Enumeration of Global path for evaluating Global Reliability of Computer Communucation Network

Dr. Mohd Ashraf
Maulana Azad National Urdu University, Department of Computer Science & Engg, Polytechnic Hyderabad, India

Abstract: The global reliability of computer communication network is the probability that any given pair of node, there exit a path for reliable data communication. Reliability plays a key role in the performance of any large-scale Computer Communication application. Computer Communication network reliability must consider several design factors, viz. coverage, connectivity, lifetime, etc. However, connectivity remains the most fundamental factor especially in a large scale harsh environment. It is based on finding minimum paths or spanning trees, need a less memory and computational time. In this paper, Authors propose and discover the difficulty of enumeration of these minimum paths that can be suitable method for further evaluation of global reliability for Communication Network.

Keywords: *communication Network, Global Path, Reliability Graph Spanning trees enumeration, global reliability evaluation*.

Introduction

The performance of any communication network depends upon the probability of successful data transmission or in other words performance of the network is defined as the communicability of the nodes. It depends upon many factors in which reliability play an important role. It is a very important parameter also in the operation of equipments, networks, system design and maintenance of the systems. The reliable communication between the users is the prime concern of the modern communication scenario. There is also a requirement of reliable connectivity between the nodes as well as from node to agent node or base station to avail the centralized network services.

Wilkov [3] first suggested the evaluation of overall network reliability by using the concepts of terminal pair reliability, i.e.,by finding all possible paths between each of the n (n-1)/2 node pairs. Fratta and Montanari [4] also proposed an approximate method based on decomposition technique for all-terminal reliability assessment, but these methods are impracticable for real time communication sensor networks. Considering links failure probabilities and using an adjacency/connection matrix, Jain and Gopal [5] proposed a method for evaluation of all-terminal reliability for complex systems. Hardy et. al. [6] suggested a binary decision diagram based approach for reliability evaluation. Using spanning trees and SDP concept with disjoint grouping approach is one of the

good ways of evaluating reliability measures can be found in Chaturvedi and Misra [7]. AboElFotoh et. al. [8] performed a factoring algorithm for Computer Communication Network based on algebraic connectivity graph. Like other factoring algorithms, it essentially has exponential complexity. Too many redundant computations from isomorphic sub-networks make this algorithm less efficient.

Reference [9, 10] gives different taxonomy to classify sensor networks according to communication functions, data delivery models, and network dynamics. It proposes that communication within WSN can be conceptually classified into two categories: application and infrastructure. Application communication relates to the transfer of sensed data about the phenomenon whereas infrastructure communication relates to the delivery of configuration and maintenance data which includes the several design factors, viz. coverage, connectivity, lifetime, etc. In this paper, we present a connectivity based reliability measures for networks i.e. an initial phase of infrastructure communication is needed to set up the network. Furthermore, if the sensors are energy-constrained, there will be additional communication for reconfiguration.

Problem Statement

In this paper, we consider the problem of modeling, enumeration of spanning trees and evaluating the connectivity oriented reliability of Computer Communication Network. We define the Computer Communication Network reliability as the probability that exist an operating communication path between the sink node, Due to unique features of Computer Communication Network, the reliability evaluation of communication faces a combination of challenges that are not common to the traditional networks. Several factors contribute to the difficulty of Computer Communication Network reliability measures and evaluation;

- The complexity of the reliability evaluation increases sharply as the number of nodes increases to the order of hundreds of nodes [11]. Therefore traditional approaches cannot directly apply to Computer Communication Network.
- There are different approaches suggested in the literature to solve reliability evaluation problems involving exact analytical calculation, lower and upper bound construction, and simulation.

To perform exact analytical reliability analysis of such Computer Communication Network in order to make the problem of computing probabilistic connectedness there are following major assumptions in our analysis:

- The first assumption is related with statistical independence of edge failures. The assumption, edge failures are statistically independent, implies that the probability of a link being operational is not dependent on the states of the other links in the network. The inherent assumption here is that the link failures are caused by random events which affect all links individually.

- The second assumption is that the perfectly reliable nodes, i.e., their probability of failure is almost zero or insignificant. Much of the development in the area of network reliability measurement is presented under the assumption of perfectly reliable nodes

Methodology and Assumption

Figure 1 depicts the proposed methodology has been adopted for applying the SDP techniques for *all-terminal/global* reliability evaluation. Clearly, the key issues in applying the approach are network representation, enumeration of all possibilities (spanning trees) for all-nodes connectivity and by making these possibilities disjoint with each other to form the reliability expression.

