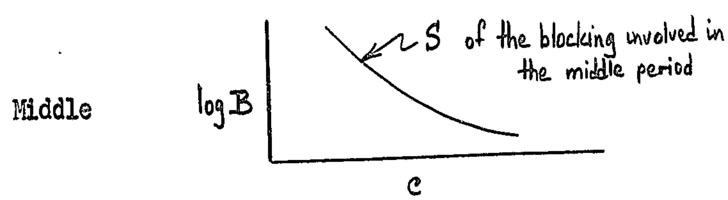
Let us assume first that alternate routing will not be required for the middle case. Hence we can get:





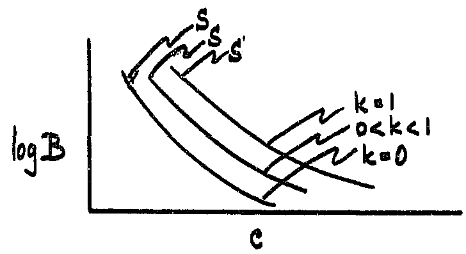
We could adjust b, and 2b, up or down until the worst of the three cases was satisfied by the given number of paths, C. However, we have only achieved three separate solutions, not one, since there is not a correspondence between the c_{ij}'s of each of the three solutions. In order to bring about a correspondence, we will want to reduce the direct route traffic at the very busy times, by using alternate routing. This will increase the tandem traffic, naturally, and as a result the early and late cases will approach the middle case. What we are looking for is a correspondence such that the solution for one of the cases will also be a solution for the other two, and such that it is best solution of all possible solutions.

Let us assume first that alternate routing will not be required for the middle case. Hence we can get:



For the alternate routing, we could first assume that the same k factor would apply to both alternate route traffic loads in the east. Thus we can get:

Early

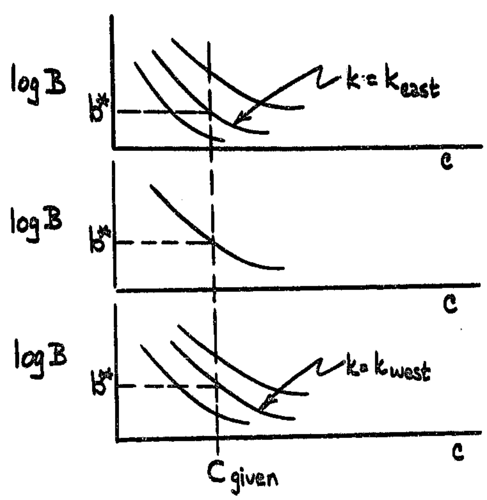


for the early period, and similarly for the late period. Assembling these and operating on them, we have:

Early

Middle

Late



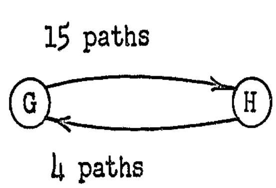
where k_{east} and k_{west} are chosen such that, at C_{given}, $b^*_{early} \doteq b^*_{middle} \doteq b^*_{late}$, and the $(c_{ij} + c_{ji})$'s correspond in the required sense.

(For a better solution, it may be necessary to use different k factors for each of the four alternate routes; and some alternate routing for the middle case, as well.)

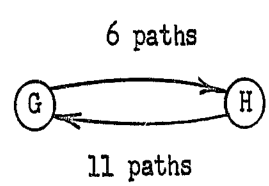
The cumbersome part of this procedure is the preparation of a few intermediate solutions for $0 < k < 1$, in order to locate the approximate magnitude of k east and k west. The graphical method of solution is, however, quite appropriate for manual calculations; with an electronic computer, the approximations and successive solutions can easily be made to suit as close a fit as one wishes to test for.

Now, for each pair of nodes, we have determined that a certain total number of paths is required in both branches together, but in the various cases the number in each branch may differ. We should therefore indicate, in some way, the number of paths which may be required to carry traffic in either direction, depending on the time of day; there are two-way paths. It is appropriate to introduce a third branch between nodes, as required, which is non-oriented, and which we shall call a two-way branch, consisting of the two-way paths. (A two-way path can only be used by one call at a time; but, when it is idle, it can be secured from either one end or the other, for use by a call.)

