Reliability Measurement and Enhancement of the Communication Networks

Mou Dasgupta
Department of Computer Science
and Engineering
Indian School of Mines
Dhanbad, India, 826004

G. P. Biswas
Department of Computer Science
and Engineering
Indian School of Mines
Dhanbad, India, 826004

ABSTRACT
Use of ICT (Information and Communication Technology) or the computer communication using electronic messaging has increased tremendously in recent years. Also the modern networks that support ICT are robust, i.e., its failure due to links, routing protocols, congestion etc is rare and as a result, the estimation of the overall reliability of the communication networks, which is #P- complete problem, is very important. This paper presents two linear time complexity algorithms for approximate assessment and the enhancement of the reliability of the given networks. The proposed techniques basically identify the node-pairs having lower reliability, insert communication links in them and calculate the increase in reliability on insertion iteratively until the satisfactory reliability is achieved. The simulated experimentation of the proposed algorithms have been done and compared with the existing methods, which show satisfactory performance.

Categories and Subject Descriptors
D.3.2 [**FORTRAN 77**]: Language Constructs and Features – *procedural, imperative suited to numeric computation and scientific computing.*

General Terms
Algorithms, Performance, Design, Reliability, Experimentation.

Keywords
ICT, Communication Networks, Network Reliability Estimation, Link Insertion and Enhancement of the Network Reliability.

1. INTRODUCTION
The ICT, which is the present days' need, plays a vital role almost in every aspect of the modern lives. For successful implementation and the increase in performance, the ICT, in addition to the others, requires highly reliable communication networks. Thus an important aspect of the communication network design is to analyze the reliability and insert additional links in the existing networks so that the overall throughput, delay, reliability etc are improved. The analysis of the reliability mainly focuses on following two key factors as

(i) Estimation of the reliability of the given network using efficient techniques, i.e. to know and measure the reliability parameters of the network.

(ii) Enhancement of the reliability by inserting the links between node-pairs of lower reliability, i.e. to improve them in the most efficient way.

There are different measures of reliability [2-4, 10-14, 19, 20]. All-terminal network reliability (also called overall network reliability) is calculated from the probability that each and every node in the network is connected to each other. One of the more useful reliability measures is the source-to-terminal (s-t) reliability, which is the probability that a given "source" node can communicate with a give "terminal" node. Related to this is the notion of K-terminal reliability, which is the probability that all nodes in a given set "K" can communicate.

Numerous algorithms and evaluation techniques have been described in the literature [2-5, 9-13] for the computation of above mentioned reliability measures. But all such methods are computationally intractable for networks of even moderate size. Even though in [2] Belovich and Konangi have proposed an algorithm in linear-time but it has an upper bound placed upon the in-degree of all nodes, which is practically not viable. There are many upper and lower bounds expressions for enumerative reliability analysis methods [2, 8, 9]. However they are too loose to be effective.

Thus, for any network with reliability as a parameter, its design and modification is a difficult task. The selection of optimal network topology with reliability constraint is an NP-hard combinatorial problem as these methods grow exponentially with network size. So, for networks of large size Genetic-algorithm based approaches are used as a new solution method for optimal design of networks considering reliability. In [15] Kumar, Pathak and Gupta developed a genetic algorithm considering distance, diameter and reliability to design and expand computer networks. Dengiz, Altiparmak and Smith in [17] focused on large backboned communication network design considering all-terminal network reliability and used a genetic algorithm, but customized it appreciably to the all-terminal design problem to give an effective, efficient optimization methodology.

The objective of this paper is to present a design strategy by inserting additional links, thereby enhancing the reliability of the network. Some heuristic design techniques are developed for the placement of links. The techniques presented in this paper are computationally tractable in all cases. It has been observed that using these network design modification methods a linear time solution can be obtained. The paper is organized in the following manner. Section II depicts the reliability calculation of the terminals. In Section III two heuristic design techniques for

network topology modification have been provided with illustrative examples. Section IV presents a comparison of the proposed techniques with the existing method along with the observations derived from the numerical results. Finally, conclusions and further discussions are given in Section V.

