Lost Call Traffic in Circuit Switched Networks

By
Rami A. Al-Hmouz

Supervisor
Professor Jamil Ayoub

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Electrical Engineering/Communication

Faculty of Graduate Studies
University of Jordan

January 2002

This thesis was successfully defined and approved on:...Jan. 16, 2002...........

Examination Committee	**Signature**
Prof. Jamil N. Ayoub, Chairman	....Jamil Ayoub
Prof. Mohamed K. Abdelaziz, Member	M.K. Abdelaziz..................
Prof. Labeeb Al-Khadra, Member	L. Khadra..........................
Dr. Ibrahim M. Mansour, Member	Ibrahim Mansour..................

Dedication

To
my parents

Without whose sacrifices none of this would be possible

Acknowledgment

I mostly thank my supervisor Dr. Jamil Ayoub who spared no effort whatsoever in helping me through each and every page of this thesis and whose encouragement and support made this thesis come out to its present form. Thank you Dr. Jamil for helping me and believing in my capabilities.

Contents

Tables List

Figures List

Appendix

Lost Call Traffic in Circuit Switched Networks

By
Rami A. Al-Hmouz

Supervisor
Professor Jamil Ayoub

Abstract

This thesis is concerned with studying the performance of circuit switched telephone networks which have two levels of switching under conditions of random link failure, and subject to certain routing disciplines. Lost call traffic is taken as a measure of performance which need to be evaluated.

Previous studies on lost call traffic are reviewed and their methods are outlined. Such studies were restricted to networks with one level of switching. These methods are extended to networks, with two levels of switching, which might represent local and national networks that are commonly employed in relatively small countries like Jordan.

Since simulation studies of such networks under condition of random failure require the consideration of a large number of network states, it is assumed that the network links have high availability or very small failure probability. As such, only the most probable states to occur will be sufficient to consider under failure simulation. Consequently, states

considered are those which admit no link failure, one-link failure at a time and two-link failures at a time.

Lost call traffic is calculated for each network state employing Erlang Formula B, subject to certain alternative routing discipline. The results are then weighted by the occurrence probabilities of the respective states to produce a statistical average for the lost call traffic parameter.

A second method to calculate the average lost call traffic in the presence of random failure by generating network states randomly in conformity with a sampling rule that employs random number generator applied to the network links to decide which links in a given state are failing. This method holds some advantage when links in a network have high probability of failure.

These methods have been applied to example networks, including that of the Jordan telephone network. The two methods were applied, and both give close results.

CHAPTER I

Introduction to Network Reliability

1.1 Introduction

Our study will focus on the reliability of circuit switched networks using different routing strategies under conditions of random failure effected on each link in the network. Lost call traffic in Erlangs is proposed as a measure for this reliability.

One possible reliability measure for a circuit switched network could be the ability of handling the average busy hour traffic. This measure is not enough since the busy hour traffic for the switching centers in some countries don't coincide. In general, there are three aspects to consider in designing a communication network that affects its reliability. These include network topology, traffic constraint and the routing design (Medhi, 1994) (Gravish, 1992) (Faratta, 1978).

When simulation is used to evaluate a reliability measure for a network under conditions of random failure, operating network states need to be considered. A network of L links, any of which could be subject to failure, has 2^L network states. While this presents a large number of states, the network usually work in few of these states. Li (1984) introduced a method to determine these states on the basis of selecting the most probable states. This approach was utilized in (Sanso, 1991) to calculate the lost call traffic for a network that has high link availability and allowing only one link to fail at a time. This method was extended in (Ayoub, 1997) by allowing two links to fail at a time, thereby extending the method to consider networks with relatively lower link availability.

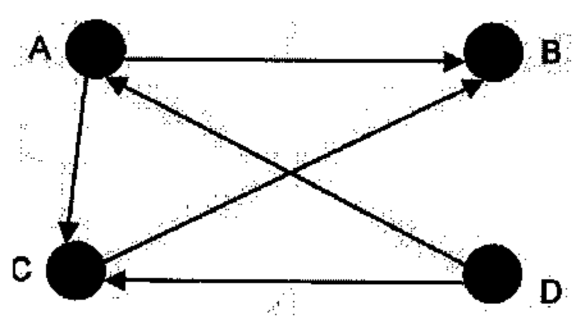
The above studies were also limited to studying networks with one switching hierarchy. The present study will extend the above approaches to consider a more realistic circuit switched networks which include two switching hierarchy that is, one with local and national interconnections.

The network is assumed to have two levels of switching hierarchy in which one level represents interconnection among local switches as in any city environment, while the other refers to trunk connections between cities. Different routing approaches will be assumed. Failure is assumed to occur randomly, but restricted to affect at most two links at any time.

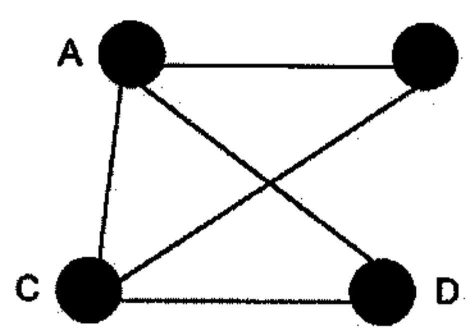
Congestion will occur near the site of a major failure, or when a link is fully loaded. Hence, a certain procedure must be added to the network in order to adapt to the new condition. In our case, we consider different routing techniques to calculate lost call traffic which give us a measure of the reliability of the network.

1.2 Basic Graph Terminology

A graph consists of a set of points or nodes, and a set of edges that link together these nodes. A simple real example of a graph would be a communication network, where the switching centers are the nodes and the links between them are the edges connecting the nodes. A graph can take on many forms: directed or undirected. A directed graph is one in which the direction of any given edge is defined. Conversely, in an undirected graph you can move in both directions between nodes as shown in Fig. (1-1). The edges can also be weighted or unweighted. Using the previous example, weights can be thought of as the distance between switches.



(a) Direct Graph.



(b) Undirect Graph.

Fig. (1-1). Graph Forms.

The type of a graph largely depends upon the features of its components, namely the attributes of the nodes and edges. A node within a graph may or may not have a label assigned to it. Similarly, an edge may have a label, weight, and/or direction associated with it. As an example, a mixed graph is one that contains both directed and undirected edges, while a null graph is one that contains only isolated nodes (i.e. no edges).

An edge in a graph that joins two nodes is said to be incident to both nodes. Furthermore, the number of distinct edges that are incident to a node determines the degree of that node.

Two edges in a graph are termed adjacent if they connect to the same node. Similarly, two nodes are termed adjacent if they are connected by the same edge. A self loop is an edge that links a node to itself. A simple graph is one that contains no self loops or parallel edges, where more than one edge connects two given nodes. A multigraph is a graph that contains multiple edges. Finally, a complete graph is a simple graph in which every pair of nodes are adjacent (Cravis, 1981).

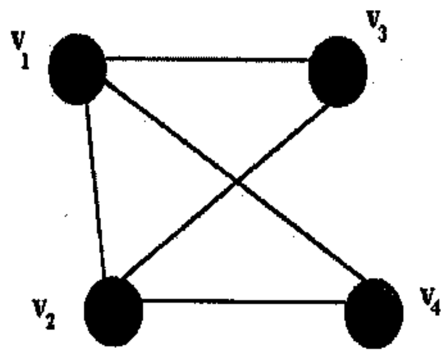
In general, if the set of nodes is N and the set of links is L, we refer to the graph as G(N,L). Graphs can also be represented in the form of matrices. A connection matrix is defined as follows: Let G be a graph with N nodes that are assumed to be ordered from v_1 to v_n. The N x N matrix A, in which

$a_{ij} = 1$ if there exists an edge connecting v_i to v_j

$a_{ij} = 0$ otherwise

is called a connection matrix.

Consider the following undirected graph G (4,5) shown in Fig. (1-2), and its equivelant connection matrix [T].



$$[T] = \begin{array}{c c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} & \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} & \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \end{array}$$

Fig. (1-2). Undirected Graph.

From the previous graph, if a path of length 1 exists from one node to another (i.e. the two nodes are adjacent), then there must be an entry of 1 in the corresponding position in the matrix. For example, from the node v_1, we can reach nodes v_2, v_3, and v_4. Therefore, we have a corresponding entry of 1 in the matrix in the first row and the second, third and forth columns. In general, the number of 1's in the ith row, corresponds to the number of edges leaving node v_i, and the number of 1's in the jth column, corresponds to the number of edges entering the node v_j.

A path through a graph is a traversal of consecutive nodes along a sequence of edges. By this definition, the node at the end of one edge in the sequence must also be the node at the beginning of the next edge in the sequence. The nodes that begin and end the path are termed the initial node and terminal node, respectively. With the exception of these initial and terminal nodes, each node within the path has two neighboring nodes that must also be adjacent to the node. The length of the path is the number of edges that are traversed along the path.

Connecting a pair of nodes forms an edge in an undirected graph but no direction is stipulated for the edge. Therefore, for a path in an undirected graph, either node may be considered as the initial or terminal node of the path and the traversal of the node along the path can occur in either direction. However, a cycle in a simple undirected graph is slightly different from the definition of a cycle for a directed graph due to the lack of direction on the edges. For undirected graphs, the traversal of a set of nodes forward and then backward can not be considered a cycle. Rather, a simple cycle for an undirected graph must contain at least three different edges and no repeated nodes, with the exception of the initial and terminal nodes.

If a node is reachable from another node then a path exists from the one node to the other node. It is assumed that every node is reachable from itself. The definition of reachability holds true for both directed and undirected graphs.

An undirected graph is considered to be connected if a path exists between all pairs of nodes, thus making each of the nodes in a pair reachable from the other. An unconnected graph may be subdivided into what are termed connected subgraphs or connected components of the graph.

The connectivity of a simple directed graph becomes more complex because direction must be considered. Because of the added complexity, there are three distinct forms of connectivity in simple directed graphs: weakly connected, unilaterally connected and strongly connected. A weakly connected graph is where the direction of the graph is ignored and the connectivity is defined as if the graph was undirected. A unilaterally connected graph is defined as a graph for which at least one node of any pair of nodes is reachable from the other. A strongly connected graph is one in which for all pairs of nodes, both nodes are reachable from the other.

1.3 Reliability of Networks

Classically, one form of reliability is defined as the probability that there exist at least one path between two specific nodes. Transmission media need to be protected. The cables that carry thousands of twisted pairs of copper wire in the trunks are subject to fail. Today's telecommunication networks are switched and controlled by computers. These computers are a prime source of network failure.

The ability of the network to accommodate itself with the occurrence of link failures, is a measure of its reliability. These measures fall into two classes:

Deterministic measures which depend only on the structure of the network, that is, on the number of nodes and links and the way they are connected. Probabilistic measures of availability, on the other hand, depend not only on the structure but also on the probabilities of failure of nodes and links (Cravis, 1981). In a network, as links or nodes may fail at random, the network reliability is influenced, in addition to the network structure, by the probabilities of failure of its components.

When the network size is relatively small, there are exact methods to compute the probability that there exist at least one path between the origin node and the destination node. These methods are impractical when the network size becomes large. In this case approximation methods including simulation are usually used.

Network availability refers to some measure of the reliability of a network. We can define availability, as the ratio of the total time of a unit is capable of being used during a given interval and specified in decimal fractions, such a 0.9998.

Communication network reliability depends on both hardware and software. A variety of network failures, lasting from a few seconds to days depending on the failure, is possible. Traditionally, such failures were primarily from hardware malfunctions of a network element (a node or a link).

In (Li, 1984) and (Sanso, 1991), the measure of reliability for networks are based on lost call traffic calculations. These calculations use the Erlang loss formula and this is also our reliability measure.

CHAPTER II

Traffic Theory

2.1 Introduction

A type of communications in which a circuit is established for the duration of a transmission is called circuit switched. The most famous circuit-switching network is the telephone system, which links together wire segments to create a single unbroken line for each telephone call.

By traffic theory, we mean the application of mathematical modeling to explain the traffic-performance relation linking network capacity, traffic demand and realized performance. Since demand is statistical in nature, performance must be expressed in terms of probabilities and the appropriate modeling tools derived from the theory of stochastic process (Bear, 1980).

Traffic theory is fundamental to the design of the telephone network. The traffic-performance relation here is typified by the Erlang loss formula which gives the probability of call blocking, B, when a certain volume of traffic, a, is offered to a given number of circuits, c.

We will assume that the switching equipment can select any idle circuit in the group to serve any call request. This is called the assumption of full availability of a trunk group.

Assuming that calls are made at random, the appropriate statistical distribution for the input traffic commonly used is Poisson. In other words the probability Pj (t) that exactly j call requests will occur in an interval of t seconds is given by.

$$p_j(t) = \frac{(\lambda t)^j \exp(-\lambda t)}{j!} \qquad \textbf{(2-1)}$$

Where λ, a positive constant, is the average rate of call arrivals per second. Equation (2-1) is known as Poisson probability, which define the probability of having exactly j arrival calls in t seconds

The intervals between successive call requests have a negative exponential distribution with mean 1/ λ. The probability h (t) that a time inteval is less than or equal to t_1 seconds is

$$\mathbf{h(t) = 1\text{-}exp(\text{-}\lambda t_1),} \qquad \mathbf{t_1 \geq 0.} \qquad \textbf{(2-2)}$$

where **λ** is the average rate arrivals per seconds.

Call arrivals follow a Poisson distribution and holding times follow a negative exponential distribution. Implied in the derivation of Eq.(2-1) is that the number of subscribers is infinite, a condition that is approximated where the number of subscribers is much larger than the number of circuits. We now introduce the relevant formulas developed by Erlang to address the traffic problem (Cravis 1981).

2.2 Erlang B Formula

In the 1920's, a Danish mathematician and telephone traffic engineer named A. K. Erlang studied delays in telephone traffic and came up with several models which are still in use today (Qiao, 1998). These models deal with the following terms:

Arrival Rate - λ

The mean number of call arrivals per unit time is denoted as arrival rate. If five hundred calls arrive in an hour, on average, then the Arrival Rate is 500 calls/hour. The reciprocal ($1/\lambda$) is then the average amount of time that separates incoming calls.

Service Rate - μ

The service rate is defined as the mean number of calls serviced per unit time. If it takes 20 minutes to service one call, then 3 calls may be serviced in one hour and the Service Rate is 3 calls/hour. The reciprocal ($1/\mu$) is the average time taken to service one call (holding time), given in units of time per call (i.e. 0.333 hours/call or 20 minutes/call in the current example)

Number of Lines - c

Given any link, it has capacity c, where c denotes the number of independent phone lines it provides. This determines the number of calls that can be serviced concurrently.

Busy Hour

This is the period of one-hour duration taken to be the busiest period of the day, and during which incoming calls are most likely to be blocked or turned away. As such it becomes the period for which statistics are calculated.

Offered traffic - a

Another common unit is the Erlang, representing the ratio of Arrival Rate to Service Rate (λ / μ). For example if we expect 500 calls/hour, the Arrival Rate λ = 500 calls/hour. If the average duration of a call is 20 minutes, or 0.333 hours, then the Service Rate $\mu = 3$ calls/hour. Offered traffic a, is then calculated as $(\lambda / \mu) = 500/3 = 166.67$ in units of Erlangs.

Grade of Service - B

Grade of service is the percentage of incoming calls blocked during the Busy Hour because all lines are busy at the time of the call. Notice that this statistic is a function of the number of lines, with more lines resulting in a lower and therefore better Grade of Service B (c,a). The Expression for B (c,a) donated as Erlang B Formula and representing the blocking probability, is given by:

$$B(c,a) = \frac{\dfrac{a^c}{c!}}{\sum_{i=0}^{c} \dfrac{a^i}{i!}} \qquad \text{(2-3)}$$

Where a= λ / μ.

When a and c are relatively high (greater than 200) the evaluation of Eq.(2-3) goes to infinity and results in overflow errors. To circumvent this problem, the Erlang B formula can be reformulated as (Qiao, 1998):

$$B(c,a) = \frac{1}{\sum_{i=0}^{c} \left(\frac{c}{a}\right)\left(\frac{c-1}{a}\right)\ldots\ldots\ldots\ \ldots\ldots\ldots\left(\frac{c-i}{a}\right)} \qquad (2\text{-}4)$$

The carried traffic becomes:

$$a' = a[1 - B(c,a)] \qquad (2\text{-}5)$$

The blocked traffic or Lost Call Traffic (LCT) is the difference between offered traffic and carried traffic:

$$\mathbf{LCT} = \mathbf{a.B(c,a)} \qquad (2\text{-}6)$$

Example

Consider the three-node circuit-switched network, Fig. (2-1), for

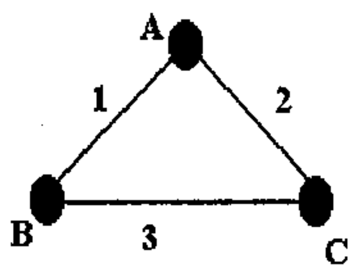


Fig. (2-1). Three Nodes Network. k.

which we are given that the availability of each link is 99.9%.

Assume that the network has symmetric offered traffic (or load) and equal link capacities. Assume that the offered symmetric load between any pair of nodes is 15 Erlangs, and that the link capacity of each link is given to be 31 trunks (or circuits). We assume that the traffic between each pair of nodes is routed on the direct link that connects the end nodes of the pair, and we would like to know the call-blocking probability. For an offered load of a Erlangs, and a link of capacity c trunks, and under the assumption that call arrival follows a Poisson process, Erlang-B loss formula can be used for computing the blocking probability. Thus, in our example, we have B(31,15)= 0.0001, that is, the network is providing a service quality (grade-of-service) of 0.01% blocking.. The lost call traffic is 0.0015 Erlangs.