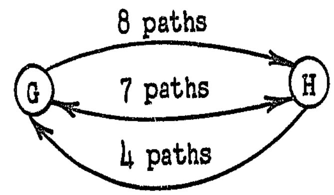
Let us say that we want to keep the capacity of the two-way branch such as to minimize the cost of required paths, since a two-way path, while less expensive than two one-way paths, is more expensive than a one-way path. For example, if we have



and



in two different cases, we can satisfy both cases if we allow:



Calculation:	max. to right = max (15, 6)	= 15
	max. to left = max (4, 11)	= 11
	total	26
	max between nodes = max. (19, 17)	= 19
	difference = two-way paths	= 7
	paths to right = 15-7	= 8
	paths to left = 11-7	= 4

By applying this method of calculating two-way path requirements to the intermediate results, we can provide a solution to the time zone problem.

CHAPTER 8

CONCLUSION

8.1 Summary

The optimal assignment of integer branch capacities to a network of given topological structure, carrying stochastic traffic, where no queuing is permitted, can be accomplished by means of the partitioning process developed in this work. This method of assignment makes another bridge between the area of multicommodity flow network theory, and its practical application in such fields as telephone communication and operational research.

The notion of blocking, or loss, enables us to set up a criterion for judging optimality when we are dealing with stochastic, rather than steady, flow. We say that the assignment is optimal when the blocking is minimum, subject to the fixed network capacity constraint. Two definitions of minimum blocking are used: near-equal, and weighted average. It is in the use of these blocking doctrines that our approach to this problem area of network design differs from other reported approaches to the same area, because we provide for the loss which occurs with stochastic traffic; and furthermore, we can control the rejected traffic. Also, the careful choice of blocking doctrines makes linear programming methods of solution unnecessary. The partitioning process, and its graphical counterpart, produce integer values of branch capacities immediately; this is a necessary attribute when non-integer capacities are not meaningful. The method is adaptable to the electronic computer, and obviates the need for simulation runs. The notion of matrix representation of the network parameters is useful in organizing the network and studying its behaviour, as well as in dealing with merging traffic.

A number of examples are given to show practical applications of the general solution.

A simple, possibly original, method of determining essential tandem nodes in a network is included in this work; and likewise, a method is included of reducing the total cost of two oppositely oriented branches between a pair of nodes by supplying a third non-oriented branch.

8.2 Future Investigations

The partitioning method and blocking doctrines described in this work are useful notions in dealing with network problems, and in future studies they can be refined and made more widely applicable in network systhesis. Areas leading from this work which suggest themselves for future investigation include:

Combining the partitioning method with other techniques to produce topological systhesis as well;

Relating the method of solution by Lagrangian multipliers (which may not produce integer results);

Solving for fixed costs in general, especially non-linear costs;

Investigating and applying helpful matrix operations to tandem networks and routing doctrines;

Trying to apply some useful elements of the method to switching networks;

Generalizing for other traffic distributions, and other blocking doctrines;

Studying local congestion and survivability of networks, and constraints on blocking or branch capacity.

APPENDIX I

TABLE OF THE ERLANG LOSS FORMULA

Formula: $$B = \frac{\frac{A^n}{n!}}{1 + A + \frac{A^2}{2!} + \dots + \frac{A^n}{n!}}$$

where B is the loss, n the number of circuits, and A the traffic intensity in erlangs.

The table gives the values of B.

A \ n	1	2	3	4	5	6	7	8	9	10
0.05	047 619	001 189	000 020							
0.10	090 909	004 525	000 151	000 004						
0.15	130 435	009 662	000 484	000 018	000 001					
0.20	166 667	016 393	001 092	000 055	000 002					
0.25	200 000	024 390	002 028	000 127	000 006					
0.30	230 769	033 457	003 335	000 250	000 015	000 001				
0.35	259 259	043 401	005 038	000 441	000 031	000 002				
0.40	285 714	054 054	007 156	000 715	000 057	000 004				
0.45	310 345	065 270	009 689	001 090	000 098	000 007				
0.50	333 333	076 923	012 658	001 580	000 158	000 013	000 001			
0.55	354 839	088 905	016 038	002 200	000 242	000 022	000 002			
0.60	375 000	101 124	019 824	002 965	000 356	000 036	000 003			
0.65	393 939	113 499	024 001	003 885	000 505	000 055	000 005			
0.70	411 765	125 964	028 553	004 972	000 696	000 081	000 008	000 001		
0.75	428 571	138 462	033 457	006 234	000 934	000 117	000 013	000 001		
0.80	444 444	150 943	038 694	007 679	001 227	000 164	000 019	000 002		
0.85	459 459	163 369	044 240	009 313	001 581	000 224	000 027	000 003		
0.90	473 684	175 705	050 072	011 141	002 001	000 300	000 039	000 004		
0.95	487 179	187 923	056 167	013 164	002 495	000 395	000 054	000 006	000 001	
1.00	500 000	200 000	062 500	015 385	003 067	000 511	000 073	000 009	000 001	