2. RELIABILITY AND ITS ESTIMATION

Networks are modeled by directed graphs (digraphs) in which the communicating entities are vertices (V) or "nodes" and the communication paths are edges (E) or "links". Network links are assumed to be failure-prone components whereas nodes are assumed to be failure-free. Link failures are mutually statistically independent. The links are unidirectional. Bidirectional communication is supported through a parallel link in the opposite direction. The network is free from self loops i.e. links originating from one node and destined for the same node is not considered as the nodes are failure free. The reliability analysis approach opted here is that of terminal-pair / s-t reliability, which is defined as the probability of successful communication between any specified pair of vertices in a network, given the probability of success for each communication link in the network. A network can be represented by its adjacency matrix, $G = [g_E]$, where

$$g_E \begin{cases} 1, \text{ if there exists an edge E} \\ 0, \text{ otherwise} \end{cases}$$

An elementary event is defined as a particular realization of the network, i.e., it is the specification of the success or failure of each edge. If an edge E succeeds then it is represented as '1' and if it fails then it is represented as '−1'. A '0' indicates both '1' and '−1'. An s-t path is a path from source to sink. A shortest s-t path is an s-t path with the least number of edges. An event is a success if the graph realization corresponding to every elementary event constituting that event has at least one s-t path. Otherwise, the event is a failure. If an event ξ has m elements, $\xi \equiv \{1\ 2\ 3\ \ldots\ldots\ m\}$, then the compliment event of ξ can be given as:

$$\bar{\xi} = \{\bar{1}\} \cup \{1\ \bar{2}\} \cup \{1\ 2\ \bar{3}\} \cup \ldots\ldots \cup \{1\ 2\ 3\ \ldots\ldots\ \bar{m}\}$$

This follows from De Morgan's laws and the absorption law. Let S be an exhaustive set of disjoint success events, such that the graph realization corresponding to an event contains an s-t path. Let S_i be success event i in the set S. Terminal-pair reliability is the sum of the probability of each event in the set S.

$$R = \sum_{i=1}^{|S|} Pr\{S_i\} \qquad (1)$$

The algorithm for reliability calculation is given below.

Step 1:   Input network in the form of its adjacency matrix.

Step 2:   Initialize R = 0.0 and ξ = [0 0 …. 0]

Step 3:   Create the graph corresponding to ξ.

Step 4:   Find the shortest s-t path, P. If P exists, then success event is $\xi_1 = \xi/P$.

Step 5:   Compute the probability of ξ_1, $R \leftarrow R + Pr(\xi_1)$.

Step 6:   Obtain the complement events of ξ_1 and repeat from Step 3 till all the success events are evaluated.

The above algorithm is an exact measure for estimating reliability, which has $O(2^m)$ time complexity. Another approximate method for reliability calculation is explained in [1, 2], which has linear time complexity when an upper bound is placed upon the indegree of all nodes. To understand the algorithm the following definitions are required.

Root Node: The root node of a network graph is that node from which the s-t reliability is to be computed.

Source Node: A source node S of a given network vertex V is a node that has at least one link, from itself, incident into vertex V.

Simple Path: A series combination of links by which a given vertex i may be reached from another vertex j such that no link is traversed more than once.

Operational Source Node: A source node is termed operational if it is connected to the root node by at least one simple path.

R(V): The reliability of network vertex V, that is, the probability that vertex V is connected to vertex 1 by at least one simple path.

l_i: The event that link i is operational.

S_i: The event that source node i of vertex V is connected to the root node by at least one simple path.

The algorithm is as followed.

Step 1:   Input network topological description file.

Step 2:   Initialize i = root node.

Step 3:   Identify all s-dependent source nodes of an arbitrary node, j.

Step 4:   Initialize $R(i,j) \leftarrow 0$ and $\{C(i,j)\} \leftarrow \{0\}$.

Step 5:   For n = 1 to number of source nodes of node j, do Step 6.

Step 6:   m ← identifier of source node n of node j.

   if m is an s-independent source node of node j

   calculate

   $R_n(i,j) \leftarrow R_{n-1}(i,j) + Pr\{l_n\}R_n(i,m) -$

   $Pr\{l_n\}Pr\{S_n|T_{n-1}(i,j)\}R_{n-1}(i,j)$

   $\{C(i,j)\} \leftarrow \{C(i,j)\} + \{m\}$

   Repeat the process until convergence criteria is satisfied.

Many other procedures for estimating reliability are given in the literature. Any s-t reliability approach is applicable for the techniques developed in this paper.

3. PROPOSED TECHNIQUES FOR INCREASING THE RELIABILITY

The objective is to increase the reliability (R) of any node that has the lowest reliability value, say R_{min}. This can be achieved by placing new links between the nodes so that R_{min} is increased as large as possible. Let us consider a network with nodes Vi, where i=1,2,………N. N is the total number of nodes. Thus, for N nodes the number of reliability calculation required is {N × (N-1)}. Out of these one node pair must have a reliability value that is minimum. Say this occurs for node pair (Vi_m,Vj_m). In this case,

$$R_{min} = R (Vi_m,Vj_m)$$
(2)

The source node of this pair is Vi_m. Placing a new link from node Vi_m to some node Vz will result in an increase of R_{min} to R_{new} as well as reliabilities of all other pairs will be increased. Thus,

$$\Delta R = R_{new} - R_{min} \qquad (3)$$

To get the maximum improvement in the reliability of the worst node pair (Vi_m,Vj_m), node Vz must be chosen so as to maximize equation (3).