Now, suppose that link 3 fails. In this case, the network is still connected because node C is connected to node B via node A. Assuming that the network still has the same amount of offered load, the load between node C and node B is now required to be routed through node A. Thus, the load offered to each link is 30 Erlangs, whereas the capacity on each link is still 31 trunks. Thus, the blocking seen by traffic on each link is B(31; 30) = 0.1136,and the blocking seen by pair B–C traffic going through node A is even higher. Under the link independence assumption, in our example, the lost call traffic in links 1 and 2 is 3.4087. Thus, we can see that, under no failure, the network provides a grade-of-service of 0.01%, whereas under a single link failure the worst traffic pair blocking is 11.36% although the network connectivity is still maintained. Recall that the link availability was assumed to be 99.9%; this means that the link can possibly be down for as long as 8 hours in a year. If we assume one event per link per year, then this link could conceivably be down for up to 8 hours straight! In some networks, this may be unacceptable given that the worst traffic pair blocking jumps to 11.36% from 0.01%. If we assume that the network should still provide a 0.01% blocking grade even under a single failure for every traffic pair, then to accommodate for the worst path blocking, we need link blocking on each of the remaining links to be such that the path blocking for traffic between node C and node B using links 1 and 2 to be 0.01%. We need to find the smallest c such that B(c, 30) = 0.0001. Solving for c, we find that c need to be at least 53 (i.e., we need to have 53 units of capacity on links 1 and 2 each). By the same argument, if we consider the failure of a different link independently, then the other two links each need 53 trunks. Thus, to cover for failure of each link independently, each link needs 53 trunks to provide the same level of blocking as was originally wanted for the network in the no failure mode. In other words, the network needs 76.6% more capacity to cover for a link failure compared to the no failure case although network availability requirement was met.

It is imperative to take into account when measuring the reliability of the network using lost call traffic the follwoing:

- Location of failure.
- Capacity of the failed link.
- Routing.

In a circuit-switched network that has N nodes and L links, we assumed that links only are able to fail, which is not necessarily true because nodes (switching centers) may also fail. We made this assumption for simplicity since a failed node implies the failure of all links incident at it. Another assumption commonly used is that links fail independent of each other. With these assumptions, solutions to real sized networks are possible.

Sanso (1991) presented a procedure to find the lost call traffic due to failure for a circuit switched telecommunication network. This procedure is summarized as follows:

1) Find the lost call traffic in the network when no failures are present.
2) Find the expected lost call traffic in the presence of failure in the network.
3) The difference between 1) and 2) will be the expected lost call traffic due only to failure.

In (Li, 1984), a method is presented to enumerate the states that need to be considered for a network of L links such that its performance remains valid by taking states less than 2^L . Li (1984) define lower and upper bounds on the expected lost call traffic which can be expressed as follows:

$$E(Z)_{up} = \sum_{k=1}^{t} P(S_k)Z(S_k) - \left(1 - \sum_{k=1}^{t} P(S_k)\right) Z(S_{2^m}) \qquad \textbf{(2-7a)}$$

$$E(Z)_{low} = \sum_{k=1}^{t} P(S_k)Z(S_k) - \left(1 - \sum_{k=1}^{t} P(S_k)\right) Z(S_1) \qquad \textbf{(2-7b)}$$

where

$E(Z)_{up}$ the expected lost call traffic upper bound,

$E(Z)_{low}$ the expected lost call traffic lower bound.

$P(S_k)$ the probability that the system is in state S_k,

$Z(S_1)$ the lost call traffic when no failures in the system,

$Z(S_2{}^m)$ the lost call traffic when all links are failed, and

t is the t most probable states.

The probability of any network state S_k to occur can be calculated:

$$\mathbf{P(S_k) = \prod_{i=1}^{L} P_i(q_i / p_i)^{T_i(S_k)}} \qquad \textbf{(2-8)}$$

Where

$$\mathbf{T_i(S_k) = \begin{cases} 0 \text{.......................if link } i \text{ is operatinin } S_k \\ 1 \text{...otherwi} \end{cases}}$$

$$\mathbf{q_i = 1 - p_i}$$

p_i is the availability of link 'i' and q_i is the probability of failure for link 'i'. and

From equation (2-7) we can evaluate the probability of the no failure state indicated by the state with the maximum probability value. The second most probable state is when a single failure is to occur, followed by states with two failures and so on..

Table (2-1) State Occurrence Probability for Fig (2-1).

State *No*	Link *1*	*Link 2*	*Link 3*	Probability
1	1	1	1	0.912576
2	1	1	0	0.018624
3	1	0	1	0.028224
4	0	1	1	0.038024
5	1	0	0	0.000576
6	0	1	0	0.000776
7	0	0	1	0.001176
8	0	0	0	0.000024

For Fig. (2-1), if we assume the availability of links 1,2 and 3 are 0.96, 0.97 and 0.98 respectively, the state occurrence probabilities are shown in Table (2-1).

Considering the network of Fig.(2-1), then according to Sanso, for networks with high link availabilities, the most probable states are state 1 for no failure, and state 2,3 and 4 for single failure, and which Table (2-1) shows very clearly. For networks with lesser link availabilities, Ayoub (1997) and Alsadi (1997) assume state 1 for no failure, states 2,3 and 4 for single failure, and states 5,6 and 7 for two failures, as the only states to be considered to calculate Lost Call Traffic. We will use the last approach when extending the network to have two levels of switching.

Table (2-2) was obtained to give the number of link failures to be considered at a time to arrive at a number of states whose sum of occurrence probabilities is 0.95 or more for different network size and link availability. These results can be used to determine the number simultaneous of failures that need to be considered at a time.

Table (2-2). Number of failures for a given network size and link availability to achieve 0.95 of the network states.

No. Of Nodes ↓ **Link Availability** →

	0.92	0.93	0.95	0.97	0.98	0.99	0.995	0.999
6	2	2	1	1	1	1	0	0
10	2	2	2	1	1	1	1	0
14	3	3	2	2	1	1	1	0
18	4	3	3	2	1	1	1	0
22	4	4	3	2	2	1	1	0
26	5	4	3	2	2	1	1	0
30	5	5	4	3	2	1	1	0
34	6	5	4	3	2	1	1	0
38	6	5	4	3	2	2	1	0
42	6	6	5	3	3	2	1	0
46	7	6	5	3	3	2	1	0
50	7	7	5	4	3	2	1	0

From Table (2-2) if N=6 then L=15(if the network is fully connected) and its states=2^{15} =32768 states. If the links availability is 0.93,the states with two links failure, and one link failure, and no link failure, need to be considered. No link failure produces one state, and one link failure produces 15 states, and two link failures produces 105 states. Togther, we have only 121 states that are important from 32768 states. In each of the 121 states, the lost call traffic is calculated.

CHAPTER III

Lost Call Traffic Evaluation

"Mr. Watson, come here, I want you." With these historic words Alexander Graham Bell called to his assistant Thomas Augustus Watson over the "telephone," and an industry was born[Bear, 1980].

3.1 Network

If there were only three or four telephones in a locale, it would make sense to connect each phone to all other phones and find a simple method of selecting the desired one. However, if there are three or four thousand phones in a locale, such a method is out of the question. Then it is appropriate to connect each phone to some centrally located office and perform switching there. This switching could be a simple manual operation using plugs and sockets or could be done with electromechanical devices. In any case, this "central office" solution is the one that has been chosen by the telecommunications industry.

As we connect each of these thousands of telephones to a one central office, we have what is called a star configuration. All lines are particular to one and only one station, and all terminate this star at the Central Office (CO). The connections themselves are often called the "local loop;" at other times we refer to them as "the last mile."

But what if a particular telephone call is not originated and terminated within the particular central office's geographic coverage? How do we get to another city or another country? The answer, of course, is to connect these central offices to a higher central office as shown in Fig. (3-1). In North America, numbers are given to these levels of offices in addition to standard names. The local office, also called the end office, is called a Class 5 office. The high level office to which it connects, is called Class 4 office, and so on, with the top level, donated as Class 1 office. It is obvious that the only office that has people as its subscribers is Class 5

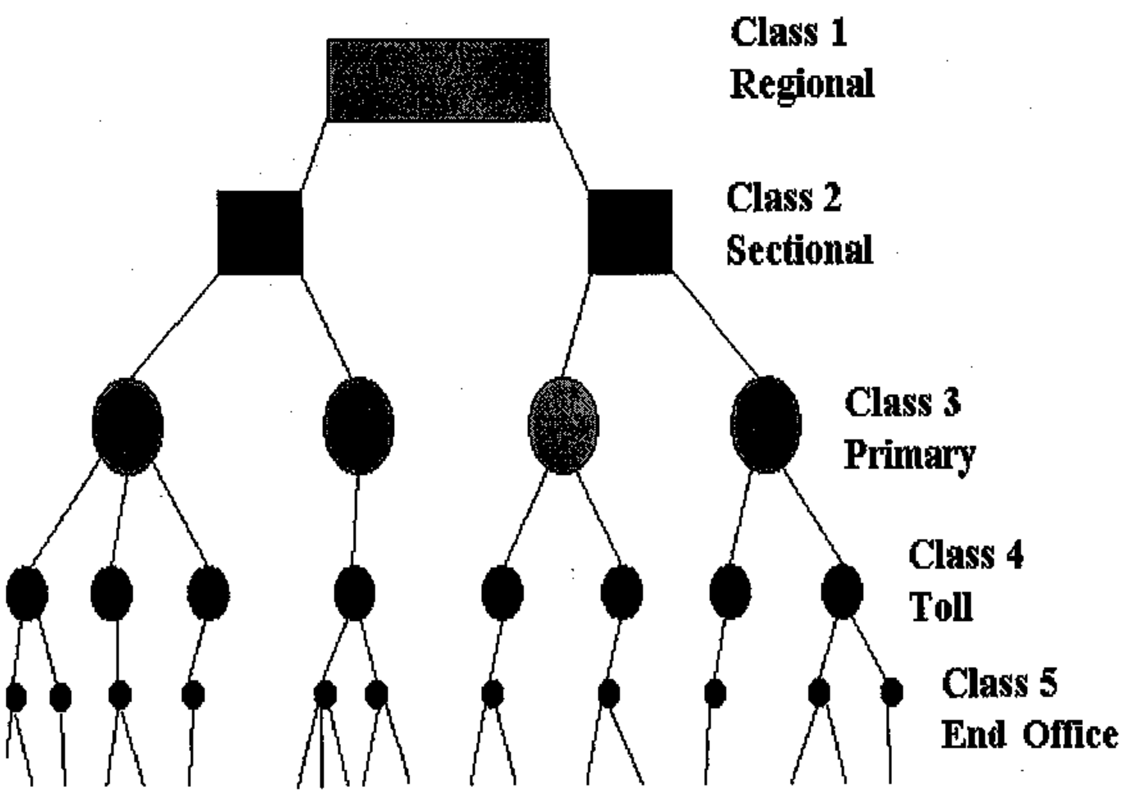


Fig. (3-1). The Hierarchy of Switching System.
(North American Practice)

office. The other offices in this hierarchy have lower level central offices as their "subscribers." Those lines connecting switching offices to switching offices, rather than to subscribers, are called trunks.

Given the fact that there are so many phones in the world, it is necessary to have a hierarchy of switching centers, in which phones are connected from a low level center to higher-level centers, and so on.

As an example, if we look at how the hierarchy of Fig. (3-1) is applied in North America, we find that there are five levels in this hierarchy. At the lowest level are tens of thousands of "Class 5" switching centers or "end offices", which are directly connected to phones over local loops. The end offices feed a thousand or so Class 4 "toll centers"; which in turn feed a few hundred Class 3 "primary centers"; which are themselves connected to Class 2 "sectional centers"; with a handful of Class 1 "regional centers" at the top. The arrangement is not strictly hierarchical: a Class 5 switching center may make a local connection without involving a higher-class switching center, and a Class 5 switching center may make a connection directly to a high-class switching center without going through any intermediate levels.

This entire structure has been titled the "hierarchy of switching systems." The total network is called the public switched telephone network (PSTN). An equivalent switching hierarchy which is a CCITT standard, is also present with a slightly different names given to the respective office (Bear, 1980), usually called exchanges.

It should be recognized that the interconnections between these various COs could be through using twisted copper pairs, microwave transmission system, satellite systems, or optical fiber.

3.2 Routing

A route between two nodes is the communication path or the link between them. For the same nodes there are more than one route and the number of routes grow exponentially by increasing the number of nodes.

The routes are categorized into two groups:

1-Direct route, which is the direct path route (shortest route) used as the first choice to carry traffic.

2-Alternative routes, which may be used in case of link failure or overflow. Choosing the alternative route might be through a process which could be:

a-Static: where the CO gets the route from a lookup table.

b-Dynamic: where the CO decides which route to use according to the network situation at that moment subject to certain constrains.

Each type of routing has advantages and disadvantages. The chief advantages of static routing are simplicity. The chief disadvantage is inflexibility, as static routes can't be changed easily. In dynamic routing the network is allowed to handle problems automatically. Most large networks use a form of dynamic routing.

In alternative routing, and as a matter of normal practice, the routing choice was limited to taking only two-link path in the event of a failure. In a large network, usually multiple routes between each origin and destination nodes are available. In the event of a failure, traffic can be sent to any of the unaffected paths. However, the actual flow on each path would depend on the actual routing rule in place as well as the availability of the network links and their capacities (Medhi, 1999). Thus, it is not hard to see that the lost call traffic in the network depends also on the actual routing schemes available in the event of a failure.

In our present study we will deal with a network that has two levels of switching, involving the level of end offices which handle the local traffic and the other level of toll centers which handle the national traffic. Then we examine the reliability of this two levels network by calculating the lost call traffic under condition of random failure for different routing approaches.

We will assume that there is no alternative routing in the lower level of the network at the event of failure, so if a link connecting two end offices fails, the offered traffic in that link will be lost. If a failure occurs in the higher level the offered traffic in that link will be routed over another unaffected two links. This assumption makes our network close to real networks. For example, subscribers in Europe and the USA pay each month the same amount of subscription fees to the communication companies for local communication whether they use the line for one minute or ten hours, and whether the links work or not. National communications however are clock timed, which start to count when they attempt to make national calls, so it is important to route these calls when link failures occur.

In our present work, network reliability performance will be tested, when a random failure occurs in the higher switching level, under the following routing approaches:

1) Traffic will be carried randomly through any two-link path between the origin and destination nodes.
2) Traffic will be carried through the two-link path that has the maximum capacity available with respect to their traffic.
3) Traffic will be carried through two-link path that has the maximum availability and have enough capacity.

3.3 Lost call Traffic Evaluation

In Chapter II, we illustrated how Li (1984) determine the most probable network states that need to be considered for simulation. Also we summarized a procedure presented by Sanso (1991) to find the lost call traffic due to failure. With the assumption that links only are able to fail, and link failures are independent of each other, we present a procedure to find the lost call traffic in a network that has two levels of switching. The following steps are made:

1) Find the lost call traffic for the no failure state for the lower networks (local networks) and the higher network (national network).

2) Find the expected lost call traffic in the presence of single failure states, by using the routing strategies that we have mentioned for the higher network. This is done when the links between the two levels are not subject to fail and when they are subject to fail.

3) Find the expected lost call traffic in the presence of two failure states for all routing approaches mentioned in step 2.

4) The average lost call traffic is obtained by weighting the results in step 1,2 and 3 by the probability of state occurrences and summing the results.

The statistical averaging stipulated in step 4 is justified by noting that the expected lost call traffic for states which allow several links to fail at the same time, will be substantial. Yet its contribution to the overall average lost call traffic will be much reduced subject to the weighting effect of the occurrence probabilities of these states which are very small for networks of high link availabilities.

As an example, let us apply the procedure to the network of Fig (3-2).

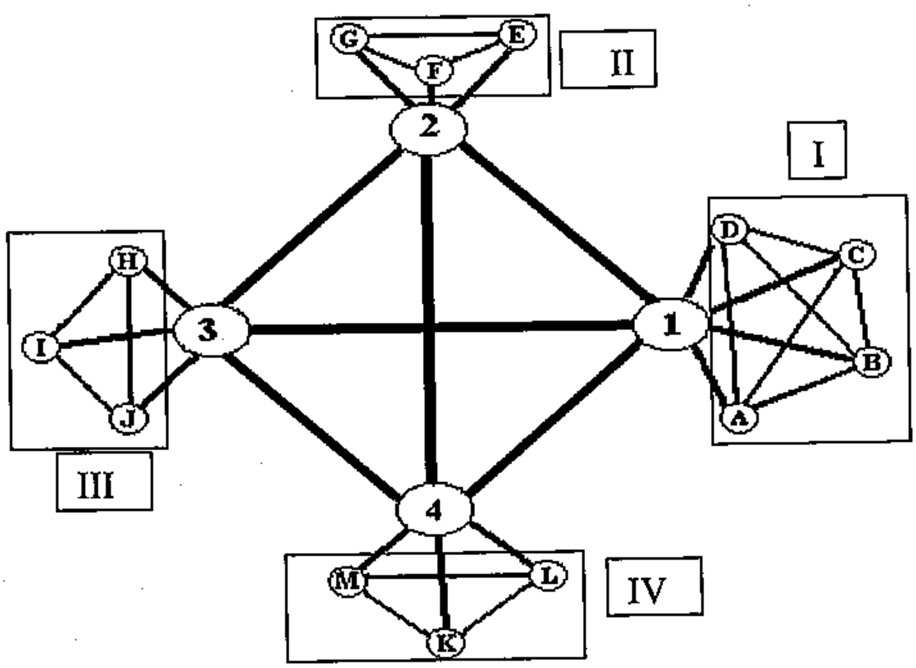


Fig. (3-2). Two Levels Network.