Therefore, in the proposed approach, after enumerating the *all spanning trees*, the next step in the reliability evaluation is to obtain the symbolic expression in terms of the probability of the various components being operational/non-operational.

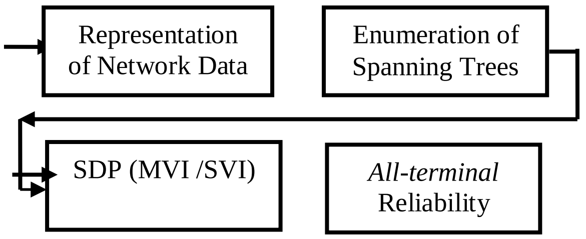


Figure 1: Block diagram of technique for

If the spanning trees are mutually exclusive, the probability of the union of ‘*n*’ events can be written as:

$$Pr(E_1 \cup E_2 \cup . . . \cup E_n) = Pr(E_1) + Pr(E_2) + . + Pr(E_n) \quad (1)$$

As the generated spanning trees are not mutually exclusive, it is desired to obtain the disjoint of the probabilistic expression of the spanning trees. Various Boolean algebra methods have been reported in the literature for disjointing the probabilistic terms so that the simple additive expression can be used for reliability estimation. One such method available in the literature as suggested by Chaturvedi and Misra [12], based on the principle of SDP has been employed in the present study. The method decomposes the set of *all spanning trees* into another set of mutually exclusive spanning trees, which has a one-to-one correspondence with the reliability expression. And we make the following assumptions in our analysis.

a) A communication network is modeled by probabilistic connected graph.
b) The nodes of the network are perfectly reliable.
c) The network and its branch have only two states **(a)** working or **(b)** failed.
d) The failure probability for each link or node is given as a fixed probability for a given mission time or in terms of a lifetime distribution.

Proposed Algorithm

The algorithm has been implemented in the JAVA. For reader’s benefit, each step of the algorithm has been illustrated with suitable example. The proposed algorithm has been implemented in two sections. First section of the algorithm defines the initialization part and rest of the section implies the connectedness of the network. Finally the Step# 8 provides all spanning tree for g-reliability evaluation.

A. Algorithm

Step # 1: [Initialize] Label all the node of communication network from 1 to N and all the link from 1 to M, where N is the number of node and M is the number of link between the nodes.

Step # 2: Represent the network with incidence matrix I.

Step #3: Generate all possible combination of link MCN-1 with N-1 link out of M link.

Step # 4: Repeat step #5 to step# 7 until the link combination list is empty.

Step # 5: Create the sub graph by taking the entire node (N) and add the N-1 link from reading link combination list.

Step # 6: Generate an adjacency matrix A corresponding to sub-graph.

Step# 7: Call Graph_connectivity (g (V, E), s), apply to the sub graph.

a) **for** all node v **do**

/* Check the status of all node in Sub graph*/

if (STATUS[v]=3) **then** Store the sub graph as Spanning tree and Go to **step# 4**.

b) **Else** reject this combination of link (Sub graph). Go to **step# 4.**

c) End of **step # 4** loop

Step# 8: Display all spanning tree generated in step #7 (a).

B. *Procedure graph_connectivity (g (V, E), s)*

The **graph_connectivity** is a sub-algorithm which is used to check the connectivity of a sub graph. This algorithm visit all the node of sub graph and store the visited status in data structure array, STATUS[ ]. If the **graph_connectivity ( )** visited all the node of the graph, Graph is connected otherwise graph is disconnected.

The general idea behind this algorithm beginning at a start node A is as follows. First we examine the starting node A. Then we examine each node v along a path P which begins at A; that is, we process a neighbor of A, then a neighbor of a neighbor of A, and so on. After coming to a “dead end” that is, to the end of path P, we backtrack on P until we can continue along another path P’ and so on. A field STATUS is used to tell us the current status of a node.

During the execution of our algorithms, each node v of G will be in one of three states, called the status of v, as follows:

STATUS [v_i] ←1: (**Ready state**) The initial state of the node N.

STATUS [v_i] ←2: (**Waiting state**) The node is in stack, waiting to be processed.

STATUS [v_i] ←3: (**Processed state.)** The Node v has been processed.