This heuristic topological design procedure is useful particularly for existing communication networks. Whenever any network experiences degradation in its performance, the network's reliability becomes abruptly low. In most of the cases a complete redesign and reinstallation of the system is not possible due to many reasons. In such cases, the installation of additional links is the only way to improve network reliability. Assuming a fixed cost per communication link and that all the links have same channel capacity, these heuristic topological modification design algorithms will yield the most reliable network topology for a given number of additional links, or equivalently additional cost.

3.1 Proposed Method I

For an arbitrary digraph G(V,E) with N nodes, a set of paths for any pair of nodes can be yielded. That is,

$$P_{i,j} = path(Vi,Vj), i,j = 1,2,…..,N, j \neq i \qquad (4)$$

Thus, for any node, say, Vi, paths can be calculated from this source node to all other nodes Vj, j=1,2,………….,N, j≠i. That is, all the paths between every pair can be obtained. Each node will have (N-1) pairs for which paths are determined. Among these (N-1) pairs at least one pair must have the number of paths which is the minimum. The reliability of this node pair is then calculated. If more than one such path exists then, select any one of them. In this way N numbers of reliabilities are obtained. Comparing these N numbers of reliabilities the least one can be found. The source node of this pair is the weakest node of the concerned network. Similarly the source node of the pair which has the maximum number of paths is the most reliable node. As suggested by equation (3), a link is then placed from the least reliable node to the most reliable node. This process is repeated until either a sufficient reliability has been achieved or the allowed numbers of additional links have been added. The algorithm is given below.

Algorithm (*Method I*)

Input network in the form of its adjacency matrix, G = [g_E]

Initialization

> *link = 0;*

> *i = 0;*

Iteration

> *while (link < max_number_of_extra_links) do*

>> *while (i < number_of_nodes) do*

>>> *calculate $P_{i,j}$ starting at node Vi, j=1,2,….,number_of_nodes, j≠i;*

>>> *find the pair having minimum number of paths (Vi,Vj) = min($P_{i,j}$);*

>>> *compute the reliability, R(i,j) of (Vi,Vj) by*

>>> *s-t reliability approach;*

>>> *i ← i+1;*

>> *end do;*

>> *find the minimum reliability*

>> *R_{min} = min{ R(i,j) };*

>> *find the least reliable node*

>> *V_{least} = source node of R_{min};*

>> *find the pair having maximum number of paths, P_{max};*

>> *find the most reliable node*

>> *V_{best} = source node of P_{max};*

>> *repeat until (no link from V_{least} to V_{best})*

>>> *if (no link from V_{least} to V_{best})*

>>>> *add a link from V_{least} to V_{best};*

>>> *else*

>>>> *find node which has next highest number of paths;*

>>> *end if;*

>> *end repeat;*

>> *link ← link+1;*

> *end do;*

Output: Modified network with additional links and increased reliabilities

3.1.1 Numerical Results

The algorithm in the previous section is coded in FORTRAN 77 and the simulation is done on a Xeon processor, 3.4 GHz PC with 4 GB RAM. In order to test the performance of the method, a series of computational experiments were performed using this algorithm. Owing to limited space of writing only one simulation example is presented. We have used Network A as depicted in Figure 1 in our experimentation, which was introduced by Belovich in [1]. The 11-node network is lightly connected with a

uniform link reliability of 0.9. The topological design modification algorithm directs the placement of new links so as to increase the reliability of Network A. An example of the first iteration is given below.

Consider node 2 of network A. It has 10 pairs that are $(2-1)$, $(2-3)$, $(2-4)$……... $(2-11)$. For each of the pair the total numbers of paths are found which are enlisted below in Table 1.