Assume the offered traffic a, and links capacity c, for this network as given in the following matrices:

$$a = \begin{array}{c} \\ \\ \end{array}\begin{array}{c c} \begin{array}{ccccc} 1 & A & B & C & D \end{array} & \\ \begin{bmatrix} 0 & 12 & 7 & 13 & 10 \\ 12 & 0 & 6 & 8 & 11 \\ 7 & 6 & 0 & 9 & 10 \\ 13 & 8 & 9 & 0 & 12 \\ 10 & 11 & 10 & 12 & 0 \end{bmatrix} & \begin{array}{c} 1 \\ A \\ B \\ C \\ D \end{array} \end{array} \qquad c = \begin{array}{c c} \begin{array}{ccccc} 1 & A & B & C & D \end{array} & \\ \begin{bmatrix} 0 & 20 & 20 & 20 & 20 \\ 20 & 0 & 15 & 15 & 15 \\ 20 & 15 & 0 & 15 & 15 \\ 20 & 15 & 15 & 0 & 15 \\ 20 & 15 & 15 & 15 & 0 \end{bmatrix} & \begin{array}{c} 1 \\ A \\ B \\ C \\ D \end{array} \end{array}$$

$$a = \begin{array}{cccc|c} 3 & H & I & J & \\ \end{array}$$

$$a = \begin{array}{c} \begin{array}{cccc} 3 & H & I & J \end{array} \\ \begin{bmatrix} 0 & 8 & 8 & 9 \\ 8 & 0 & 6 & 6 \\ 8 & 6 & 0 & 10 \\ 9 & 6 & 10 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 3 \\ H \\ I \\ J \end{array} \qquad c = \begin{array}{c} \begin{array}{cccc} 3 & H & I & J \end{array} \\ \begin{bmatrix} 0 & 18 & 18 & 18 \\ 18 & 0 & 14 & 14 \\ 18 & 14 & 0 & 14 \\ 18 & 14 & 14 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 3 \\ H \\ I \\ J \end{array}$$

$$a = \begin{array}{c} \begin{array}{cccc} 4 & K & L & M \end{array} \\ \begin{bmatrix} 0 & 10 & 10 & 5 \\ 10 & 0 & 11 & 10 \\ 10 & 11 & 0 & 7 \\ 5 & 10 & 7 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 4 \\ K \\ L \\ M \end{array} \qquad c = \begin{array}{c} \begin{array}{cccc} 4 & K & L & M \end{array} \\ \begin{bmatrix} 0 & 18 & 18 & 18 \\ 18 & 0 & 14 & 14 \\ 18 & 14 & 0 & 14 \\ 18 & 14 & 14 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 4 \\ K \\ L \\ M \end{array}$$

$$a = \begin{array}{c} \begin{array}{cccc} 2 & G & F & E \end{array} \\ \begin{bmatrix} 0 & 6 & 5 & 5 \\ 6 & 0 & 8 & 6 \\ 5 & 8 & 0 & 12 \\ 5 & 6 & 12 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 2 \\ G \\ F \\ E \end{array} \qquad c = \begin{array}{c} \begin{array}{cccc} 2 & G & F & E \end{array} \\ \begin{bmatrix} 0 & 18 & 18 & 18 \\ 18 & 0 & 14 & 14 \\ 18 & 14 & 0 & 14 \\ 18 & 14 & 14 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 2 \\ G \\ F \\ E \end{array}$$

a_{ij} and c_{ij} are the offered traffic and the link capacity for the link between node i and j.

In every matrix, the entries of the first row and first column specify traffic offered and link capacities for the higher level network. The remaining entries belong to the lower level network. The lost call traffic in Erlangs can be calculated by using equation (2-6). When there is no failure the LCT for each network.

$$LCT = \begin{array}{ccccc} I & II & III & IV & Network \\ [2.2975 & 1.5577 & 0.5950 & 1.5552] & LCT \end{array}$$

The average lost call traffic in Erlangs when there is single failure:

$$LCT = \begin{array}{cccc} I & II & III & IV \\ \left[\boldsymbol{11.2479} \right. & \boldsymbol{9.7051} & \boldsymbol{7.7300} & \left. \boldsymbol{10.3701} \right] \end{array}$$

The average lost call traffic in Erlangs when there are two failures:

$$LCT = \begin{array}{cccc} I & II & III & IV \\ \left[\boldsymbol{20.1983} \right. & \boldsymbol{17.8526} & \boldsymbol{14.8650} & \left. \boldsymbol{19.1851} \right] \end{array}$$

The traffic offered to the higher network is equal to the summation of the traffic that is generated form the lower networks and direct to the higher network. The traffic in Erlangs offered from the lower to the higher level per network is obtained by summing the values of the top row in every network to give for each network the following :

$$\begin{array}{cccc} I & II & III & IV \\ \left[\boldsymbol{42} \right. & \boldsymbol{16} & \boldsymbol{25} & \left. \boldsymbol{25} \right] \end{array}$$

The over all traffic in the higher network equal to:

$$\sum \frac{\textbf{Trafficgeneratedfromlowernodestohighernodes}}{\textbf{2}}$$

$$\frac{\mathbf{42 + 16 + 25 + 25}}{\mathbf{2}} = \mathbf{54} \qquad \textbf{Erlangs}$$

We divide the summation by two because of symmetry. Hence, The traffic from node 1 to node 2 equals to that from node 2 to node 1.

Since We don't know the route of the traffic that comes from the lower networks to the higher network, and that the offered traffic for the links of the higher network is required, let us assume that the traffic in these links is proportional to the number of nodes in the two networks connecting this link (lower networks). The offered traffic in the link between node 1 and node 2 equals:

$$(42).\frac{4+3}{4+3+4+3+4+3}=14 \text{ Erlangs}$$

The denominator is the summation of all possible states of the number of nodes between network I which is connected to node 1 and other networks. When the offered traffic that goes form node 1 to other nodes is calculated, the traffic between nodes 2 which is connected to network II and remaining nodes in the higher network must be:

16-14=2 Erlangs

The offered traffic from node 2 to node3 equals:

$$(2).\ \frac{3+3}{3+3+3+3} = 1 \text{ Erlang}$$

Continuing this procedure, the traffic generated in the lower network is divided as offered traffic to the link in higher network as follows:

$$a = \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 14 & 14 & 14 \\ 14 & 0 & 1 & 1 \\ 14 & 1 & 0 & 10 \\ 14 & 1 & 10 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}$$

It should be noted that this offered traffic could have been distributed among the higher network links in any different way, and can still be used in the subsequent calculations. Assuming links capacities for the higher network as:

$$c = \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 30 & 30 & 30 \\ 30 & 0 & 30 & 30 \\ 30 & 30 & 0 & 30 \\ 30 & 30 & 30 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}$$

The lost call traffic for the higher network when there is no failure is equal to 0.5594 Erlangs. Assuming links availabilities, A for the lower networks is 0.96, and for the higher network as:

$$A = \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 0.95 & 0.98 & 0.97 \\ 0.95 & 0 & 0.95 & 0.96 \\ 0.98 & 0.95 & 0 & 0.95 \\ 0.97 & 096 & 0.95 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}$$

We will consider two cases for the states of the links between higher and lower networks:

Case 1:

Links are subject to fail with link availability<0.999.

If a link fails, the traffic in that link will be lost. Let us take one of the states that will happen if there is a single failure when the link between node 2 and node 3 is failed. All possible alternative routes between node 2 and node 3 are over node 1 and node 4. We will test three routing approaches:

(1-A): Traffic between node i and j will be carried over the two-link path that has the maximum capacity with respect to their traffic. The traffic will be carried over the node k determined according to the following expression:

$$\mathbf{Max}_k \{(c_{ik}-a_{ik})+(c_{kj}-a_{kj})\} \qquad k\neq i\neq j$$

The traffic will be routed over node 4 because the links of node 1 is more congested than that of node 4. The traffic between node 2 and node 3 is added to the traffic in the links that attaches node 4.The offered traffic in the link between node 2 and node 4 becomes:

1+1=2 Erlangs

The offered traffic in the link between node 4 and node 3 becomes:

10+1=11 Erlangs

Then we calculate the lost call traffic for the new offered traffic. The lost call traffic for the higher network when there is single failure is obtained by averaging the lost call traffic for all states which is equal 6.6473 Erlangs. The lost call traffic when there are two failures at a time using the same routing approach is equal 14.5095 Erlangs.

(1-B): The traffic will be carried over the two-link path that have the maximum availability and have enough capacity according:

$$\mathbf{Max_k\ \{(A_{ik}+A_{kj})/2\}} \qquad \mathbf{k \neq i \neq j}$$

There is enough link capacity for the links that attach to node 1 and node 4 to carry the traffic from node 2 to node 3.But the average link availability of the route over node 1:

$$\frac{\mathbf{0.95+0.98}}{\mathbf{2}} = \mathbf{0.965}$$

And over node 4:

$$\frac{\mathbf{0.96+0.95}}{\mathbf{2}} = \mathbf{0.955}$$

The traffic will be routed over node 1. The traffic of the failed link will be added to the links that attach to node 1 to carry the traffic from node 2 to node 3. The average lost call traffic when there is single failure and two failures equal 6.8497 Erlangs and 15.0261 Erlangs.

(1-C): The traffic will be carried randomly over any two-link path. There is no cost in this approach. Any unaffected rout to carry the traffic can be selected. The lost call traffic using this approach when there is single failure and two failures are 6.7485 Erlangs and 14.5853 Erlangs.

Case 2:

The links between higher and lower networks have high availability to the extent that we can assume they don't fail. As a result, the number of states will decrease and of course the lost call traffic will decrease. Using the same routing that have been mentioned in case 1:

(2-A): The traffic is carried over the highest capacity alternative route. The lost call traffic when there are single failure and two failures are 0.9934 Erlangs and 6.0455 Erlangs.

(2-B): The traffic is carried over a random unaffected links. The lost call traffic when there are single failure and two failures are 1.0949 Erlangs and 6.0456 Erlangs.

Now we have to weight the lost call traffic that has been calculated by the respective state occurrence probabilities as follows:

$$\mathbf{WLCT=BN.P_N+BL.P_L+BLL.P_{LL}} \qquad \mathbf{(3\text{-}1)}$$

Where:

WLCT: Expected Lost Call Traffic in the network.

BN: Expected Lost Call Traffic when there is no failure.

P_N: probability of state occurrence when there is no failure.

BL: Expected Lost Call Traffic when there is single failure.

P_L: Summation of state occurrence probabilities for states with single failure.

BLL: Expected Lost Call Traffic when there are two failures.

P_{LL}: Summation of state occurrence probabilities for states with two failures.

Table (3-1) shows the summation of state occurrence probabilities for the previous network when assuming the links between higher and lower networks have 0.96 availability.

NETWORK	Nodes	Links	P_N	P_L	P_{LL}	SUM	
1	4	6	0.7828	0.1957	0.0204	0.9989	
2	3	3	0.8847	0.1106	0.0046	0.9999	
3	3	3	0.8847	0.1106	0.0046	0.9999	
4	3	3	0.8847	0.1106	0.0046	0.9999	
HIGHER NETWORK	4	19	0.4602	0.3647	0.1386	0.9635	Case 1
	4	19	0.7824	0.1963	0.0207	0.9994	Case 2
	17	34					

Table (3-1) State Occurrence Probability for Fig (3-2).

We will call this method of evaluating lost call traffic as method 1.The weighting results are shown in table (3-2).

NETWORK	ROUTING	BN	BL	BLL	WLCT
1		2.30	11.25	20.19	4.40
2		1.56	9.71	17.85	2.53
3		0.60	7.73	14.87	1.46
4		1.55	10.37	19.19	2.78
HIGHER NETWORK	(1-A)	0.56	6.65	14.51	4.69
	(1-B)	0.56	6.85	15.03	4.84
	(1-C)	0.56	6.75	14.59	4.74
	(2-A)	0.56	0.99	6.05	0.75
	(2-B)	0.56	1.09	6.05	0.77

Table (3-2) Weighting for Fig (3-2) (method 1).

3.4 LCT Evaluation by State Simulation

Instead of calculating the lost call traffic for certain states and weighting the results, we can use a more general simulation method, which give us close results by generating network sates randomly. One possible method to generate a network state is by generating random numbers between 0 and 1 for all the network links. If the number is greater than the link availability, the link fails; otherwise, it is operational. Then calculate the lost call traffic for this state and repeating this procedure K times and averaging the results. We will call this method as method 2.

This approach is useful when the network size is large and when the network has low link availability thereby, allowing many links to fail simultaneously. By taking K large enough, a good estimation for lost call traffic is obtained. Table (3-3) shows such results using this method by taking K=1000. The results show good agreement with Table (3-2).

NODES		LCT
1		4.65
2		3.12
3		1.91
4		2.94
HIGHER NETWORK	(1-A)	5.02
	(1-B)	5.53
	(1-C)	5.25
	(2-A)	1.03
	(2-B)	1.10
	ROUTING	

Table (3-3) Lost Call Traffic for Fig (3-2) (method 2).

CHAPTER IV

Results and Conclusions

4.1 Introduction

In this chapter we will evaluate the reliability measure based on lost call traffic under condition of random failure and overload. A real network that has two levels of switching will be considered for this evaluation. As a first example let us find the lost call traffic for a network that has N=10 nodes representing a higher level network as shown in Fig. (4-1).

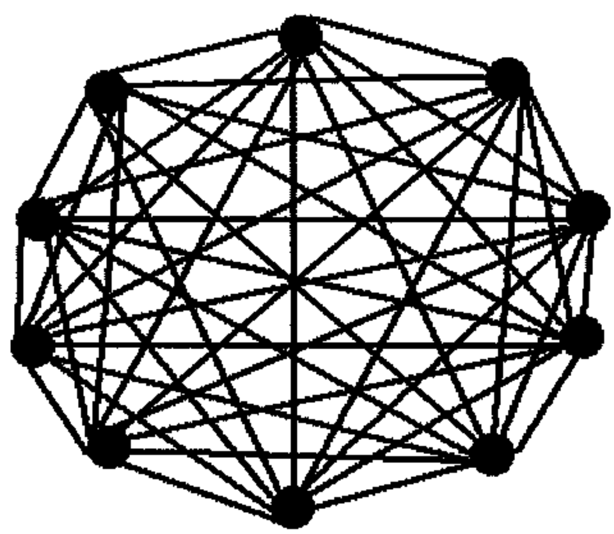


Fig (4-1). Higher Level Network with 10 nodes.

The number of links in this network is 45. So the connection matrix T will be a unity matrix with zero for diagonal elements.

The offered traffic to this network is generated from the traffic that comes from the lower level networks, so the summation of the traffic in the links that connect any higher node with lower nodes is equal to the summation of the traffic of links that connect the same higher node with other higher nodes. The over all traffic, H on the higher network (if there is no overload) can be calculated as:

$$H = \sum_{j=1}^{N} \frac{d_j}{2}$$

Where:

N: Number of nodes in the higher network.

d_j: Summation of the Traffic that comes from the links that connect the lower nodes to the higher node, j.

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

Assume that capacities of the links for the network is as shown in the following capacity matrix:

$$c = \begin{bmatrix} 0 & 45 & 59 & 47 & 58 & 64 & 74 & 54 & 36 & 64 \\ 45 & 0 & 25 & 85 & 45 & 69 & 58 & 78 & 41 & 35 \\ 59 & 25 & 0 & 26 & 58 & 49 & 87 & 68 & 23 & 14 \\ 47 & 85 & 26 & 0 & 59 & 48 & 78 & 69 & 85 & 74 \\ 58 & 45 & 58 & 59 & 0 & 63 & 52 & 48 & 75 & 48 \\ 64 & 69 & 49 & 48 & 63 & 0 & 68 & 47 & 59 & 47 \\ 74 & 58 & 87 & 78 & 52 & 68 & 0 & 78 & 88 & 64 \\ 54 & 78 & 68 & 69 & 48 & 47 & 78 & 0 & 66 & 45 \\ 36 & 41 & 23 & 85 & 75 & 59 & 88 & 66 & 0 & 55 \\ 64 & 35 & 14 & 74 & 48 & 47 & 64 & 45 & 55 & 0 \end{bmatrix}$$

We will assume the offered traffic to be a percent of the link capacity and allow no more than two links to fail at a time. When a failure occurs, the traffic of any failed link will be carried over a two-link path depending on the routing discipline selected. The following routing methods will be tested.:

1) Traffic will be carried over the two-link path that has the maximum capacity with respect to their traffic. The program subtracts the offered traffic from the link capacity for the links that can route the traffic for its destination, and the results for the two links are added together. The node k to be selected for the two-link route will be the one satisfied by the following expression:

$$\mathbf{Max_k \{(c_{ik}-a_{ik})+(c_{kj}-a_{kj})\}} \qquad \mathbf{k \neq i \neq j}$$

Where c_{ik} is the capacity of link (i,k) connected between nodes i and k and a_{ik} is the offered traffic for link (i,k).