From: C. Palm, Table of the Erlang Loss Formula, Royal Board of Swedish Telecommunications, (2nd ed., 1954).

APPENDIX II

THE DERIVATIVES OF ERLANG'S B FORMULA

Example of the first derivatives where E_T is the loss, T the number of circuits, and a the traffic in erlangs.

$E_T = .010$

T	a	$\frac{T}{a}$	$\frac{\partial E_T}{\partial a} / E_T$	$-\frac{\partial E_T}{\partial T} / E_T$	$\frac{dT}{da}$
1	.0101	99.0000	98.0100	5.0182	19.5310
2	.1526	13.1067	12.1167	2.8044	4.3206
3	.4555	6.5864	5.5964	2.0454	2.7361
4	.8694	4.6008	3.6108	1.6498	2.1886
5	1.3608	3.6743	2.6843	1.4025	1.9139
6	1.9090	3.1430	2.1530	1.2312	1.7487
7	2.5009	2.7989	1.8089	1.1044	1.6380
8	3.1276	2.5579	1.5679	1.0061	1.5584
9	3.7825	2.3794	1.3894	.9274	1.4982
10	4.4612	2.2416	1.2516	.8626	1.4509
11	5.1599	2.1318	1.1418	.8082	1.4128
12	5.8760	2.0422	1.0522	.7618	1.3813
13	6.6072	1.9676	.9776	.7216	1.3548
14	7.3517	1.9043	.9143	.6863	1.3322
15	8.1080	1.8500	.8600	.6552	1.3126
16	8.8750	1.8028	.8128	.6274	1.2954
17	9.6516	1.7614	.7714	.6025	1.2803
18	10.4369	1.7247	.7347	.5799	1.2668
19	11.2301	1.6919	.7019	.5594	1.2548
20	12.0306	1.6624	.6724	.5406	1.2438
21	12.8378	1.6358	.6458	.5234	1.2340
22	13.6513	1.6116	.6216	.5074	1.2249
23	14.4705	1.5894	.5994	.4927	1.2166
24	15.2950	1.5691	.5791	.4790	1.2090
25	16.1246	1.5504	.5604	.4662	1.2020
26	16.9588	1.5331	.5431	.4543	1.1955
27	17.7974	1.5171	.5271	.4431	1.1894
28	18.6402	1.5021	.5121	.4326	1.1838
29	19.4869	1.4882	.4982	.4227	1.1785
30	20.3373	1.4751	.4851	.4134	1.1735
31	21.1912	1.4629	.4729	.4046	1.1688
32	22.0483	1.4514	.4614	.3962	1.1644
33	22.9087	1.4405	.4505	.3883	1.1603
34	23.7720	1.4303	.4403	.3807	1.1564
35	24.6381	1.4206	.4306	.3735	1.1527
36	25.5070	1.4114	.4214	.3667	1.1492
37	26.3785	1.4027	.4127	.3601	1.1458
38	27.2525	1.3944	.4044	.3539	1.1426
39	28.1288	1.3865	.3965	.3479	1.1396
40	29.0074	1.3790	.3890	.3422	1.1367
41	29.8882	1.3718	.3818	.3367	1.1339
42	30.7712	1.3649	.3749	.3314	1.1313
43	31.6561	1.3583	.3683	.3263	1.1288
44	32.5430	1.3521	.3621	.3214	1.1263
45	33.4317	1.3460	.3560	.3167	1.1240
46	34.3223	1.3402	.3502	.3122	1.1218
47	35.2146	1.3347	.3447	.3078	1.1196
48	36.1086	1.3293	.3393	.3036	1.1176
49	37.0042	1.3242	.3342	.2996	1.1156
50	37.9014	1.3192	.3292	.2956	1.1136

From: H. Akimaru and T. Nishimura, "The Derivatives of Erlang's B Formula", Review of the Electrical Communication Laboratory, 11, (1963), 428–445

APPENDIX II

THE DERIVATIVES OF ERLANG'S B FORMULA

Example of the first derivatives where E_T is the loss, T the number of circuits, and a the traffic in erlangs.