It is clear that the pair $(2-5)$ has minimum number of paths from all other pairs. The reliability of this pair is 0.900000. Thus, for node 2, reliability is calculated only one time. Likewise the reliabilities of all other nodes are computed, shown in Table 2.
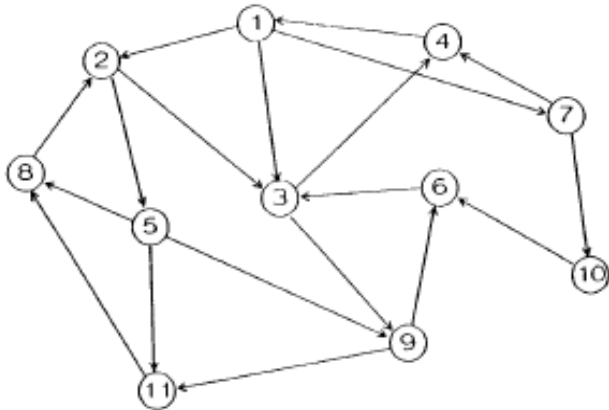

Figure 1   The topology structure of 11 node Network A

TABLE 1.   NUMBER OF PATHS FOR NODE 2

Node pair (source-terminal)	No. of paths
$(2-1)$	2
$(2-3)$	2
$(2-4)$	2
$(2-5)$	1
$(2-6)$	3
$(2-7)$	2
$(2-8)$	4
$(2-9)$	2
$(2-10)$	2
$(2-11)$	3

TABLE 2.   MINIMUM RELIABILITIES OF 11 NODES

Node	Reliability
1	0.810000
2	0.900000
3	0.656100
4	0.729000
5	0.620317
6	0.590490
7	0.900000
8	0.810000
9	0.800120
10	0.656100
11	0.729000

From the data of Table 2, it is quite obvious that node 6 has the minimum reliability. The pair having the maximum number of paths is $(7-8)$. Thus the least and the most reliable nodes are obtained as 6 and 7. A link is then placed from node 6 to node 7. Thus, Network A receives the first link which has been labelled as "1", is shown in Figure 2.
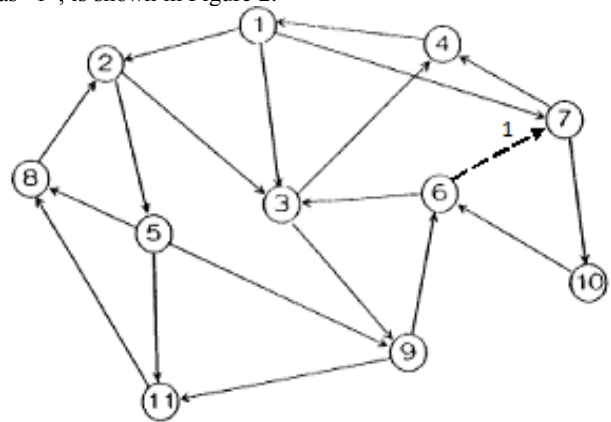

Figure2   The topology structure of Network A after 1st link

The algorithm is run for 10 iterations, i.e. 10 new links are added to Network A for increasing reliability. Figure 3 illustrates the result. The links placed to Network A are labelled in numerical order. Label "1" indicates the 1st link added to the network, the 2nd link is labelled as "2", and so on.
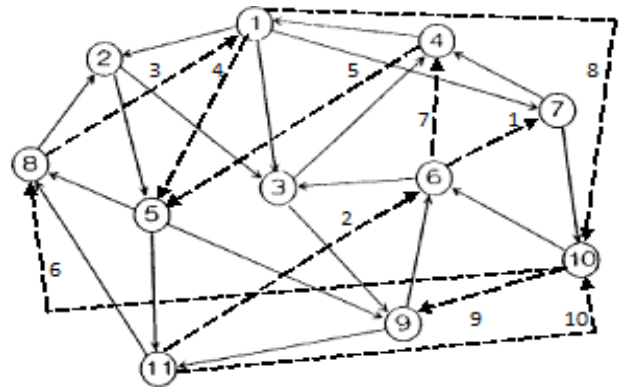

Figure 3   The topology structure of Network A after 10 links are added

The efficiency of the least reliable node increases as each link is being added by the design algorithm. This is true except for the case of link "6". The reason is as follows. In this design algorithm the terminal-pair reliability is being calculated only for the pair having minimum number of paths from rest of the pairs. But the reliability of any node pair is also dependent on the number of hops that the paths contain. It is quite obvious that as the path gets longer, thereby raising its hop count, the more is the chance that the traffic will not reach the destination. This hidden parameter is not considered in this algorithm. In spite of this fact the algorithm yields the ultimate goal of improving the

performance by increasing reliabilities of the network nodes. This approach owes its efficiency due to its ability to conclude in a very faster time, thereby reducing the time complexity. Whereas all the enumerative-based topology design algorithms calculate reliabilities of every node pair for finding the worst node and hence the computational complexity is much higher.

Table 3 depicts the improvement in the reliabilities of the worst nodes. The improvement is most significant for the first four links added. As additional links are being placed the improvement is not so dramatic, since the node reliabilities should asymptotically approach unity.