Step #1: Initialize all the node to ready state [STATUS[v]←1

Step #2: call Push(STACK, s) ;

/ push() used to insert the vertex on the top of the stack where STACK [1,-- n] be an array implementation of stack, s is the starting vertex/**

Set STATUS[s] ←2 *;*

Step # 3: While stack is not empty

Step # 4: call Pop(STACK, v)

/ Remove the top node of stack an become visited node N */*

set STATUS [v] ←3; /* visited node*/

Step #5: for each neighbor of processed node v

a) **if**(STATUS [next node]= 1)
*/*ready State of the node*/*
then call PUSH(STACK, v) ;
/ insert the adjacent node of N to the top of the stack*/*
Set STATUS[v] ←2 ; */*waiting State*/*

b) **If** (STATUS [next node]= 2)

Call Pop(STACK, top) ;
*/*delete the current node from the stack*/*
Set STATUS [v] ←3;
call PUSH(STACK, v) */*Insert the Adjacent node of v which have STATUS[v] ←1 */*

c) **If** STATUS[next node]= 3 processed state, ignore the vertex.

Step # 6: END graph connectivity

C. *Flow chart*

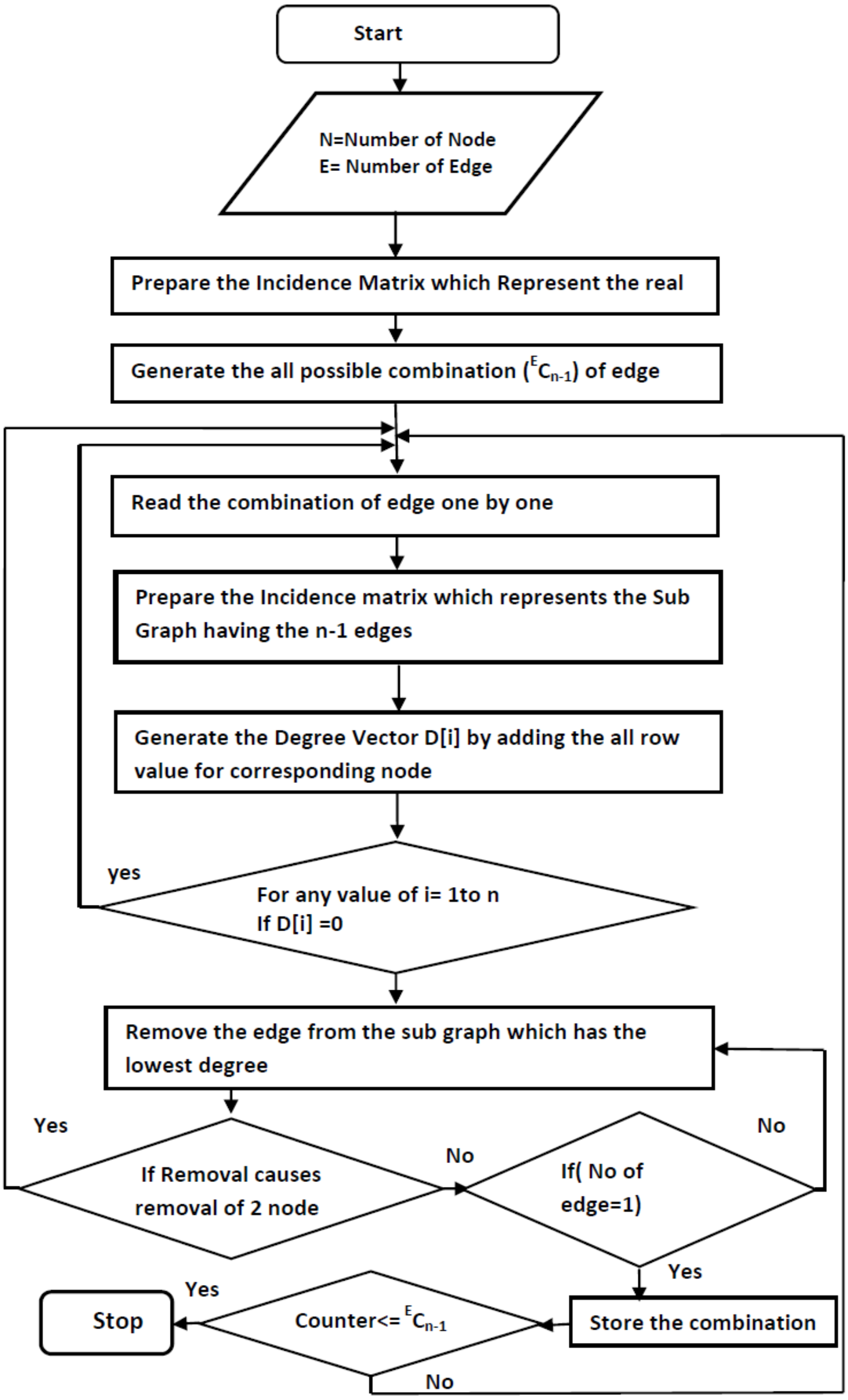
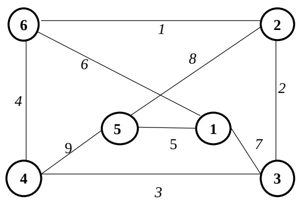