$E_T = .010$

T	a	$\frac{T}{a}$	$\frac{\partial E_T}{\partial a}/E_T$	$-\frac{\partial E_T}{\partial T}/E_T$	$\frac{dT}{da}$
1	.0101	99,0000	98,0100	5,0182	19,5310
2	.1526	13,1067	12,1167	2,8044	4,3206
3	.4555	6,5864	5,5964	2,0454	2,7361
4	.8694	4,6008	3,6108	1,6498	2,1886
5	1,3608	3,6743	2,6843	1,4025	1,9139
6	1,9090	3,1430	2,1530	1,2312	1,7487
7	2,5009	2,7989	1,8089	1,1044	1,6380
8	3,1276	2,5579	1,5679	1,0061	1,5584
9	3,7825	2,3794	1,3894	.9274	1,4982
10	4,4612	2,2416	1,2516	.8626	1,4509
11	5,1599	2,1318	1,1418	.8082	1,4128
12	5,8760	2,0422	1,0522	.7618	1,3813
13	6,6072	1,9676	.9776	.7216	1,3548
14	7,3517	1,9043	.9143	.6863	1,3322
15	8,1080	1,8500	.8600	.6552	1,3126
16	8,8750	1,8028	.8128	.6274	1,2954
17	9,6516	1,7614	.7714	.6025	1,2803
18	10,4369	1,7247	.7347	.5799	1,2668
19	11,2301	1,6919	.7019	.5594	1,2548
20	12,0306	1,6624	.6724	.5406	1,2438
21	12,8378	1,6358	.6458	.5234	1,2340
22	13,6513	1,6116	.6216	.5074	1,2249
23	14,4705	1,5894	.5994	.4927	1,2166
24	15,2950	1,5691	.5791	.4790	1,2090
25	16,1246	1,5504	.5604	.4662	1,2020
26	16,9588	1,5331	.5431	.4543	1,1955
27	17,7974	1,5171	.5271	.4431	1,1894
28	18,6402	1,5021	.5121	.4326	1,1838
29	19,4869	1,4882	.4982	.4227	1,1785
30	20,3373	1,4751	.4851	.4134	1,1735
31	21,1912	1,4629	.4729	.4046	1,1688
32	22,0483	1,4514	.4614	.3962	1,1644
33	22,9087	1,4405	.4505	.3883	1,1603
34	23,7720	1,4303	.4403	.3807	1,1564
35	24,6381	1,4206	.4306	.3735	1,1527
36	25,5070	1,4114	.4214	.3667	1,1492
37	26,3785	1,4027	.4127	.3601	1,1458
38	27,2525	1,3944	.4044	.3539	1,1426
39	28,1288	1,3865	.3965	.3479	1,1396
40	29,0074	1,3790	.3890	.3422	1,1367
41	29,8882	1,3718	.3818	.3367	1,1339
42	30,7712	1,3649	.3749	.3314	1,1313
43	31,6561	1,3583	.3683	.3263	1,1288
44	32,5430	1,3521	.3621	.3214	1,1263
45	33,4317	1,3460	.3560	.3167	1,1240
46	34,3223	1,3402	.3502	.3122	1,1218
47	35,2146	1,3347	.3447	.3078	1,1196
48	36,1086	1,3293	.3393	.3036	1,1176
49	37,0042	1,3242	.3342	.2996	1,1156
50	37,9014	1,3192	.3292	.2956	1,1136

From: H. Akimaru and T. Nishimura, "The Derivatives of Erlang's B Formula", Review of the Electrical Communication Laboratory, 11, (1963), 428–445

APPENDIX III

TABLE OF THE FIRST DERIVATIVE OF THE ERLANG LOSS FORMULA WITH RESPECT TO PATHS

The table gives approximate values of $\frac{\partial B}{\partial c}$ where B is the blocking according to the Erlang loss formula, c is the number of paths (circuits), and a is the traffic in erlangs.