Table 3. Percentage Increase after Each Link Is Added

Number of links added	Link	Minimum Reliability	Percentage Increase
0	—	0.590490	—
1	6 - 7	0.729000	13.8
2	11 - 6	0.810000	21.9
3	8 - 1	0.867018	27.6
4	1 - 5	0.900000	30.9
5	4 - 5	0.900000	30.9
6	10 - 8	0.879519	28.9
7	6 - 4	0.889843	29.9
8	1 - 10	0.977738	38.7
9	10 - 9	0.978223	38.8
10	11 - 10	0.980801	39.0

Figure 4 illustrates the graph of the above data along with a trendline. A trendline is a function that shows a trend in a series of points, i.e. a trendline is the function that fits a point series best. When the trendline is added, the correlation coefficient R^2 is shown in the comment. The value of R^2 has come out to be 0.7901. The closer R^2 is to 1 the closer the trendline is to the points.
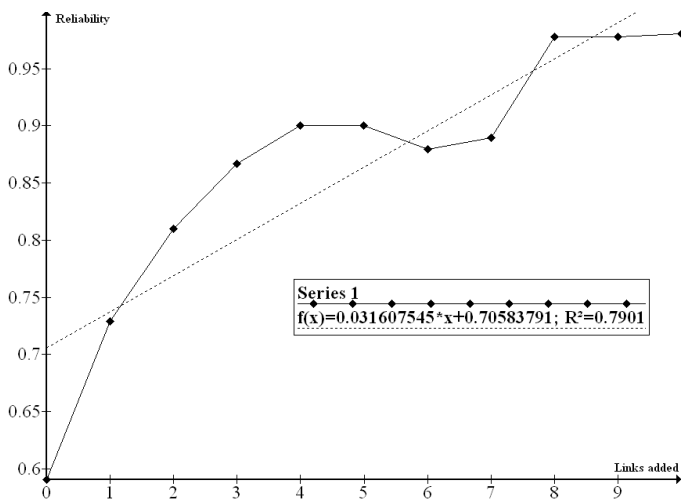

Figure 4   Reliability improvement of Network A

3.1.2   Time Complexity of Method I

The time complexity depends on the complexity of the algorithm used to calculate the reliability of individual nodes of the given network, say $R_{analysis}$ and to insert the links in the minimal reliability node-pairs. Since, the linear time complexity for $R_{analysis}$ has been used and it has been called in total *N* number of times for *N* nodes. Thus, the time complexity of Method I is given by:

$$Time\ Complexity = N * R_{analysis}$$
(5)

Therefore, Method I has linear time complexity.

3.2 Proposed Method II

As stated in Method I that the reliability of a pair of nodes is also dependent on the number of hops the paths have between them, is not taken into account in the former algorithm. With the increase in network size the hop count also increases. This affects the communications between any pair of nodes. Thus, the probability, that traffic reaching the destination is in an erroneous state or is lost in between and never reaches the target node, will also increase with the increment of hop count. Method II described in this section considers the number of hops along with the number of paths between the nodes. By doing so, the weakest or the least reliable node can be obtained more accurately.

Considering an N node network, for each node there are (N-1) pairs for which paths are determined. In each of these pairs, the number of hops is calculated for each path. Let there be m number of paths for the pair (Vi,Vj). The number of hops for each path is:

Number of hops for path 1: h_1(path 1)
Number of hops for path 2: h_2(path 2)
.
.
Number of hops for path m: h_m(path m)

Summing over all the number of hops will yield the total hop count for the node pair (Vi,Vj). That is,

$$H_{i,j} = h_1(path\ 1) + h_2(path\ 2) + \ldots\ldots + h_m(path\ m)$$

i.e. $$H_{i,j} = \sum_{k=1}^{m} h_k$$
(6)

Now, as the number of paths increases reliability of that pair also increases. But with the increase in the number of hops the reliability of that pair decreases. Thus, the total number of hops is divided by the number of paths to obtain the average, and the integer value is considered. That is,

$$A(i,j) = H_{i,j} / P_{i,j}$$
(7)

Likewise, the value of A, i.e. the average can be determined for every pair. The reliability is calculated only for that pair which has maximum value of A. To get more accurate result reliabilities of the pairs having the second larger values of A is also computed and the minimum of them is considered as the

weakest pair for that node. In this manner for N nodes N numbers of pairs are obtained along with their reliabilities. From these N node pairs the pair with the least reliability is found whose source node is considered as the weakest node of the network. Similar to Method I the source node of the pair which has the maximum number of paths is taken as the most reliable node. According to equation (3), a link is then placed from the least reliable node to the most reliable node and the process is repeated till the allowed number of links are added or the reliability has reached to a satisfactory level. The algorithm is given below.