2) The traffic will be carried over the two-link path that have the maximum availability and have enough capacity. Thus node k is selected such that the following is satisfied:

$$\mathbf{Max_k \{(A_{ik}+A_{kj})/2\}} \qquad \mathbf{k \neq i \neq j}$$

Where A_{ik} is availability of link (i,k).

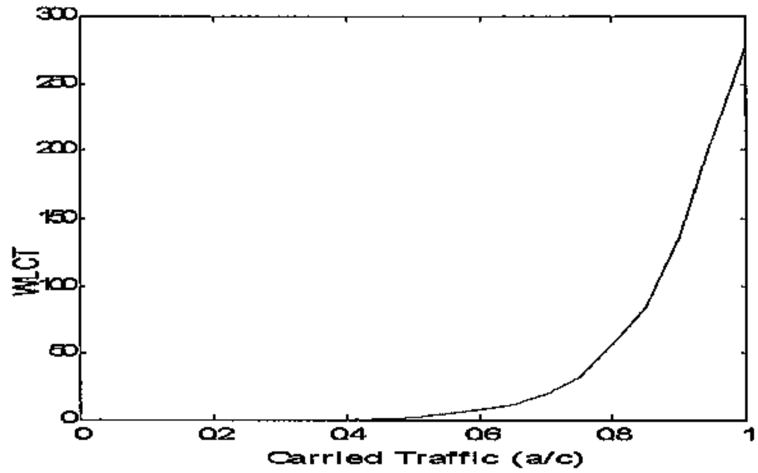
3) The traffic is carried over any unaffected two-link path selected randomly. We assume the link availability A for the network of Fig (4-1) as:

$$A = \begin{bmatrix} 0 & 0.99 & 0.98 & 0.97 & 0.95 & 0.96 & 0.97 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0 & 0.98 & 0.98 & 0.97 & 0.97 & 0.98 & 0.96 & 0.99 & 0.98 \\ 0.98 & 0.98 & 0 & 0.97 & 0.97 & 0.97 & 0.98 & 0.98 & 0.98 & 0.99 \\ 0.97 & 0.98 & 0.97 & 0 & 0.99 & 0.99 & 0.99 & 0.99 & 0.98 & 0.99 \\ 0.95 & 0.97 & 0.97 & 0.99 & 0 & 0.98 & 0.98 & 0.98 & 0.99 & 0.98 \\ 0.96 & 0.97 & 0.97 & 0.99 & 0.98 & 0 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.97 & 0.98 & 0.98 & 0.99 & 0.98 & 0.99 & 0 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.96 & 0.98 & 0.99 & 0.98 & 0.99 & 0.99 & 0 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.98 & 0.98 & 0.99 & 0.99 & 0.99 & 0.99 & 0 & 0.99 \\ 0.99 & 0.98 & 0.99 & 0.99 & 0.98 & 0.99 & 0.99 & 0.99 & 0.99 & 0 \end{bmatrix}$$

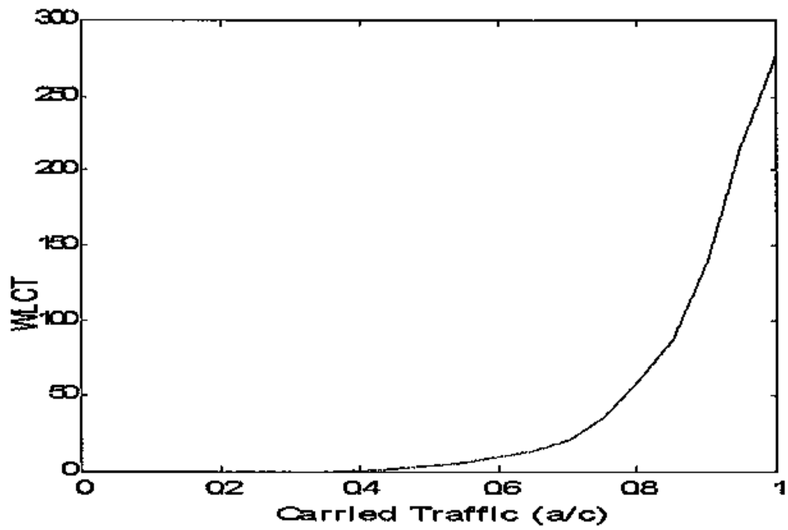
Using equation (2-6) to calculate lost call traffic for all network states that allow no failure, single failure, and two failures at a time, then weighting the results by the state occurrence probabilities, we obtain the results shown in Table (4-1) for the three routing methods.

Traffic % capacity	BN	BL			BLL			WLCT		
Routing		1	2	3	1	2	3	1	2	3
0	0	0	0	0	0	0	0	0	0	0
5	2.70E-14	2.30E-11	8.40E-09	5.20E-10	9.70E-05	3.10E-03	5.30E-04	1.55E-05	4.96E-04	8.48E-05
10	4.40E-10	5.70E-08	7.30E-08	9.00E-07	1.90E-03	1.40E-02	1.60E-02	3.04E-04	2.24E-03	2.56E-03
15	9.50E-08	3.70E-06	1.20E-04	7.30E-05	5.50E-03	3.60E-02	2.70E-02	8.81E-04	5.81E-03	4.35E-03
20	3.60E-06	5.90E-04	8.90E-04	6.60E-03	8.70E-03	5.70E-02	9.30E-02	1.62E-03	9.46E-03	1.74E-02
25	5.00E-05	8.4E-03	2.60E-02	5.50E-02	3.80E-02	0.2	0.21	9.29E-03	4.19E-02	5.45E-02
30	4.00E-04	2.30E-03	6.40E-02	0.10	9.00E-02	1.11	0.98	1.54E-02	0.20	0.19
35	2.00E-03	0.07	0.37	0.64	0.12	1.64	2.34	0.05	0.40	0.62
40	8.20E-03	0.28	0.94	1.38	0.74	5.94	6.37	0.23	1.31	1.55
45	2.70E-02	1.43	1.64	2.05	2.65	8.26	7.98	0.98	1.96	2.07
50	8.00E-02	2.89	3.84	3.64	6.7	13.35	14.94	2.20	3.63	3.81
55	0.22	6.50	7.49	8.22	13.23	21.65	22.97	4.67	6.40	6.89
60	0.63	10.68	12.36	15.24	22.07	32.61	31.01	7.84	10.17	11.01
65	1.81	16.25	18.26	19.07	33.15	36.21	40.53	12.20	13.46	14.46
70	5.07	26.05	28.11	27.34	47.27	50.69	55.87	19.49	20.82	21.36
75	13.17	43.14	45.64	49.15	66.65	78.69	74.37	32.33	35.20	35.84
80	30.39	75.69	79.21	76.36	94.99	99.36	101.26	56.12	58.15	57.37
85	61.25	102.54	104.90	109.84	136.2	138.26	139.99	85.26	86.48	88.64
90	108.48	166.36	170.54	175.95	192.59	194.26	195.24	137.42	139.28	141.49
95	172.05	233.25	248.26	240.95	310.85	325.47	330.58	207.19	215.23	213.27
100	249.93	309.84	309.84	309.84	375.35	375.35	375.35	277.77	277.77	277.77

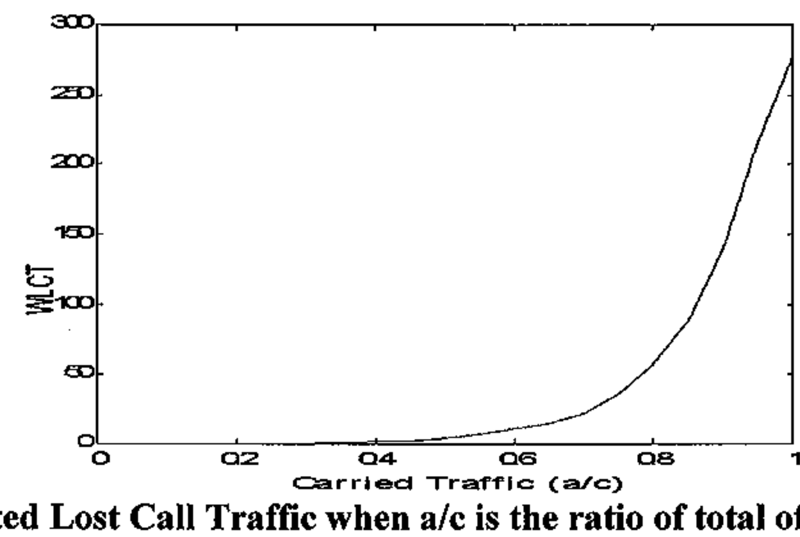
Table (4-1) Average BN, BL, BLL and WLCT for Fig. (4-1).



A- Weighted Lost Call Traffic when a/c is the ratio of total offered traffic using routing 1.



B- Weighted Lost Call Traffic when a/c is the ratio of total offered traffic using routing 2.



C-Weighted Lost Call Traffic when a/c is the ratio of total offered traffic using routing 3.

Fig. (4-2). Weighted Lost Call Traffic for Fig. (4-1)

Fig (4-2) represents the weighted lost call traffic in case of failure for the higher network. Normally, the network example considered would have lower level networks connected with it. The average lost call traffic for the overall network (higher and lower) can be calculated by adding the weighted lost call traffic in the lower networks to that of the higher network.

Our next network example is to study the Jordan telephone network as a practical case for a two-level network. We proceed to evaluate the lost call traffic following the same steps that have been mentioned in chapter III. The program is written by matlab programming language because of its ready build-in functions and speed.

Now we will give a description of the Jordan telephone network which would be our model to calculate lost call traffic under the condition of random failure.

4.2 Jordan Telephone Network

Jordan switching centers are categorized into four geographical regions:

1- 24 switches in Amman.

2- 8 switches in the north region.

3- 10 switches in the center region.

4- 9 switches in the south region.

There are 51 switching centers in the Jordan telephone network which are French made E10B or OCB238, Japanese made F-100 or F150 and German made EWSD or SDE.

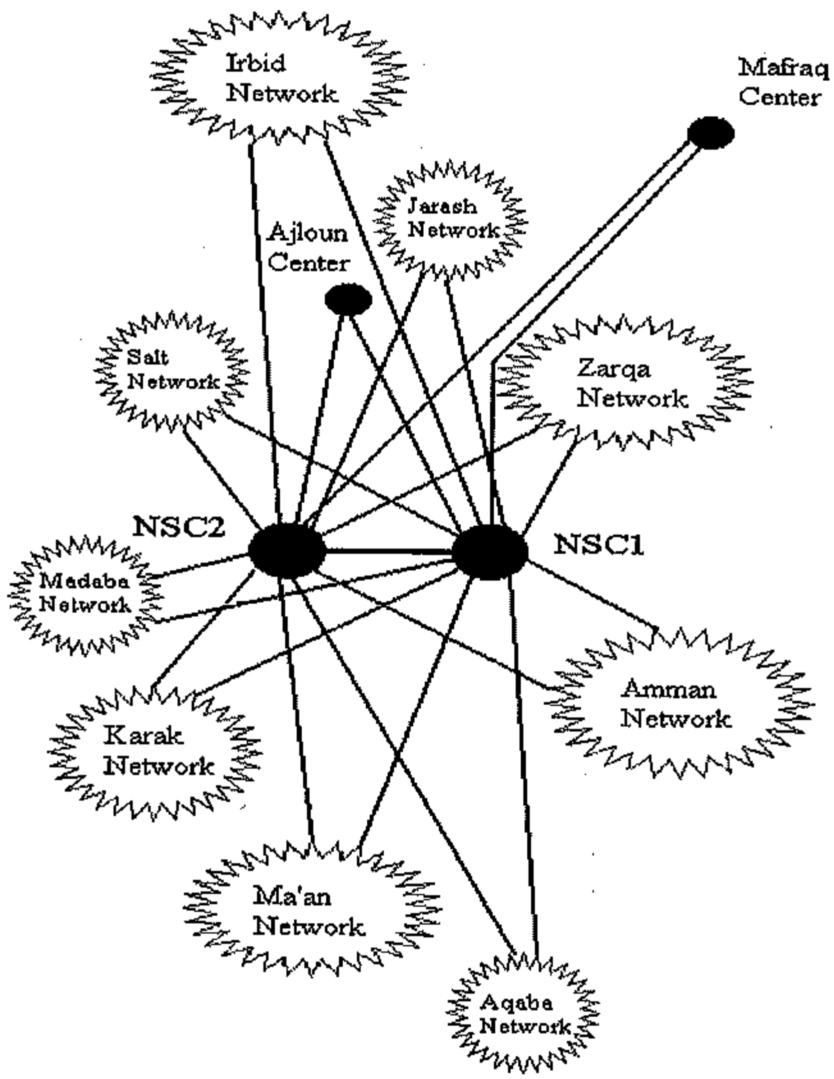


Fig. (4-3). Jordan Area Telephone Network.

Fig. (4-3) represents the Jordan telephone network. The clouds represent the lower networks and the large nodes NSC1 and NSC2 represent the higher switching centers. Every node in the lower networks is connected to the higher centers NSC1 and NSC2 (national centers) and to the centers in the same networks (local centers).

The links' capacities and the offered traffic in Erlangs were taken from the Jordan Telecommunication Company measurements at the busy hour on the first Sunday of July.2000.

1)**Amman Area Network:**

In Amman there are 22 centers which are almost connected with each other and with the national centers NSC1 and NSC2. The connection matrix for the Amman area network is given by Matrix T:

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

Table (4-2) and Table (4-3) show the direct offered load and links' capacities in Amman area network.

	ABD2	ABD3	ABDOUN	ALANDA	ASH2	ASH3	CEN2	CEN3	MRJ	MRK	NASSER	NAZAL	NSC1	NSC2	OMQASIR	QAIA	RASHEED	SAHAB	SWL	SWL2	TAREQ	TLA ALI	WDS1	WDS2
ABD2																								
ABD3	356.5																							
ABDOUN	46.7	69.5																						
ALANDA	20.4	18.1	4.25																					
ASH2	79.5	74.4	12.4	44.8																				
ASH3	94.8	74.4	10.7	3.4	297																			
CEN2	110.8	101.2	0	0	16.6	60.9																		
CEN3	234.9	241.2	51.3	20.7	142	93.1	315.8																	
MRJ	23.4	18	4.65	2.1	17.1	13.8	10.8	19.2																
MRK	58.5	39.3	0	0	70.8	86.2	50.9	90	6.3															
NASSER	49.4	39	8.57	19.6	56.2	55.1	49.7	63.7	5.5	107														
NAZAL	36.1	52.5	37.3	22.9	60.6	53.8	115.3	56.6	22.9	50.7	22.2													
NSC1	342	126	11.3	12.8	68.8	49.8	39.2	354	32.7	64.1	29.6	27.4												
NSC2	189	110.3	38.1	28.9	73.2	75.6	48	177.9	0	57.9	53.9	30.4	4.6											
OMQASIR	38.3	44.7	7.5	24	77.8	60.4	0	54	0	0	25.3	86.9	42.4	53.1										
QAIA	0	0	0	0	0	0	0	0	0	0	0	35	12.8	14	0									
RASHEED	55.4	84.9	9.4	5.7	22.9	16.6	0	49.6	7.8	0	21.3	43.3	26.5	32.1	8.9	0								
SAHAB	31.2	29.3	3.05	20.2	36.1	26.9	0	33.8	0	0	34	12.9	18.8	40.8	27	0	4.4							
SWL	91	73.8	0	11.1	42.7	50.4	26	51	8.4	14	13.2	12.3	35.5	79.2	0	0	0	0						
SWL2	45	42.6	5.5	2.6	25.7	28.3	0	49.8	0	0	10.6	7.8	29.3	31.2	9.8	0	13.3	5.4	76.1					
TAREQ	58.6	78.4	7.8	7.1	41.2	33.8	0	40.2	10	0	86.7	47.3	54.4	60.5	13.9	0	23.8	8	0	18.6				
TLA ALI	313.6	305.8	116.8	17.4	75.5	73.9	104.7	239.8	43.5	60.2	35.1	31.9	163	90.5	58.8	120	128.5	36.5	169	66	91.2			
WDS1	188.2	210.9	46.4	13.1	58	85.8	0	232.5	35.2	0	25.8	32.3	75	125	40.2	0	40.9	17.7	0	43.1	39.3	292.3		
WDS2	196.4	200.9	0	13.5	0	0	0	243	34.8	0	28.8	35.9	0	0	39.3	0	0	17.2	0	0	0	0	0	

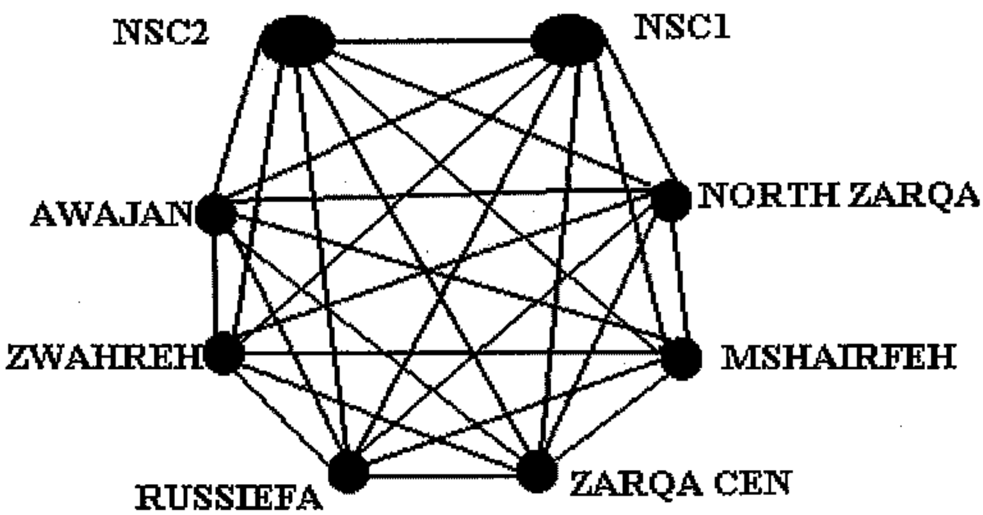
Table (4-2) Amman Area Telephone Network Offered Traffic in Erlangs.