a \ c	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1.0		-.219	-.0923	-.0297	-.00744	-.00150								
2.0		-.228	-.152	-.0872	-.0416	-.0166	-.00562	-.00162	-.000411					
3.0		-.202	-.162	-.118	-.0769	-.0441	-.0220	-.00960	-.00366	-.00124	-.000378	-.000104		
4.0		-.175	-.152	-.126	-.0970	-.0682	-.0433	-.0247	-.0125	-.00569	-.00233	-.000867	-.000293	-.0000910
5.0		-.152	-.139	-.123	-.103	-.0820	-.0610	-.0418	-.0258	-.0146	-.00748	-.00349	-.00148	-.000582
6.0		-.134	-.130	-.115	-.102	-.0875	-.0715	-.0550	-.0395	-.0261	-.0159	-.00889	-.00459	-.00216
7.0		-.119	-.114	-.107	-.0980	-.0880	-.0760	-.0635	-.0502	-.0372	-.0258	-.0167	-.00998	-.00554
8.0		-.107	-.103	-.0980	-.0925	-.0855	-.0770	-.0675	-.0570	-.0459	-.0353	-.0253	-.0171	-.0108

REFERENCES

1. Kim, W.H., and Chien, R.T., "Topological Analysis and Synthesis of Communication Networks", Columbia Press, (1962)

2. Wernander, M.A., "Systems Engineering for Communications Networks", IEEE Trans. on Comm. and Electronics, (1964), 603-611

3. Brockmeyer, E., Halstrom, H.L., and Jensen, A., "The Life and Works of A.K. Erlang", Acta Polytechnica Scandinavica, (1960)

4. Syski, R., "Introduction to Congestion Theory in Telephone Systems", Oliver & Boyd, London, (1960)

5. "Switching Systems", American Telephone and Telegraph Company, New York, (1961)

6. Ford, L.R., and Fulkerson, D.R., "Flows in Networks", Princeton University Press, Princeton, N.J. (1962)

7. Ford, L.R., and Fulkerson, D.R., "Maximal Flow Through a Network", Can. J. of Math., 8, (1956), 399-404

8. Dantzig, G.B., and Fulkerson, D.R., "On the Max-Flow Min-Cut Theorem of Networks", Annals of Math. Studies, 38, (1956), 215-221

9. Gomory, R.E., and Hu, T.C. "Synthesis of a Communication Network", J. Soc. Ind. Appl. Math., 12, (1964), 348-369

10. Kalaba, R.E., and Juncosa, M.L., "Optimal Design and Utilization of Communication Networks", Management Science, 3, (1956), 33-44

11. Tang, D.T., "Communication Networks with Simultaneous Flow Requirements", IRE Trans. on Circuit Theory, CT9, (1962), 176-182

12. Hakimi, S.L., "Simultaneous Flows Through a Communication Network", IRE Trans. on Circuit Theory, CT9, (1962), 169-175

13. Ling, S.T., Tezuka, Y., and Kasahara, Y., "Optimal Allocation of Channels in an Alternate Route Communication Network", IEEE Trans. on Comm. Syst., (1964), 185-190

14. Kleinrock, L., "Communication Nets", McGraw-Hill, New York, (1964)

15. Gomory, R.E., "Outline of an Algorithm for Integer Solutions to Linear Programs", Bull. Amer. Math. Soc., 64, (1958), 275-278

16. Akimaru, H., and Nishimura, T., "The Derivatives of Erlang's B Formula", Rev. of the Elec. Comm. Lab., 11, (1963), 428-445

17. Akimaru, H., and Nishimura, T., "The First Derivatives of Erlang's B Formula", Elec. Comm. Lab. Tech. J., Extra Issue No. 8, (1962), 1-81, (in Japanese)

18. Kemeny, J.G., Snell, J.L., and Thompson, G.L., "Introduction to Finite Mathematics", Prentice-Hall, Englewood Cliffs, N.J., (1956)

VITA

Name:	David Wingate McLimont
Born:	Quebec, Canada, June 23, 1925
Education:	
Primary:	Montreal, Canada
Secondary:	Bishop's College School, Lennoxville, P.Q.
University:	McGill University, Montreal, Canada
Course:	Chemical Engineering
Degree	B. Eng.

END OF
REEL