Algorithm (Method II)

Input network in the form of its adjacency matrix, $G = [g_E]$

Initialization

   link = 0;

   i = 0;

Iteration

   while (link < max_number_of_extra_links) do

      while (i < number_of_nodes) do

         calculate $P_{i,j}$ starting at node Vi, j=1,2,....,number_of_nodes, j≠i;

         calculate $H_{i,j}$ starting at node Vi, j=1,2,....,number_of_nodes, j≠i;

         $A(i,j) = H_{i,j} / P_{i,j}$;

         find the pairs having maximum and next maximum values of A

         (Vi,Vj) = max{A(i,j)};

         compute the reliabilities, R(i,j) of all (Vi,Vj) by s-t reliability approach;

         find the minimum reliability;

         i ← i+1;

      end do;

      find the minimum reliability

      $R_{min} = min\{ R(i,j) \}$;

      find the least reliable node

      V_{least} = source node of R_{min};

      find the pair having maximum number of paths, P_{max};

      find the most reliable node

      V_{best} = source node of P_{max};

      repeat until (no link from V_{least} to V_{best})

         if (no link from V_{least} to V_{best})

            add a link from V_{least} to V_{best};

         else

            find node which has next highest number of paths;

         end if;

         end repeat;

      link ← link+1;

   end do;

Output: Modified network with additional links and increased reliabilities

3.2.1 Numerical Results

The algorithm in the previous section is also coded in FORTRAN 77, the simulation is done on a Xeon processor, 3.4 GHz PC with 4 GB RAM and is applied to the same set of networks. Network A of Figure 1 is given as an example. Considering the same node, i.e. node 2 as in the example of method I. Starting from iteration 1, the total hop count for every pair of node 2 is divided by the number of paths of that pair. The result is shown in Table 4.

TABLE 4. HOP COUNT DIVIDED BY NUMBER OF PATHS FOR NODE 2

Node pair (source-terminal)	A = (Hop count / No. of paths)
(2 − 1)	4
(2 − 3)	2
(2 − 4)	3
(2 − 5)	1
(2 − 6)	4
(2 − 7)	5
(2 − 8)	3
(2 − 9)	2
(2 − 10)	6
(2 − 11)	3

The pair (2 − 10) has the maximum value. Thus, the reliability of this pair is obtained, which has come out to be 0.633537. In this manner all the node reliabilities are computed. Table 5 depicts them.

Table 5. Minimum Reliabilities of 11 Nodes

Node	Reliability
1	0.810000
2	0.633537
3	0.656100
4	0.729000
5	0.620317
6	0.590490
7	0.800871
8	0.570183
9	0.613230
10	0.590490
11	0.513165

Clearly, the data shows that node 11 has the minimum reliability in comparison to other nodes. Hence, it is considered as the least reliable node. The pair having the maximum number of paths is (7 − 8), giving node 7 as the most reliable node for iteration 1. According to the proposed algorithm a link is placed from node

11 to node 7. This is shown in Figure 5, where the first link added has been labelled as "1". The algorithm is run for 10 iterations and the link placement is shown in Figure 6.
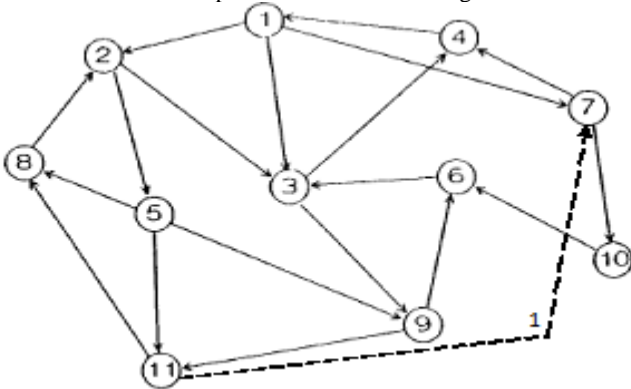