	ABD2	ABD3	ABDOUN	ALANDA	ASH2	ASH3	CEN2	CEN3	MRJ	MRK	NASSER	NAZAL	NSC1	NSC2	OMQASIR	QAIA	RASHEED	SAHAB	SWL	SWL2	TAREQ	TLA ALI	WDS1	WDS2
ABD2																								
ABD3	420																							
ABDOUN	120	155																						
ALANDA	60	93	62																					
ASH2	240	124	62	93																				
ASH3	240	124	62	93	525																			
CEN2	264	150	0	0	30	90																		
CEN3	540	360	90	60	330	270	469																	
MRJ	120	120	30	30	90	60	60	120																
MRK	180	60	0	0	120	150	162	270	60															
NASSER	90	155	124	62	186	217	76	120	30	120														
NAZAL	120	155	124	62	217	217	120	150	60	90	124													
NSC1	450	154	61	61	148	91	180	420	119	202	92	61												
NSC2	240	154	60	61	149	123	120	210	0	117	123	60	60											
OMQASIR	60	93	62	93	93	124	0	90	0	0	62	124	61	92										
QAIA	0	0	0	0	0	0	0	0	0	0	0	31	40	80	0									
RASHEED	120	153	124	62	62	93	0	120	30	0	124	155	61	61	93	0								
SAHAB	60	93	124	62	62	93	0	60	0	0	93	124	61	61	93	0	31							
SW1	266	120	0	60	118	90	146	120	60	96	60	60	89	120	0	0	0	0						
SWL2	90	124	31	31	62	62	0	120	0	91	31	93	60	61	31	0	62	31	235					
TAREQ	60	93	124	124	62	61	0	90	30	0	124	124	61	61	62	0	124	124	0	31				
TLA ALI	450	310	284	62	186	148	150	360	150	89	217	465	278	122	93	155	217	93	210	155	155			
WDS1	330	248	124	124	186	124	0	360	120	118	217	248	123	154	124	0	124	124	0	186	155	463		
WDS2	330	248	0	124	0	0	0	360	120	0	217	248	0	0	124	0	0	124	0	0	0	0	0	

Table (4-3) Amman Area Telephone Network Links' Capacities.

2) Zarqa Area Network

Fig. (4-4), shows Zarqa area network, which is a mesh shaped network. Each node is connected to every other node in the network and to NSC1



The offered traffic for this network and the capacities of its links are shown in Tables (4-4) and (4-5).

2) Zarqa Area Network

Fig. (4-4), shows Zarqa area network, which is a mesh shaped network. Each node is connected to every other node in the network and to NSC1

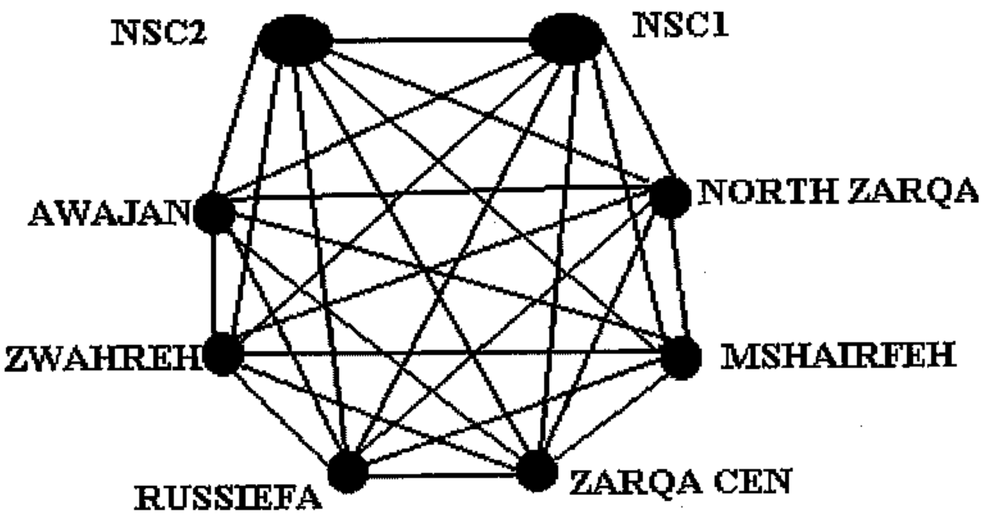


Fig. (4-4). Zarqa Area Telephone Network.

The offered traffic for this network and the capacities of its links are shown in Tables (4-4) and (4-5).

	ZARQA CEN	NORTH ZARQA	AWAJAN	ZWAHREH	RUSSIEFA	MSHAIRFEH
NORTH ZARQA	149.9					
AWAJAN	74.8	19.5				
ZWAHREH	43.8	16.1	5.2			
RUSSIEFA	63.6	18.2	22.7	8.7		
MSHAIRFEH	13.7	5.1	5.8	0.9	24.2	
NSC1	219.8	59.6	36.2	13.3	75.9	39.6
NSC2	238.53	81.5	39.2	15.2	76.3	44.5

Table (4-4) Zarqa Area Telephone Network Offered Traffic in Erlangs.

	ZARQA CEN	NORTH ZARQA	AWAJAN	ZWAHREH	RUSSIEFA	MSHAIRFEH
NORTH ZARQA	310					
AWAJAN	152	153				
ZWAHREH	93	93	122			
RUSSIEFA	217	217	153	124		
MSHAIRFEH	153	153	184	91	153	
NSC1	430	154	92	61	123	123
NSC2	61	154	123	92	185	123

Table (4-5) Zarqa Area Telephone NetworkLinks' Capacities.

3) Irbid Area Network

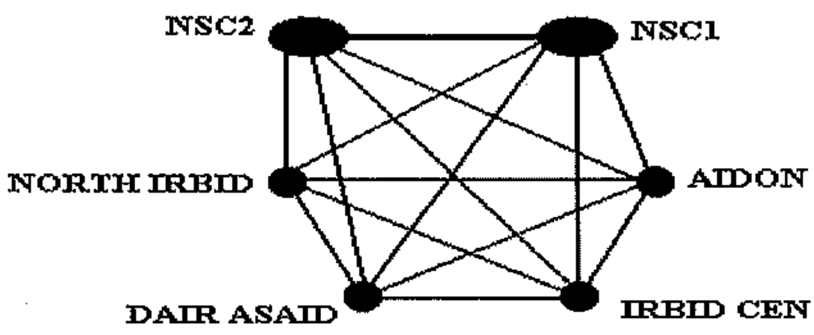


Fig (4-5). Irbid Area Telephone Network. rks.

AIDON	AIDON			
DAIR ASAID	40.4	DAIR ASAID		
IRBID CEN	65	12.3	IRBID CEN	
NORTH IRBID	60.7	4.3	40.5	NORTH IRBID
NSC1	123.3	45.7	168.9	92.9
NSC2	148.6	50.3	91	152.1

Table (4-6) Irbid Area Telephone Network Offered Traffic.

AIDON	AIDON			
DAIR ASAID	150	DAIR ASAID		
IRBID CEN	217	217	IRBID CEN	
NORTH IRBID	123	123	310	NORTH IRBID
NSC1	247	90	309	216
NSC2	247	90	91	247

Table (4-7) Irbid Area Telephone Network Links' Capacities.

4) Ma'an Area Network

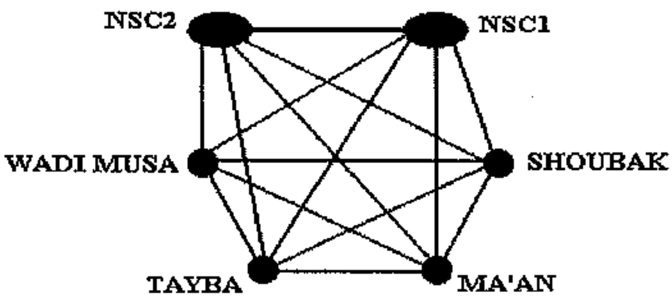


Fig (4-6). Ma'an Area Telephone network.

WADI MUSA	WADI MUSA			
SHOUBAK	26.1	SHOUBAK		
TAYBA	40.3	3.4	TAYBA	
MA'AN	60.3	11.5	50.9	MA'AN
NSC1	15.2	14.3	10.5	74.80
NSC2	32.3	30.1	25.3	65.50

Table (4-8) Ma'an Area Telephone Network Offered Traffic in Erlangs.

WADI MUSA	WADI MUSA			
SHOUBAK	120	SHOUBAK		
TAYBA	120	150	TAYBA	
MA'AN	150	210	210	MA'AN
NSC1	60	60	61	153
NSC2	60	60	60	92

Table (4-9) Ma'an Area Telephone Network Links' Capacities.

5) Karak Area Network

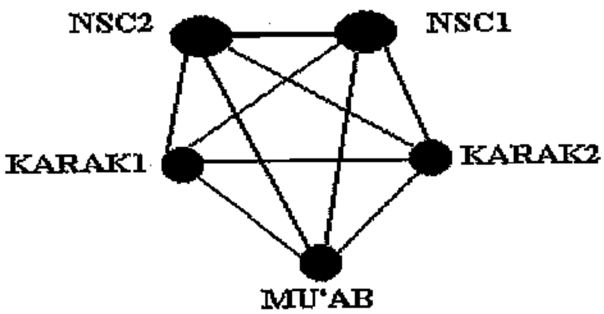


Fig. (4-7). Karak Area Telephone Network.

KARAK1	KARAK1		
KARAK2	90	KARAK2	
MU'AB	12.1	6.3	MU'AB
NSC1	86.2	36.3	5.4
NSC2	164.1	60.4	13.4

Table (4-10) Karak Area Telephone Network Offered Traffic in Erlangs.

KARAK1	KARAK1		
KARAK2	210	KARAK2	
MU'AB	150	181	MU'AB
NSC1	90	60	60
NSC2	240	123	61

Table (4-11) Karak Area Telephone Network Links' Capacities.

6) Salt Area Telephone Network

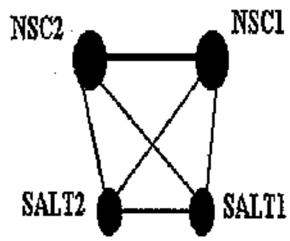


Fig. (4-8). Salt Area Telephone Network.

ALT1	SALT1	
SALT2	54	SALT2
NSC1	144.4	46.5
NSC2	1.8	0.97

Table (4-12) Salt Area Telephone Network Offered Traffic in Erlangs.

SALT1	SALT1	
SALT2	105	SALT2
NSC1	157	92
NSC2	60	92

Table (4-13) Salt Area Telephone Network Links' Capacities.

7) Madaba Area Telephone Network

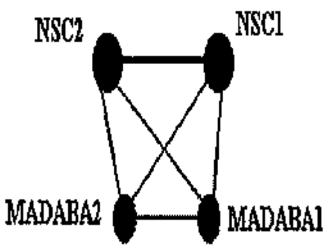


Fig (4-9). Madaba Area Telephone Network.

MADABA1	MADABA1	
MADABA2	28.3	MADABA2
NSC1	76.1	26.9
NSC2	57.7	29.2

Table (4-14) Madaba Area Telephone Network Offered Traffic in Erlangs.

MADABA1	MADABA1	
MADABA2	120	MADABA2
NSC1	268	154
NSC2	60	92

Table (4-15) Madaba Area Telephone Network Links' Capacities.

8) <u>Jarash Area Telephone Network</u>

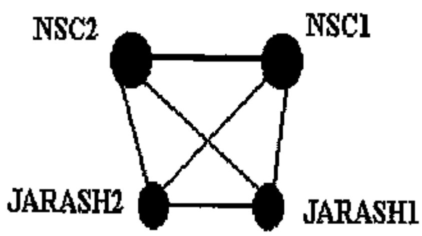


Fig. (4-10). Jarash Area Telephone Network.

JARASH1	JARASH1	
JARASH2	108.3	JARASH2
NSC1	25.4	30.3
NSC2	75.8	60.9

Table (4-16) Jarash Area Telephone Network Offered Traffic in Erlangs.

JARASH1	JARASH1	
JARASH2	150	JARASH2
NSC1	60	60
NSC2	123	123

Table (4-17) Jarash Area Telephone Network Links' Capacities.

9) Aqaba Area Telephone Network

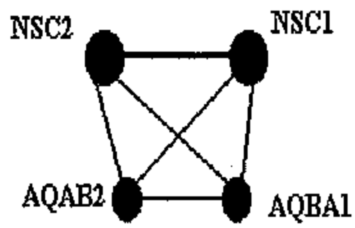


Fig. (4-11). Aqaba Area Telephone Network.

AQBA1	AQBA1	
AQAB2	57.1	AQAB2
NSC1	45.4	41.3
NSC2	49.1	116.4

Table (4-18) Aqaba Area Telephone Network Offered Traffic in Erlangs.

AQBA1	AQBA1	
AQAB2	220	AQAB2
NSC1	60	61
NSC2	61	180

Table (4-19) Aqaba Area Telephone Network Links' Capacities.

10) Ajloun Center

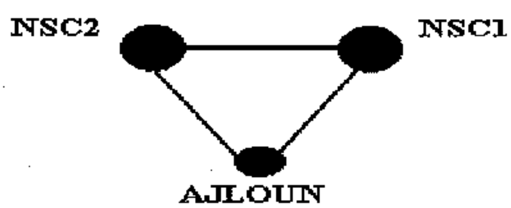


Fig. (4-12). Ajloun Center.

AJLOUN	AJLOUN
NSC1	150
NSC2	153

Table (4-20) Ajloun Links' Capacities.

AJLOUN	AJLOUN
NSC1	101.2
NSC2	70.4

Table (4-21) Ajloun Offered Traffic.

11) Mafraq Center

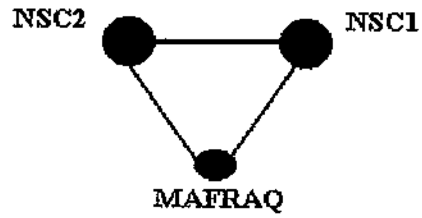


Fig. (4-13). Mafraq Center.

MAFRAQ	MAFRAQ
NSC1	85.4
NESC2	102.7

Table (4-22) Mafraq Offered Traffic.

MAFRAQ	MAFRAQ
NSC1	153
NESC2	202

Table (4-23) Mafraq Links' Capacities.

4.3 LCT Results for the Jordan Network

In the Jordan telephone network there is only one possible alternative route in the higher network to carry the traffic over a two-link path when there is only one failure. The traffic of the failed link will be distributed in proportion to the number of nodes in the lower networks. When there are two failures at a time, there are states in the higher network where the offered traffic to the failed links will be lost due to the absence of an alternative rout and these states are:

1-The two links that connect any lower node with NSC1 and NSC2 are failed.

2-Any link that connects any lower node with a higher node, and the link that completes its alternative route are failed.

Since the availability of the links is unknown, we will make our calculations when the network is homogenous and have equal high link availability, around 0.99 and when the link availability is less than 0.95. The inputs to our program are number of nodes in the higher network, number of nodes in the lower level for each lower network, offered traffic matrix for lower networks and link capacity matrix for lower networks. The offered traffic matrix in the higher network is unknown, hence we assumed that the traffic in each link in the higher network is distributed in proportion to the number of nodes in the lower networks as mentioned in chapter III. The same assumption is made for the Jordan telephone Network when there are failures to rout the traffic. Fig. (4-14) shows a flowchart to calculate the expected lost call traffic using method 1for a network, which have two levels of switching when there are no failure, single failure and two failures occurring at a time.

The Occurrence probabilities of the states when there is no failure, single failure and two failures using the Li method (method 1) (1984) is shown in Table (4-24) for two cases of links availability 0.995 and 0.94. When method 2 for random simulation of states is considered, and k replication are used, Table (4-25) shown the occurrence probabilities of these sates among all k=1000 simulated states which admit no failure, one failure and two failures at a time, and this respective summation.

Comparing Table (4-24) and (4-25) shows that both methods are in agreement for networks of high link availability.