Figure 5   The topology structure of Network A after 1st link

The link placement by this algorithm is more balanced than the previous one as the process of finding the weakest node is more accurate. This is evident from the data depicted in Table 6. The total percentage increase for Method II is 46.3% which is much higher than that of Method I being 39.0%.
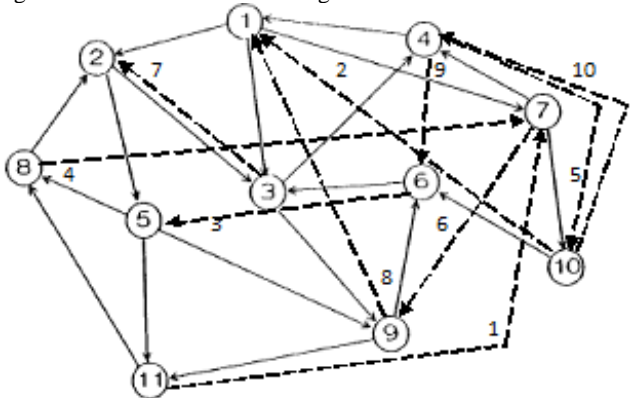

Figure 6   The topology structure of Network A after 10 links are added

Table 6.   Percentage Increase after Each Link Is Added

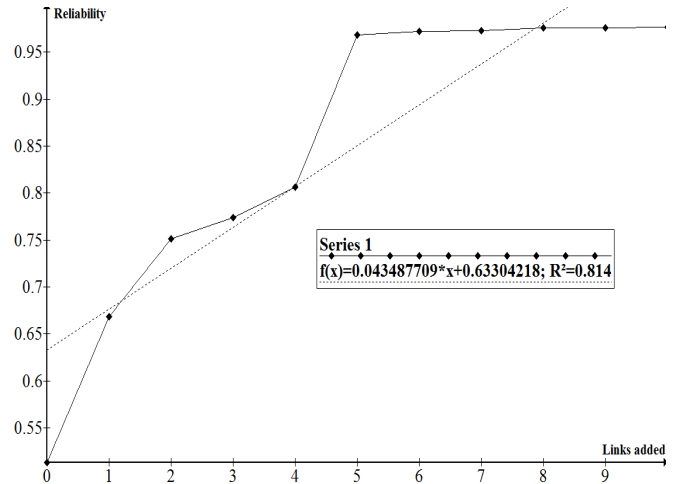| Number of links added	Link	Minimum Reliability	Percentage Increase
0	—	0.513165	—
1	11 - 7	0.668420	15.5
2	10 - 1	0.751704	23.8
3	6 - 5	0.773874	26.0
4	8 - 7	0.806398	29.3
5	4 - 10	0.968499	45.5
6	7 - 9	0.971868	45.8
7	3 - 2	0.973180	46.0
8	9 - 1	0.975674	46.2
9	4 - 6	0.976169	46.3
10	10 - 4	0.976437	46.3

Figure 7   Reliability improvement of Network A

This improvement is also reflected when the graph of the above data is plotted. In Figure 7 the trendline is shown with the correlation coefficient, R^2 being 0.814.

3.2.2 Time Complexity of Method II

The average value of A may be same for more than one node-pair; hence, for each node $R_{analysis}$ function may be called multiple times. But on an average it has been seen that two times is sufficient. Thus, for N nodes $R_{analysis}$ is called $2N$ times. Thus, the time complexity of Method II is given by:

$$Time\ Complexity = 2N * R_{analysis} \qquad (8)$$

Therefore, Method II has linear time complexity.

Each method has some advantages of its own. Method I incorporates very simple calculations and quickly reaches to the target, whereas, Method II involves more operations and provides balanced interconnection networks.

3.3 Comparison of the proposed techniques with an existing method (Belovich [1])

In this sub-section, the proposed schemes have been compared with the technique proposed in [1], the outcome of which is given in Table 7.

Table 7.   Comparison with the Existing Method

Proposed Methods	Existing Methods
The percentage increase for the first four links is 29.3%.	The percentage increase for the first four links is 8.3%.
Nodes 4 and 11 have three links, two incoming and one outgoing. Node 4 receives three additional links and node 11 receives one.	None of the nodes 4 and 11 receive extra link.
Linear time complexity.	Quadratic time complexity.

24

The least reliable node is determined by considering the weakest pair. The first link is added from node 11 to node 7, because node pair (11-10) has the minimum reliability.	The least reliable node is determined by considering the sum of reliabilities of all node pairs. Thus, the weakest node may not be found (first link is added between 8 and 7)

Finally, the following three key features of the proposed techniques may be noted:

- The time complexity is linear since for all iterations reliability is calculated only once.
- Additions of links are balanced, i.e. nodes having less number of indegree or outdegree receives the links.
- Exact reflection of minimum node reliability is achieved.

4. CONCLUSIONS

Two linear time algorithms for designing highly reliable communication networks have been presented, which use a reliability estimation method to determine the reliability between all possible node-pair of the given networks. The node-pairs having lower reliability are then identified, where additional links are inserted as a whole to increase the network reliability. The algorithms terminate when either a maximum number of additional links are added or a desired level of reliability is achieved. The proposed techniques are compared with other techniques and satisfactory results have been found.

5. REFERENCES

[1] S.G. Belovich, "A Design Technique for Reliable Networks under a Non-Uniform Traffic Distribution", IEEE Transaction on Reliability, Vol. 44, No. 3, pp 377-386, 1995.

[2] S.G. Belovich, V.K. Konangi, "A Linear-time Approximation for Computing the Reliability of a Network", Computer Networks and ISDN Systems, Vol. 21, pp 121-127, 1991.

[3] W.P. Dotson, J.O. Gobien, "A New Analysis Technique for Probabilistic Graphs", IEEE Transactions on Circuits & Systems, Vol. CAS-26, No. 10, pp 855-865, 1979.

[4] Y.B. Yoo, N. Deo, "A Comparison of Algorithms for Terminal-Pair Reliability", IEEE Transaction on Reliability, Vol. 37, No. 2, pp 210-215, 1988.

[5] A.R. Majeed, J.A. Hussein, "Weighted Network Reliability and Modeling", IEEE 5th IMCSSD, 2008.

[6] B. Liu, I. Iwamura, "Topological Optimization Models for Communication Network with Multiple Reliability Goals", Elsevier Science Ltd., Computers & Mathematics with Applications, Vol. 39, Issues 7-8, pp 59-69, 2000.

[7] D.L. Darren, A.E. Smith, "Heuristic Optimization of Network Design Considering All-Terminal Reliability",
IEEE Transaction on Reliability and Maintainability Symposium. Annual Proceedings, pp 194-199, 1997.

[8] Rong-Hong Jan, "Design of Reliable Networks", Computers and Operations Research, Vol. 20, No. 1, pp 25-34, 1993.

[9] Shuichi Shinmori, Takeshi Koide and Hiroaki Ishii, "On Lower Bound for Network Reliability by Edge-Packing", Transactions of the Japan Society for Industrial and Applied Mathematics, Vol. 5, No. 2, pp 139-151, 1995.

[10] T. Politof, A. Satyanarayana, L. Tung, "An O(n.log(n)) Algorithm to Compute the All-Terminal Reliability of $(K_5, K_{2,2,2})$ Free Networks", IEEE Transaction on Reliability, Vol. 41, No. 4, pp 512-517, 1992.

[11] M.P. Bailey, V.G. Kulkarni, "A Recursive Algorithm for Computing Exact Reliability Measures", IEEE Transaction on Reliability, Vol R-35, pp 36-40, 1986.

[12] O.J. Wasem, "Optimal Topologies for Survivable Fiber Optic Networks Using SONET Self-Healing Rings", Proc. Globecom '91, pp 2032-2038, 1991.

[13] O.J. Wasem, T. Wu, R. Cardwell, "Survivable SONET Networks -Design Methodology", IEEE J. Selected Areas in Communications, Vol. 12, pp 205-212, 1994.

[14] P.T.T. Binh, T.Q.D. Khoa, "Application of Fuzzy Markov in Calculating Reliability of Power Systems", IEEE Transmission & Distribution Conference and Exposition, pp 1-4, 2006.

[15] A. Kumar, R. M. Pathak and Y.P. Gupta, "Genetic-Algorithm Based Reliability Optimization for Computer Network Expansion", IEEE Transaction on Reliability, Vol. 44, No. 1 pp 63-72, 1995.

[16] A. Kumar, P. P. Parida, and M. Gupta," Topological Design of Communication Networks using Multi Objective Genetic Optimization", Evolutionary Computation, Vol.1,pp 425-430, 2002.

[17] B. Dengiz, F. Altiparmak and A.E. Smith, "Efficient Optimization of All-Terminal Reliable Networks, Using an Evolutionary Approach", IEEE Transaction on Reliability, Vol. 46, No. 1 pp 18-26, 1997.

[18] D.L. Deeter, A.E. Smith, "Heuristic Optimization of Network Design Considering All-Terminal Reliability", IEEE Transaction on Reliability and Maintainability Symposium, pp 194-199, 1997.

[19] I. Gertsbakh , Y. Shpungin , "Network Reliability Importance Measures: Combinatorics and Monte Carlo Based Computations", WSEAS Transactions on Computers, Vol. 7 , Issue 4, pp 216-227, 2008.

[20] H. Boudali, J.B. Duqan, "A Continuous-time Bayesian Network Reliability Modeling and Analysis Framework", IEEE Transaction on Reliability, Vol. 55, No. 1, pp 86-97, 2006.