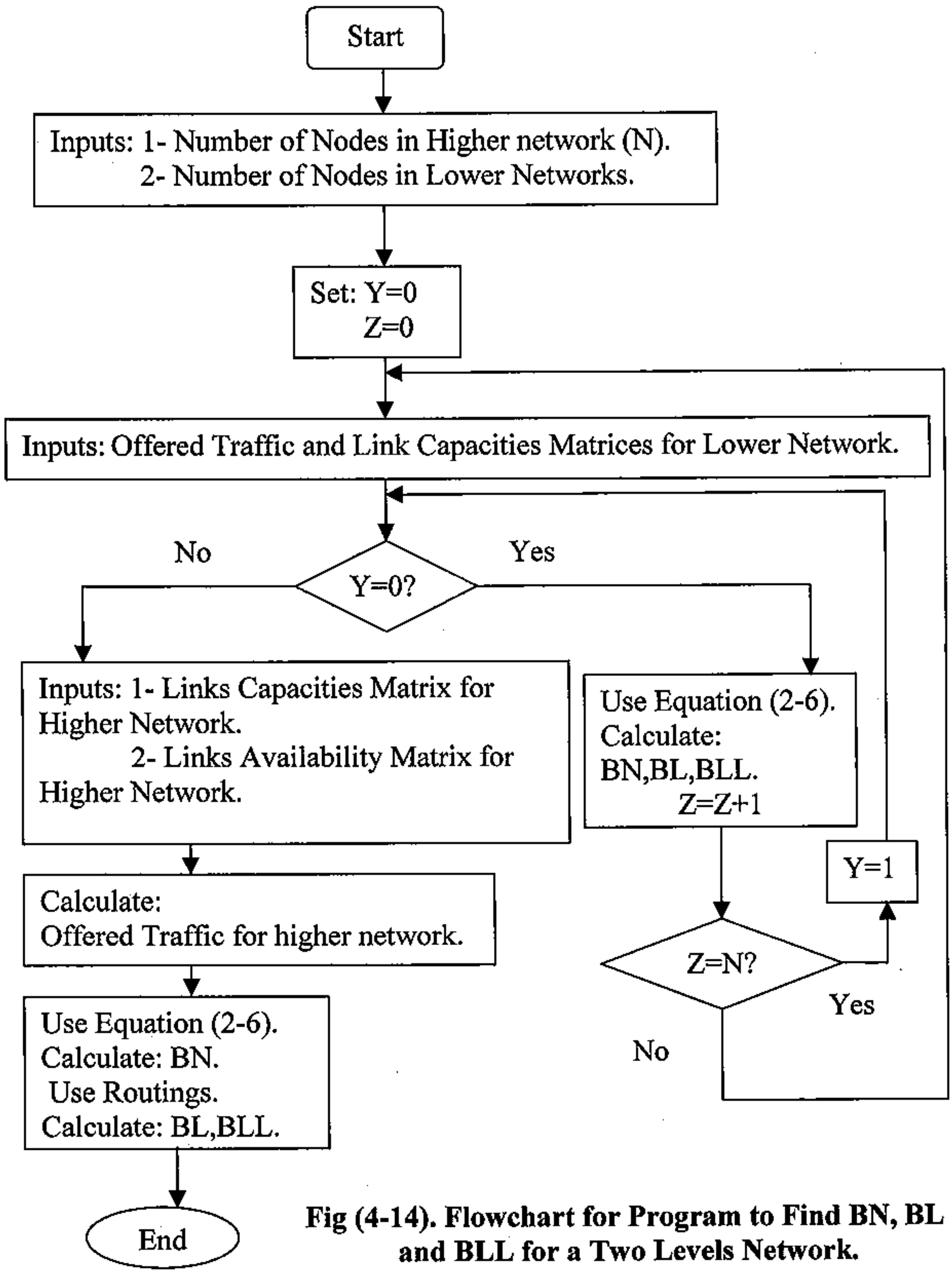


Fig (4-14). Flowchart for Program to Find BN, BL and BLL for a Two Levels Network.

	Node	Links	P_N 0.9950	P_N 0.9400	P_L 0.9950	P_L 0.9400	P_{LL} 0.9950	P_{LL} 0.9400	Sum 0.9950	Sum 0.9400
AMMAN	22	175	0.4160	1.98E-05	0.3660	2.20E-04	0.1590	1.20E-03	0.9410	0.0014
ZARQA	6	15	0.9280	0.3950	0.0700	0.3780	0.0010	0.1690	0.9990	0.9420
SALT	2	1	0.9950	0.9400	0.0050	0.0600			1.0000	1.0000
MADABA	2	1	0.9950	0.9400	0.0050	0.0600			1.0000	1.0000
IRBID	4	6	0.9700	0.6900	0.0290	0.2640	3.60E-04	0.0420	0.9990	0.9960
JARASH	2	1	0.9950	0.9400	0.0050	0.0600			1.0000	1.0000
MAFRAQ	1	0								
AJLOUN	1	0								
KARAK	3	3	0.9850	0.8300	0.0140	0.1560	7.40E-05	0.0100	0.9990	0.9960
MA'AN	4	6	0.9700	0.6900	0.0290	0.2640	3.60E-04	0.0420	0.9990	0.9960
AQABA	2	1	0.9950	0.6900	0.0050	0.0600			1.0000	1.0000
NATIONAL	2	102	0.6000	1.80E-03	0.3070	0.0120	0.0600	0.0290	0.9670	0.0430
	51	311								

Table (4-24) State Occurrece probabilities for Jordan Telphnone Network (Method 1).

	Node	Links	P_N 0.9950	P_N 0.9400	P_L 0.9950	P_L 0.9400	P_{LL} 0.9950	P_{LL} 0.9400	Sum 0.9950	Sum 0.9400
AMMAN	22	175	0.3610	0	0.3900	0	0.1810	0.0020	0.9320	0.0020
ZARQA	6	15	0.9220	0.4120	0.0750	0.3490	0.0030	0.1740	1.0000	0.9350
SALT	2	1	0.9950	0.9250	0.0050	0.0750			1.0000	1.0000
MADABA	2	1	0.9960	0.9430	0.0040	0.0570			1.0000	1.0000
IRBID	4	6	0.9720	0.6940	0.0280	0.2630		0.0400	1.0000	0.9970
JARASH	2	1	0.9960	0.9500	0.0040	0.0500			1.0000	1.0000
MAFRAQ	1	0								
AJLOUN	1	0								
KARAK	3	3	0.9840	0.8330	0.0160	0.1620		0.0040	1.0000	0.9990
MA'AN	4	6	0.9720	0.6690	0.0280	0.2760		0.0480	1.0000	0.9930
AQABA	2	1	0.9940	0.9300	0.0060	0.0700			1.0000	1.0000
NATIONAL	2	102	0.6110	3.00E-03	0.3090	0.0120	0.0600	0.0360	0.9800	0.0510
	51	311			k=1000					

Table (4-25) State Occurrece probabilities for Jordan Telphnone Network (Method 2)

In these tables, columns P_L and P_{LL} represent the summation of state occurrence probabilities for single and two failures. We see from Tables (4-24) and (4-25) that the summation of state occurrence probabilities when there are no failure, single failure and two failure is greater than 0.93 for all networks. So our approximation of taking no failure, single failure and two failures is sufficient when the link availability is 0.995. When the links availability is 0.94, method 1 for Amman area telephone network and for the national network will inaccurate and that we have to consider more than two links to fail at a time to achieve the most probable states that will occur.

Table (4-26) represents the results of weighted lost call traffic, WLCT, for the Jordan telephone network using method 1. When simulation method 2 was used, the results for lost call traffic are as shown in Table (4-27). Both tables present the results for the Jordan telephone network when routing is used.

	BN	BL	BLL	WLCT
AMMAN	38.84	70.90	153.2	66.47
ZARQA	1.6E-09	31.50	63.00	2.27
SALT	1.4E-08	54.00		0.27
MADABA	3.6E-36	28.80		0.14
IRBID	1.4E-20	37.20	74.40	1.11
JARASH	6.9E-44	50.30		0.25
MAFRAQ				
AJLOUN				
KARAK	1.7E-25	36.10	72.30	0.51
MA'AN	7.7E-21	32.10	64.20	0.95
AQABA	1.1E-58	57.10		0.29
NATIONAL	10.39	11.12	30.53	11.45
Availability=0.995	49.2	408.5	445.9	83.2

WLCT: weighted LCT

Table (4-26) Average Lost Call Traffic for Jordan Telephone Network (Method 1)

	LCT
AMMAN	72.43
ZARQA	3.50
SALT	0.54
MADABA	0.28
IRBID	1.23
JARASH	0.15
MAFRAQ	
AJLOUN	
KARAK	0.67
MA'AN	0.61
AQABA	0.52
NATIONAL	12.57
	92.25

Link availability: 0.995
K=1000

Table (4-27) Average Lost Call Traffic for Jordan Telephone Network (Method 2).

Now if we assume that there is no routing in the higher network for the Jordan telephone network, the results using the two methods are shown in Tables (4-28) and (4-29).

	BN	BL	BLL	WLCT
AMMAN	38.84	70.90	153.2	66.47
ZARQA	1.6E-09	31.50	63.00	2.27
SALT	1.4E-08	54.00		0.27
MADABA	3.6E-36	28.80		0.14
IRBID	1.4E-20	37.20	74.40	1.11
JARASH	6.9E-44	50.30		0.25
MAFRAQ				
AJLOUN				
KARAK	1.7E-25	36.10	72.30	0.51
MA'AN	7.7E-21	32.10	64.20	0.95
AQABA	1.1E-58	57.10		0.29
NATIONAL	10.39	82.20	152	40.59
	49.2	480.2	579.1	112.9

Link availability: 0.995

Table (4-28) Average Lost Call Traffic for Jordan Telephone Network (Method 1).

	WLCT
AMMAN	73.43
ZARQA	3.50
SALT	0.54
MADABA	0.28
IRBID	1.23
JARASH	0.15
MAFRAQ	
AJLOUN	
KARAK	0.67
MA'AN	0.61
AQABA	0.52
NATIONAL	38.02
Availability=0.995 K=1000	118.9

Table (4-29) Average Lost Call Traffic for Jordan Telephone Network (Method 2).

If the links availability is 0.94 then method 1 requires that states with more than two links failing at a time. However, method 2 can be used which have no restriction on the number of failures in any simulated states. The results of using method 2, with no routing, is shown in Table (4-30) for k=1000 replications.

	LCT
AMMAN	589.91
ZARQA	23.70
SALT	5.40
MADABA	2.26
IRBID	15.63
JARASH	5.42
MAFRAQ	
AJLOUN	
KARAK	12.59
MA'AN	12.55
AQABA	1.14
NATIONAL	200.10
	868.7

Availability: 0.94

Table (4-30) Average Lost Call Traffic for Jordan Telephone Network (Method 2).

We can see from Table (4-26) and table (4-27) the maximum lost call traffic happens in Amman telephone network because its links have a high offered traffic with respect to their capacity and especially Tla'a Alali center links. Because Tla'a Alali is a commercial and business area and most internet service provider companies lie their.

4.4 Conclusions

This thesis have discussed the idea of lost call traffic as a reliability measure for circuit switched networks which have two levels of switching, subject to random failure and using different routing techniques in case of failure and overload. The lost call traffic parameter was evaluated as a statistical average by weighting the lost call traffic values of network states by their respective state occurrence probabilities.

Results show that as the network size is increased or the links availability aren't high enough, we will have to consider more than two links to fail at a time, and this leads to large number of states to be considered. As an alternative method which could be used when links availability aren't known, we can consider randomly failing simulated states and calculate the lost call traffic for each state, then average the results. This method has been tested and has given good estimation of the lost call traffic without having to enumerate specific states.

This work has shown that lost call traffic related reliability under conditions of random failure can be numerically evaluated, and that its values depend on a number of issues, such- the capacity of the network links and their availability, in addition to the routing disciplines to be invoked whenever failure or traffic overload are detected. This approach is suitable when considering large networks for which more accurate methods can't be applied. Noting that operating conditions of networks could be complex and in particular the issue of routing. The present method was successful even for the most complicated routing discipline.

For future work we suggest to develop this method to be suitable for another hierarchical networks such as:

- A local area network (LAN). Connect workstations in an office or building.

- The internet (packet switched network).

Appendix

```
%*************************************************************
%*      Lost Call Traffic in a Circuit Switching Network which has
*
%*      Two Levels of Switching using Different Routing Approaches
%*************************************************************
N=input ('Enter the number of higher level switches ');
n=[];
xx=[];;
xxx=[];
for i=1:N
n(1,i)=input ('Enter the number of lower level switches ');
end
for i=1:N
for j=i+1:N
xx(i,j)=n(1,i)+n(1,j);
end
end
xxx=sum(xx,2);
pp=[];
cp=[];
BN=[];
BL=[];
BLL=[];
BLLL=[];
for s=1:N+1
a=[];
a6=[];
c6=[];
c=[];
c2=[];
if s==N+1
NN=N;
else
NN=n(1,s)+1;
end
for j=1:NN
if j~=NN
for k=j+1:NN
if s~=N+1
if s~=N
```

```
a(j,k)=input ('The direct offered traffic from i toj =');
else
if j~=1
a(j,k)=input ('The direct offered traffic from i toj =');
c(j,k)=input ('Link Capacity from i to j =');
else
a(j,k)=0;
c(j,k)=0;
end
end
end
if s~=N
c(j,k)=input ('Link Capacity from i to j =');
c6(j,k)=c(j,k);
c6(k,j)=c6(j,k);
end
if s==N+1
A(j,k)=input ('The availability from i to j=');
A(k,j)=A(j,k);
end
if s~=N+1
if j==1
pp(s,k-1)=a(j,k);
cp(s,k-1)=c(j,k);
end
end
end
end
end
if s~=N+1
c1=c;
a1=a;
c=[];
a=[];
for j=2:NN
if j~=NN
for k=j+1:NN
a(j-1,k-1)=a1(j,k);
c(j-1,k-1)=c1(j,k);
end
end
end
NN=n(1,s);
```

```
else
for j=1:N
for k=1:max(n)
if j~=N
a(j,NN+k)=pp(j,k);
end
c(j,NN+k)=cp(j,k);
end
end
ss=[];
sss=[];
ss=sum(a,2);
offered=0;
for j=1:NN-1
for k=j+1:NN
a(j,k)=ss(j,1)*(n(1,j)+n(1,k))/xxx(j,1);
if k~=NN
ss(k,1)=ss(k,1)-a(j,k);
else
offered=offered+a(j,k);
end
a6(j,k)=a(j,k);
a6(k,j)=a(j,k);
end
end
disp ('The offerd of the first');
n(1,s-1)
disp('Links must be equal to');
for kl=N+1:N+n(1,s-1)
a(N,kl)=input ('The direct offered traffic from i toj =');
c(N,kl)=input ('Link capacity from i to j =');
c6(N,kl)=c(N,kl);
c6(kl,N)=c6(N,kl);
end
ss(N,1)=offered;
NN=length(a);
end
sss=ss;
if s==N+1
z1=N;
for j=1:z1
for k=j+1:N
if a(j,k)/c(j,k)>1
```

```
as=[];
for j1=1:N
if j1~=j
if j1~=k
as(1,j1)=c6(j,j1)+c6(j1,k)-a6(j,j1)-a6(j1,k);
end
end
end
p=max(as);
for i1=1:length(as)
if as(1,i1)==p
j1=i1;
end
end
if j1>j
a(j,j1)=a(j,j1)+a(j,k)-c(j,k);
a6(j,j1)=a(j,j1);
a6(j1,j)=a6(j,j1);
else
a(j1,j)=a(j1,j)+a(j,k)-c(j,k);
a6(j1,j)=a(j1,j);
a6(j,j1)=a6(j1,j);
end
if k>j1
a(j1,k)=a(j1,k)+a(j,k)-c(j,k);
a6(j1,k)=a(j1,k);
a6(k,j1)=a6(j1,k);
else
a(k,j1)=a(k,j1)+a(j,k)-c(j,k);
a6(k,j1)=a(k,j1);
a6(j1,k)=a6(k,j1);
end
a(j,k)=c(j,k);
a6(j,k)=a(j,k);
a6(k,j)=a(j,k);
end
end
end
else
z1=n(1,s)-1;
end
a1=[];
c1=[];
```

```
a1=a;
c2=c;
B4=[];
BF=[];
zf=0;
% Calculate the LCT when there is no failure and the Expected LCT
when there is single failure
for jj=1:5
for ff=1:2
if ff==1
for j=1:z1
for k=j+1:NN
if a1(j,k)~=0
if s~=N+1
 a(j,k)=a1(j,k);        % Offered Traffic is lost in lower networks in case of
                                   failure
c(j,k)=0;
else
if jj==1              % Case (1-A)
if k<=N
a2=[];
for i1=1:N
for j1=i1+1:N
a2(i1,j1)=c2(i1,j1)-a1(i1,j1);
a2(j1,i1)=a2(i1,j1);
end
end
as=[];
for j1=1:N
if j1~=j
if j1~=k
as(1,j1)=a2(j,j1)+a2(j1,k);
end
end
end
p=max(as);
for i1=1:length(as)
if as(1,i1)==p
j1=i1;
end
end
if j1>j
a(j,j1)=a1(j,j1)+a1(j,k);
```

```
else
a(j1,j)=a1(j1,j)+a1(j,k);
end
if k>j1
a(j1,k)=a1(j1,k)+a1(j,k);
else
a(k,j1)=a1(k,j1)+a1(j,k);
end
a(j,k)=0;
else
ss=[];
ss(1,1)=sss(1,1)-a(j,k);
if j>1
for k22=1:j-1
a(k22,j)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,j);
end
end
for j1=1:N
if j1~=N
for k1=j1+1:N
if j1==j
a(j1,k1)=ss(j1,1)*(n(1,j1)+n(1,k1))/xxx(j,1);
end
end
end
end
a(j,k)=a1(j,k);
c(j,k)=0;
end
elseif jj==2             % Case (2-A)
if k<=N
a2=[];
for i1=1:N
for j1=i1+1:N
a2(i1,j1)=c2(i1,j1)-a1(i1,j1);
a2(j1,i1)=a2(i1,j1);
end
end
as=[];
for j1=1:N
if j1~=j
if j1~=k
```

```
as(1,j1)=a2(j,j1)+a2(j1,k);
end
end
end
p=max(as);
for i1=1:length(as)
if as(1,i1)==p
j1=i1;
end
end
if j1>j
a(j,j1)=a1(j,j1)+a1(j,k);
else
a(j1,j)=a1(j1,j)+a1(j,k);
end
if k>j1
a(j1,k)=a1(j1,k)+a1(j,k);
else
a(k,j1)=a1(k,j1)+a1(j,k);
end
a(j,k)=0;
end
elseif jj==3        % Case (2-B)
if k<=N
ks=0;
while ks==0
j1=ceil(N.*rand(1));
if j1~=j
if j1~=k
ks=1;
end
end
end
if j1>j
a(j,j1)=a1(j,j1)+a1(j,k);
else
a(j1,j)=a1(j1,j)+a1(j,k);
end
if k>j1
a(j1,k)=a1(j1,k)+a1(j,k);
else
a(k,j1)=a1(k,j1)+a1(j,k);
end
```

```
a(j,k)=0;
end
elseif jj==4   % Case (1-B)
if k<=N
ks=0;
while ks==0
j1=ceil(N.*rand(1));
if j1~=j
if j1~=k
ks=1;
end
end
end
if j1>j
a(j,j1)=a1(j,j1)+a1(j,k);
else
a(j1,j)=a1(j1,j)+a1(j,k);
end
if k>j1
a(j1,k)=a1(j1,k)+a1(j,k);
else
a(k,j1)=a1(k,j1)+a1(j,k);
end
a(j,k)=0;
else
ss=[];
ss(1,1)=sss(1,1)-a(j,k);
if j>1
for k22=1:j-1
a(k22,j)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,j);
end
end
for j1=1:N
if j1~=N
for k1=j1+1:N
if j1==j
a(j1,k1)=ss(j1,1)*(n(1,j1)+n(1,k1))/xxx(j,1);
end
end
end
end
a(j,k)=a1(j,k);
```

```
c(j,k)=0;
end
else
if k<=N
as=[];
ps=[];
for j1=1:N
if j1~=j
if j1~=k
as(1,j1)=A(j,j1)+A(j1,k);
ps(1,j1)=c6(j,j1)+c6(j1,k)-a6(j,j1)-a6(j1,k);
end
end
end
kk=0;
df=0;
while kk==0
p=max(as);
for i1=1:length(as)
if as(1,i1)==p
if df==inf
j3=i1;
kk=1;
end
if ps (1,i1)>0
j3=i1;
kk=1;
end
end
end
as(1,i1)=0;
df=df+1;
end
j1=j3;
if j1>j
a(j,j1)=a1(j,j1)+a1(j,k);
else
a(j1,j)=a1(j1,j)+a1(j,k);
end
if k>j1
a(j1,k)=a1(j1,k)+a1(j,k);
else
a(k,j1)=a1(k,j1)+a1(j,k);
```

```
end
a(j,k)=0;
else          % Case (1-C)
ss=[];
ss(1,1)=sss(1,1)-a(j,k);
if j>1
for k22=1:j-1
a(k22,j)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,j);
end
end
for j1=1:N
if j1~=N
for k1=j1+1:N
if j1==j
a(j1,k1)=ss(j1,1)*(n(1,j1)+n(1,k1))/xxx(j,1);
end
end
a(j,k)=a1(j,k);
c(j,k)=0;
end
end
end
end
end
a3=a;
c4=c;
B=[];
B1=[];
B2=[];
B3=[];
w=0;
while w==0
if zf==0
a=a1;
c=c2;
w=0;
else
a=a3;
c=c4;
end
% Using Erlang B formula to Calculate the lost call traffic when there is
no failure and the expected LCT when there is single failure
```

```
for x=1:z1;
for y=x+1:NN;
m2(x,y)=1;
if c(x,y)~=0
c1=1:c(x,y);

if c(x,y)>100
D= ((a(x,y).^(c(x,y)/5))/((prod(c1(1:ceil(c(x,y)/5))))));
D1=D*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)/5)+1):ceil(c(x,y)*2/5)))
)));
D2=D1*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)*2/5)+1):ceil(c(x,y)*3/
5))))));
D3=D2*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)*3/5)+1):ceil(c(x,y)*4/
5))))));
d(x,y)=D3*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)*4/5)+1):c(x,y))))));
for ks=1:c(x,y)
k1=1:ks;
if ks>100
M=((a(x,y).^(ks/5))/(prod(k1(1:ceil(ks/5)))));
M1=M*((a(x,y).^(ks/5))/(prod(k1(ceil((ks/5)+1)):ceil(ks*2/5)))));
M2=M1*((a(x,y).^(ks/5))/(prod(k1(ceil((ks*2/5)+1)):ceil((ks*3)/5)))));
M3=M2*((a(x,y).^(ks/5))/(prod(k1(ceil(((ks*3)/5)+1):ceil((ks*4)/5))))));
m(x,y)=M3*((a(x,y).^(ks/5))/(prod(k1(ceil(((ks*4)/5)+1):ks))));
m2(x,y)=m2(x,y)+m(x,y);
else
m(x,y)=(a(x,y).^ks)/prod(k1);
m2(x,y)=m2(x,y)+m(x,y);
end
end
else
d(x,y)=(a(x,y)^c(x,y))/prod(c1);
for ks=1:c(x,y)
k1=1:ks;
m(x,y)= (a(x,y)^ks)/prod(k1);
m2(x,y)=m2(x,y)+m(x,y);
end
end
else
d(x,y)=1;
m2(x,y)=1;
end
B(x,y)=d(x,y)/m2(x,y);
B1(x,y)=a(x,y)*B(x,y);
```

```
end
end
B2=sum(B1);
B3=sum(B2)/2;
B4(j,k)=B3;
B4(k,j)=B4(j,k);
a=a1;
c=c2;
if zf==1
BF=B4;
w=1;
else
B5=sum(B4);
B6=sum(B5);
BN(jj,s)=B6
zf=1;
end
end
end
end
end
B5=sum(BF);
B6=sum(B5);
if s==N+1
L(1,s)=(((N)^2)-(N))/2+sum(n);
else
L(1,s)=(((NN)^2)-(NN))/2;
end
BL(jj,s)=B6/L(1,s)
else
B23=[];
n1=0;
a=a1;
i1=1;
j2=2;
k2=1;
c2=c;
q=3;
s1=1;
j1=2;
l=0;
L=[];
z=1;
```

```
end
end
end
if j3~=j2
if j3~=i1
if j3~=q
if j3~=k2
if q1~=q
if q1~=k2
if q1~=i1
if q1~=j2
if N<=5
as(1,j3)=a2(i1,j3)+a2(j3,j2);
ps(1,q1)=a2(k2,q1)+a2(q1,q);
end
if q1~=j3
as(1,j3)=a2(i1,j3)+a2(j3,j2);
ps(1,q1)=a2(k2,q1)+a2(q1,q);
end
end
end
end
end
end
end
end
end
end
end
p=max(as);
for i2=1:length(as)
if as(1,i2)==p
j3=i2;
end
end
p=max(ps);
for i2=1:length(ps)
if ps(1,i2)==p
q1=i2;
end
end
y3=y3+1;
if j3>i1
```

```
h1=i1;
h2=j3;
else
h1=j3;
h2=i1;
end
if j3>j2
h3=j3;
h4=j2;
else
h3=j2;
h4=j3;
end
if q1>k2
h5=k2;
h6=q1;
else
h5=q;
h6=k2;
end
if q>q1
h7=q1;
h8=q;
else
h7=q;
h8=q1;
end
if j3==q1
if i1==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2)+a1(k2,q);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==q
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q)+a1(i1,j2);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==k2
```

```
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
ss=[];
ss(1,1)=sss(1,1)-a(k2,q);
if k2>1
for k22=1:k2-1
a(k22,k2)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,k2);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==k2
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(k2,1);
end
end
end
end
a5=a;
a2=[];
for i11=1:N
```

```
for j11=i11+1:N
a2(i11,j11)=c2(i11,j11)-a1(i11,j11);
a2(j11,i11)=a2(i11,j11);
end
end
as=[];
for j3=1:N
if N==4
if (i1+j2+k2+q)==10
if j2==2
as(1,3)=a2(i1,3)+a2(3,j2);
else
as(1,2)=a2(i1,2)+a2(2,j2);
end
end
end
if j3~=i1
if j3~=j2
if j3~=k2
if j3~=q
as(1,j3)=a2(i1,j3)+a2(j3,j2);
end
end
end
end
end
p=max(as);
for i11=1:length(as)
if as(1,i11)==p
j3=i11;
end
end
if j3>i1
a(i1,j3)=a5(i1,j3)+a5(i1,j2);
else
a(j3,i1)=a5(j3,i1)+a5(i1,j2);
end
if j2>j3
a(j3,j2)=a5(j3,j2)+a5(i1,j2);
else
a(j2,j3)=a5(j2,j3)+a5(i1,j2);
end
a(i1,j2)=0;
```

```
a(k2,q)=a1(k2,q);
c(k2,q)=0;
y3=y3+1;
end
elseif q<=N
ss=[];
ss(1,1)=sss(1,1)-a(i1,j2);
if i1>1
for k22=1:i1-1
a(k22,i1)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,i1);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==i1
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(i1,1);
end
end
end
end
a5=a;
a2=[];
for i11=1:N
for j11=i11+1:N
a2(i11,j11)=c2(i11,j11)-a1(i11,j11);
a2(j11,i11)=a2(i11,j11);
end
end
as=[];
for j3=1:N
if N==4
if (i1+j2+k2+q)==10
if j2==2
as(1,2)=a2(k2,2)+a2(2,q);
else

if j2==3
as(1,2)=a2(k2,2)+a2(2,q);
else
as(1,4)=a2(k2,4)+a2(4,q);
end
```

```
end
end
end
if j3~=i1
if j3~=j2
if j3~=k2
if j3~=q
as(1,j3)=a2(k2,j3)+a2(j3,q);
end
end
end
end
end
p=max(as);
for i11=1:length(as)
if as(1,i11)==p
j3=i11;
end
end
if j3>k2
a(k2,j3)=a5(k2,j3)+a5(k2,q);
else
a(j3,k2)=a5(j3,k2)+a5(k2,q);
end
if q>j3
a(j3,q)=a5(j3,q)+a5(k2,q);
else
a(q,j3)=a5(q,j3)+a5(k2,q);
end
a(k2,q)=0;
a(i1,j2)=a1(i1,j2);
c(i1,j2)=0;
y3=y3+1;
else
ss=[];
ss1=[];
if i1==k2
ss(1,1)=sss(1,1)-a(k2,q)-a(i1,j2);
ss1=ss;
else
ss(1,1)=sss(1,1)-a(i1,j2);
ss1(1,1)=sss(1,1)-a(k2,q);
end
```

```
if k2>1
for k22=1:k2-1
a(k22,k2)=ss1(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss1(k22+1,1)=ss1(k22,1)-a(k22,k2);
end
end
if i1>1
for k22=1:i1-1
a(k22,i1)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,i1);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==k2
a(j11,k11)=ss1(j11,1)*(n(1,j11)+n(1,k11))/xxx(k2,1);
end
end
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==i1
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(i1,1);
end
end
end
end
a(i1,j2)=a1(i1,j2);
a(k2,q)=a1(k2,q);
c(i1,j2)=0;
c(k2,q)=0;
y3=y3+1;
end
end
elseif jj==2
if s~=N+1
a(i1,j2)=a1(i1,j2);
a(k2,q)=a1(k2,q);
c(i1,j2)=0;
c(k2,q)=0;
```

```
y3=y3+1;
else
if j2<=N
if q<=N
a2=[];
for i2=1:N
for j4=i2+1:N
a2(i2,j4)=c2(i2,j4)-a1(i2,j4);
a2(j4,i2)=a2(i2,j4);
end
end
as=[];
ps=[];
for j3=1:N
for q1=1:N
if N==4
if (i1+j2+k2+q)==10
if j2==2
as(1,3)=a2(i1,3)+a2(3,j2);
ps(1,2)=a2(k2,2)+a2(2,q);
else
as(1,2)=a2(i1,2)+a2(2,j2);
if j2==3
ps(1,2)=a2(k2,2)+a2(2,q);
else
ps(1,4)=a2(k2,4)+a2(4,q);
end
end
end
end
if j3~=j2
if j3~=i1
if j3~=q
if j3~=k2
if q1~=q
if q1~=k2
if q1~=i1
if q1~=j2
if N<=5
as(1,j3)=a2(i1,j3)+a2(j3,j2);
ps(1,q1)=a2(k2,q1)+a2(q1,q);
end
if q1~=j3
```

```
as(1,j3)=a2(i1,j3)+a2(j3,j2);
ps(1,q1)=a2(k2,q1)+a2(q1,q);
end
end
end
end
end
end
end
end
end
end
end
p=max(as);
for i2=1:length(as)
if as(1,i2)==p
j3=i2;
end
end
p=max(ps);
for i2=1:length(ps)
if ps(1,i2)==p
q1=i2;
end
end
y3=y3+1;
if j3>i1
h1=i1;
h2=j3;
else
h1=j3;
h2=i1;
end
if j3>j2
h3=j3;
h4=j2;
else
h3=j2;
h4=j3;
end
if q1>k2
h5=k2;
h6=q1;
```

```
else
h5=q;
h6=k2;
end
if q>q1
h7=q1;
h8=q;
else
h7=q;
h8=q1;
end
if j3==q1
if i1==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2)+a1(k2,q);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==q
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q)+a1(i1,j2);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
```

```
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
end
end
end
elseif jj==3
if s~=N+1
a(i1,j2)=a1(i1,j2);
a(k2,q)=a1(k2,q);
c(i1,j2)=0;
c(k2,q)=0;
y3=y3+1;
else
if j2<=N
if q<=N
ks=0;
while ks==0
j3=ceil(N.*rand(1));
q1=ceil(N.*rand(1));
if N==4
if (i1+j2+k2+q)==10
if j2==2
as(1,3)=a2(i1,3)+a2(3,j2);
ps(1,2)=a2(k2,2)+a2(2,q);
ks=1;
else
as(1,2)=a2(i1,2)+a2(2,j2);
ks=1;
if j2==3
ps(1,2)=a2(k2,2)+a2(2,q);
else
ps(1,4)=a2(k2,4)+a2(4,q);
end
end
end
end
if j3~=j2
if j3~=i1
if j3~=q
```

```
if j3~=k2
if q1~=q
if q1~=k2
if q1~=i1
if q1~=j2
ks=1;
end
end
end
end
end
end
end
end
end
y3=y3+1;
if j3>i1
h1=i1;
h2=j3;
else
h1=j3;
h2=i1;
end
if j3>j2
h3=j3;
h4=j2;
else
h3=j2;
h4=j3;
end
if q1>k2
h5=k2;
h6=q1;
else
h5=q;
h6=k2;
end
if q>q1
h7=q1;
h8=q;
else
h7=q;
h8=q1;
```

```
end
if j3==q1
if i1==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2)+a1(k2,q);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==q
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q)+a1(i1,j2);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
end
end
end
elseif jj==4
```

```
end
end
end
end
end
end
y3=y3+1;
if j3>i1
h1=i1;
h2=j3;
else
h1=j3;
h2=i1;
end
if j3>j2
h3=j3;
h4=j2;
else
h3=j2;
h4=j3;
end
if q1>k2
h5=k2;
h6=q1;
else
h5=q;
h6=k2;
end
if q>q1
h7=q1;
h8=q;
else
h7=q;
h8=q1;
end
if j3==q1
if i1==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2)+a1(k2,q);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
```

```
elseif j2==q
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q)+a1(i1,j2);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
ss=[];
ss(1,1)=sss(1,1)-a(k2,q);
if k2>1
for k22=1:k2-1
a(k22,k2)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,k2);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==k2
```

```
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(k2,1);
end
end
end
end
a5=a;
ks=0;
while ks==0
j3=ceil(N.*rand(1));
if N==4
if (i1+j2+k2+q)==10
if j2==2
j3=3;
ks=1;
else
j3=2;
ks=1;
end
end
end
if j3~=i1
if j3~=j2
if j3~=k2
if j3~=q
ks=1;
end
end
end
end
end
if j3>i1
a(i1,j3)=a5(i1,j3)+a5(i1,j2);
else
a(j3,i1)=a5(j3,i1)+a5(i1,j2);
end
if j2>j3
a(j3,j2)=a5(j3,j2)+a5(i1,j2);
else
a(j2,j3)=a5(j2,j3)+a5(i1,j2);
end
a(i1,j2)=0;
a(k2,q)=a1(k2,q);
c(k2,q)=0;
```

```
y3=y3+1;
end
elseif q<=N
ss=[];
ss(1,1)=sss(1,1)-a(i1,j2);
if i1>1
for k22=1:i1-1
a(k22,i1)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,i1);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==i1
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(i1,1);
end
end
end
end
a5=a;
ks=0;
while ks==0
j3=ceil(N.*rand(1));
if N==4
if (i1+j2+k2+q)==10
if j2==2
j3=2;
ks=1;
else
ks=1;
if j2==3
j3=2;
else
j3=4;
end
end
end
end
if j3~=i1
if j3~=j2
if j3~=k2
if j3~=q
```

```
ks=1;
end
end
end
end
end
if j3>k2
a(k2,j3)=a5(k2,j3)+a5(k2,q);
else
a(j3,k2)=a5(j3,k2)+a5(k2,q);
end
if q>j3
a(j3,q)=a5(j3,q)+a5(k2,q);
else
a(q,j3)=a5(q,j3)+a5(k2,q);
end
a(k2,q)=0;
a(i1,j2)=a1(i1,j2);
c(i1,j2)=0;
y3=y3+1;
else
ss=[];
ss1=[];
if i1==k2
ss(1,1)=sss(1,1)-a(k2,q)-a(i1,j2);
ss1=ss;
else
ss(1,1)=sss(1,1)-a(i1,j2);
ss1(1,1)=sss(1,1)-a(k2,q);
end
if k2>1
for k22=1:k2-1
a(k22,k2)=ss1(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss1(k22+1,1)=ss1(k22,1)-a(k22,k2);
end
end
if i1>1
for k22=1:i1-1
a(k22,i1)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,i1);
end
end
for j11=1:N
```

```
if j11~=N
for k11=j11+1:N
if j11==k2
a(j11,k11)=ss1(j11,1)*(n(1,j11)+n(1,k11))/xxx(k2,1);
end
end
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==i1
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(i1,1);
end
end
end
end
a(i1,j2)=a1(i1,j2);
a(k2,q)=a1(k2,q);
c(i1,j2)=0;
c(k2,q)=0;
y3=y3+1;
end
end
else
if s~=N+1
a(i1,j2)=a1(i1,j2);
a(k2,q)=a1(k2,q);
c(i1,j2)=0;
c(k2,q)=0;
y3=y3+1;
else
if j2<=N
if q<=N
as=[];
ps=[];
a1s=[];
p1s=[];
for j3=1:N
for q1=1:N
if N==4
if (i1+j2+k2+q)==10
if j2==2
```

```
as(1,3)=A(i1,3)+A(3,j2);
a1s(1,3)=c6(i1,3)+c6(3,j2)-a6(i1,3)-a6(3,j2);
ps(1,2)=A(k2,2)+A(2,q);
p1s(1,2)=c6(k2,2)+c6(2,q)-a6(2,q)-a6(k2,2);
else
as(1,2)=A(i1,2)+A(2,j2);
a1s(1,2)=c6(i1,2)+c6(2,j2)-a6(i1,2)-a6(2,j2);
if j2==3
ps(1,2)=A(k2,2)+A(2,q);
p1s(1,2)=c6(k2,2)+c6(2,q)-a6(2,q)-a6(k2,2);
else
ps(1,4)=A(k2,4)+A(4,q);
p1s(1,4)=c6(k2,4)+c6(4,q)-a6(4,q)-a6(k2,4);
end
end
end
end
if j3~=j2
if j3~=i1
if j3~=q
if j3~=k2
if q1~=q
if q1~=k2
if q1~=i1
if q1~=j2
as(1,j3)=A(i1,j3)+A(j3,j2);
a1s(1,j3)=c6(i1,j3)+c6(j3,j2)-a6(i1,j3)-a6(j3,j2);
ps(1,q1)=A(k2,q1)+A(q1,q);
p1s(1,q1)=c6(k2,q1)+c6(q1,q)-a6(q1,q)-a6(k2,q1);
end
end
end
end
end
end
end
end
end
end
kk=0;
df=0;
while kk==0
p=max(as);
```

```
for i12=1:length(as)
if as(1,i12)==p
if df==inf
kk=1;
j33=i12;
end
if a1s (1,i12)>0
j33=i12;
kk=1;
end
end
end
as(1,i12)=0;
df=df+1;
end
j3=j33;
kk=0;
df=0;
while kk==0
p=max(ps);
for i12=1:length(ps)
if ps(1,i12)==p
if df==inf
kk=1;
j33=i12;
end
if p1s (1,i12)>0
j33=i12;
kk=1;
end
end
end
ps(1,i12)=0;
df=df+1;
end
q1=j33;
y3=y3+1;
if j3>i1
h1=i1;
h2=j3;
else
h1=j3;
h2=i1;
```

```
end
if j3>j2
h3=j3;
h4=j2;
else
h3=j2;
h4=j3;
end
if q1>k2
h5=k2;
h6=q1;
else
h5=q;
h6=k2;
end
if q>q1
h7=q1;
h8=q;
else
h7=q;
h8=q1;
end
if j3==q1
if i1==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2)+a1(k2,q);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==q
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q)+a1(i1,j2);
a(i1,j2)=0;
a(k2,q)=0;
elseif j2==k2
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q)+a1(i1,j2);
a(h3,h4)=a1(h3,h4)+a1(i1,j2)+a1(k2,q);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
```

```
a(k2,q)=0;
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
a(h1,h2)=a1(h1,h2)+a1(i1,j2);
a(h5,h6)=a1(h5,h6)+a1(k2,q);
a(h3,h4)=a1(h3,h4)+a1(i1,j2);
a(h7,h8)=a1(h7,h8)+a1(k2,q);
a(i1,j2)=0;
a(k2,q)=0;
end
else
ss=[];
ss(1,1)=sss(1,1)-a(k2,q);
if k2>1
for k22=1:k2-1
a(k22,k2)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,k2);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==k2
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(k2,1);
end
end
end
end
a5=a;
as=[];
ps=[];
for j3=1:N
if N==4
if (i1+j2+k2+q)==10
if j2==2
as(1,3)=A(i1,3)+A(3,j2);
```

```
ps(1,3)=c6(i1,3)+c6(3,j2)-a6(i1,3)-a6(3,j2);
else
as(1,2)=A(i1,2)+A(2,j2);
ps(1,2)=c6(i1,2)+c6(2,j2)-a6(i1,2)-a6(2,j2);
end
end
end
if j3~=i1
if j3~=j2
if j3~=k2
if j3~=q
as(1,j3)=A(i1,j3)+A(j3,j2);
ps(1,j3)=c6(i1,j3)+c6(j3,j2)-a6(i1,j3)-a6(j3,j2);
end
end
end
end
end
kk=0;
df=0;
while kk==0
p=max(as);
for i12=1:length(as)
if as(1,i12)==p
if df==inf
kk=1;
j33=i12;
end
if ps (1,i12)>0
j33=i12;
kk=1;
end
end
end
as(1,i12)=0;
df=df+1;
end
j3=j33;
if j3>i1
a(i1,j3)=a5(i1,j3)+a5(i1,j2);
else
a(j3,i1)=a5(j3,i1)+a5(i1,j2);
end
```

```
if j2>j3
a(j3,j2)=a5(j3,j2)+a5(i1,j2);
else
a(j2,j3)=a5(j2,j3)+a5(i1,j2);
end
a(i1,j2)=0;
a(k2,q)=a1(k2,q);
c(k2,q)=0;
y3=y3+1;
end
elseif q<=N
ss=[];
ss(1,1)=sss(1,1)-a(i1,j2);
if i1>1
for k22=1:i1-1
a(k22,i1)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,i1);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==i1
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(i1,1);
end
end
end
end
a5=a;
as=[];
ps=[];
for j3=1:N
if N==4
if (i1+j2+k2+q)==10
if j2==2
as(1,2)=A(k2,2)+A(2,q);
ps(1,2)=c6(k2,2)+c6(2,q)-a6(2,q)-a6(k2,2);
else
if j2==3
as(1,2)=A(k2,2)+A(2,q);
ps(1,2)=c6(k2,2)+c6(2,q)-a6(2,q)-a6(k2,2);
else
as(1,4)=A(k2,4)+A(4,q);
```

```
ps(1,4)=c6(k2,4)+c6(4,q)-a6(4,q)-a6(k2,4);
end
end
end
end
if j3~=i1
if j3~=j2
if j3~=k2
if j3~=q
as(1,j3)=A(k2,j3)+A(j3,q);
ps(1,j3)=c6(k2,j3)+c6(j3,q)-a6(k2,j3)-a6(j3,q);
end
end
end
end
end
kk=0;
df=0;
while kk==0
p=max(as);
for i12=1:length(as)
if as(1,i12)==p
if df==inf
kk=1;
j33=i12;
end
if ps (1,i12)>0
j33=i12;
kk=1;
end
end
end
as(1,i12)=0;
df=df+1;
end
j33=q1;
if j3>k2
a(k2,j3)=a5(k2,j3)+a5(k2,q);
else
a(j3,k2)=a5(j3,k2)+a5(k2,q);
end
if q>j3
a(j3,q)=a5(j3,q)+a5(k2,q);
```

```
else
a(q,j3)=a5(q,j3)+a5(k2,q);
end
a(k2,q)=0;
a(i1,j2)=a1(i1,j2);
c(i1,j2)=0;
y3=y3+1;
else
ss=[];
ss1=[];
if i1==k2
ss(1,1)=sss(1,1)-a(k2,q)-a(i1,j2);
ss1=ss;
else
ss(1,1)=sss(1,1)-a(i1,j2);
ss1(1,1)=sss(1,1)-a(k2,q);
end
if k2>1
for k22=1:k2-1
a(k22,k2)=ss1(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss1(k22+1,1)=ss1(k22,1)-a(k22,k2);
end
end
if i1>1
for k22=1:i1-1
a(k22,i1)=ss(k22,1)*(n(1,k22)+n(1,j))/xxx(k22,1);
ss(k22+1,1)=ss(k22,1)-a(k22,i1);
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==k2
a(j11,k11)=ss1(j11,1)*(n(1,j11)+n(1,k11))/xxx(k2,1);
end
end
end
end
for j11=1:N
if j11~=N
for k11=j11+1:N
if j11==i1
a(j11,k11)=ss(j11,1)*(n(1,j11)+n(1,k11))/xxx(i1,1);
```

```
end
end
end
end
a(i1,j2)=a1(i1,j2);
a(k2,q)=a1(k2,q);
c(i1,j2)=0;
c(k2,q)=0;
y3=y3+1;
end
end
end
% Using Erlang B formula to Calculate the expected LCT when there are
two failures
B=[];
B1=[];
B2=[];
B3=[];
for x=1:z1;
for y=1:NN;
m2(x,y)=1;
if c(x,y)~=0
c1=1:c(x,y);
if c(x,y)>100
D= ((a(x,y).^(c(x,y)/5))/((prod(c1(1:ceil(c(x,y)/5))))));
D1=D*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)/5)+1):ceil(c(x,y)*2/5)))
)));
D2=D1*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)*2/5)+1):ceil(c(x,y)*3/
5))))));
D3=D2*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)*3/5)+1):ceil(c(x,y)*4/
5))))));
d(x,y)=D3*((a(x,y).^(c(x,y)/5))/((prod(c1(ceil((c(x,y)*4/5)+1):c(x,y))))));
for ks=1:c(x,y)
k1=1:ks;
if ks>100
M=((a(x,y).^(ks/5))/(prod(k1(1:ceil(ks/5)))));
M1=M*((a(x,y).^(ks/5))/(prod(k1(ceil((ks/5)+1)):ceil(ks*2/5))));
M2=M1*((a(x,y).^(ks/5))/(prod(k1(ceil((ks*2/5)+1)):ceil((ks*3)/5))));
M3=M2*((a(x,y).^(ks/5))/(prod(k1(ceil(((ks*3)/5)+1):ceil((ks*4)/5)))));
m(x,y)=M3*((a(x,y).^(ks/5))/(prod(k1(ceil(((ks*4)/5)+1):ks))));
m2(x,y)=m2(x,y)+m(x,y);
else
m(x,y)=(a(x,y).^ks)/prod(k1);
```

```
m2(x,y)=m2(x,y)+m(x,y);
end
end
else
d(x,y)=(a(x,y)^c(x,y))/prod(c1);
for ks=1:c(x,y)
k1=1:ks;
m(x,y)= (a(x,y)^ks)/prod(k1);
m2(x,y)=m2(x,y)+m(x,y);
end
end
else
d(x,y)=1;
m2(x,y)=1;
end
B(x,y)=d(x,y)/m2(x,y);
B1(x,y)=a(x,y)*B(x,y);
end
end
B2=sum(B1);
B3=sum(B2);
a=a1;
c=c2;
B23(1,y3)=B3;
end
end
o=i1;
u=j2;
if q==NN
if k2==z1
if j2==NN
i1=i1+1;
j2=i1+1;
else
j2=j2+1;
end
end
end
g=k2;
if q==NN
k2=k2+1;
if g==z1
k2=d1;
```

```
if u==NN-1
d1=d1+1;
k2=d1;
end
end
end
r=q;
if z==1
q=q+1;
if r==NN
if s1==1
q=3;
s1=2;
m1=4;
else
q=m1;
m1=m1+1;
end
end
end
l=l+1;
L(1,l)=r;
if r==NN
if g==z1
L1=[];
for p=2:l
L1(1,p-1)=L(1,p);
end
l=0;
Z=0;
x1=1;
w=1;
L=[];
end
end
if o==z1-1
if u==NN
if g==z1;
if r==NN
w=0;
E=1;
end
end
```

```
end
end
if w==1;
q=L1(1,x1);
x1=x1+1;
end
end
B5=sum(B23)/y3;
BLL(jj,s)=B5
end
end
end
end
```

References

Alsadi, M. 1997. Reliability of Circuit Switched Network. M.Sc. Thesis, University of Jordan, Amman, Jordan.

Ayoub, J and Sutri, M. 1997. Network Reliability Based on Lost Call Traffic. Fourth Communication Networks symposium, Manchester. July: 35-42.

Bear, D. 1980. Principles of Telecommunication Traffic Engineering. 2nd edition. A. Wheaton & Co., Ltd., Exeter. UK.

Cravis, H. 1981. Communication Network Analysis. An Arther D. Little, Inc. USA.

Fratta, L and Montanari, U. 1978. A Recursive Method Based on Case Analysis for Computing Network Terminal Reliability. IEEE Transaction On Communications. VOL. 26. No. 8 August: 1166-1177.

Gavish, B and Neuman, I. 1992. Routing in a Network with Unreliable Components. IEEE Transaction On Communications. VOL. 40. No. 7 July: 1248-1257.

Li, V and Silvester, J. 1984. Performance Analysis of Networks with Unreliable Components. IEEE Transaction On Communications. VOL. Com-23. No. 10, October: 1105-1110.

Medhi, D. 1994. A Unified Approach to Network Survivability for Teletraffic Networks: Models, Algorithms and Analysis. IEEE Transaction On Communications. VOL. 42. No. 2/3/4 February/ March/ April: 534-548.

Medhi, D. 1999. Reliability and fault tolerance. Wiley Encyclopedia of Electrical & electronic Engineering. Vol. 14, [Ed. J. G. Webster]: 213-218.

Qiao, L. and Qiao, S. 1998, A Robust and Efficient Algorithm for Evaluating Erlang's Formula. Technical Report CAS98-03. Department of Computing and Software, McMaster University. Canada. L8S 4L7, August.

Sanso, B, Soumis, F and Gendreau, M. 1991. On The Evaluation of Telecommunications Network Reliability Using Routing Models. IEEE Transaction On Communications. VOL. 39. No.10 October: 1494-1501.

مقدار الفقد في الحركة الهاتفية لشبكات الاتصال الهاتفية

إعداد

رامي عبد الفتاح الحموز

المشرف

الدكتور جميل أيوب

ملخص

تقوم هذه الرسالة على دراسة شبكات الاتصال الهاتفية ذات المستويين واعتماديتـــها في ظـــروف التعطـــل العشوائي لخطوط الاتصال. عندما يحصل عطل يتم اختبار عدة مسارات بديله لخط الاتصال المعطل، وتعتمد الدراسـة في قياس اعتمادية الشبكات على حساب مقدار الفقد في الحركة الهاتفية بين المقاسم.

في الدراسات السابقة تم تقديم عملية لحساب مقدار الفقد في الحركة لشبكات الاتصال الهاتفية ذات المستوى الواحد. وفي هذا البحث تم تطوير الأسلوب السابق بحيث يعالج شبكات الاتصال ذات المستويين، المســـتوى الأول يعبر عن شبكات الاتصال المحلية والمستوى الثاني يعبر عن شبكة الاتصال الوطنية. وتم تطبيق هذا الأسلوب على شبكة الأردن المحلية.

يجب أن يأخذ بعين الاعتبار في ظروف التعطل العشوائي أكبر قدر ممكن من حالات الشبكة، تم افـتراض أن خطوط التوصيل ذات اعتمادية عالية أو أن احتمالية تعطلها قليلة. حالات الشبكة ذات الاحتمالية العالية تكون كافية لأن تأخذ بعين الاعتبار، وهذه الحالات:

١-حالة عدم وجود تعطل في الشبكة.

٢-حالة وجود تعطل واحد في الشبكة.

٣-حالة وجود تعطلين في الشبكة.

يتم حساب مقدار الفقد لكل حالة في الشبكة عن طريق تطبيق معادلة (Erlang) واختبار اكثر من مسار بديل عند العطل. ومن ثم فإن نتائج الفقد توزن باحتمالية حدوثها من أجل الحصول على معدل لفقد الشبكة.

تم اختبار طريقة أخرى لحساب معدل الفقد في كمية الحركة الهاتفية وذلك عن طريق أخذ عينة من الحالات التي تكون عليها الشبكة وذلك بتوليد أرقام عشوائية على خطوط توصيل الشبكة لتقرير عطل الخط. هذه الطريقة لها أفضلية وخصوصا عندما تكون اعتمادية خطوط توصيل الشبكة قليلة.

هاتان الطريقتان تم اختبارهما وتطبيقهما على شبكة الأردن المحلية. ووجد أن هاتان الطريقتان تعطيان نتائج متقاربة.