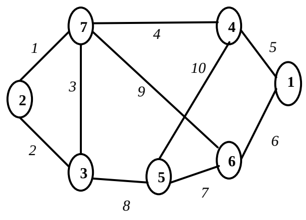
Figure 2: Flow chart of the proposed algorithm

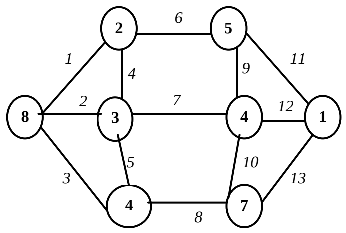
IJEE 0973-7383 Volume-8 • Number-1 Jan -June 2016 pp. 137-143 available online at www.csjournals.com

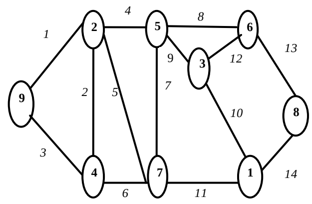
Experimental Result and Discussion

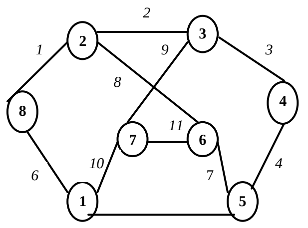
Authors have applied the proposed algorithm to several networks taken from the literature of varied complexities, and verified the reliability obtained by other researchers. Experimental results on several networks taken for this study from the published work. Among them, the results of comparison all spanning trees for few networks (shown in Figure 4 - Figure 9) [13, 14] are provided in Table 1 to show the efficacy of the algorithm and proposed framework to evaluate global reliability using connectivity criterion. Besides, Table 1 also provide the enumeration time for each network with reliability











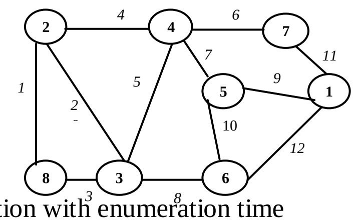


Table 1 Spanning trees enumeration with enumeration time

Size of graph	#spanning tree	CPU time in micro second	Density of Graph	Global reliability figure for p = 0.9
6N9L	81	4.8×10^{-3}	.6	0.993263229
7N10L	96	6.3×10^{-2}	.476	0.972218165
8N11L	168	9.5×10^{-2}	.392	0.969699135
8N12L	247	1.1×10^{-1}	.428	0.971153157
8N13L	576	1.8×10^{-1}	.464	0.991942810
9N14L	647	5.7×10^{-1}	.388	0.970104348
15N25L	111866	0370.93	.238	0.9965432
16N27L	580309	1599.975	.225	0.9800674
17N30L	2215933	11698.5	.220	0.9678342
20N25L	6128	2.57	.131	0.9864531

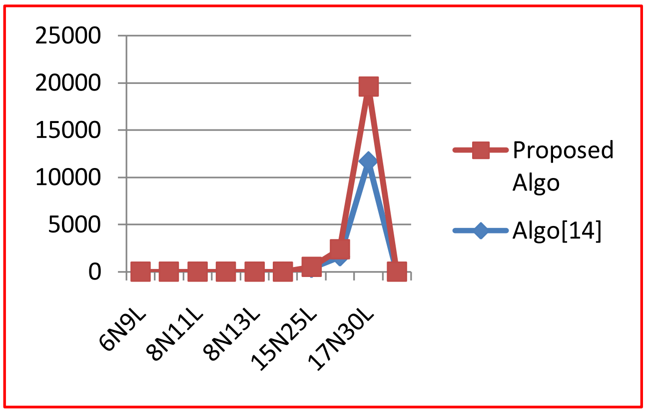


FIGURE 3: COMPUTATIONAL TIME IN MICRO SECOND FOR ALGO [14] AND PROPOSED ALGORITHM

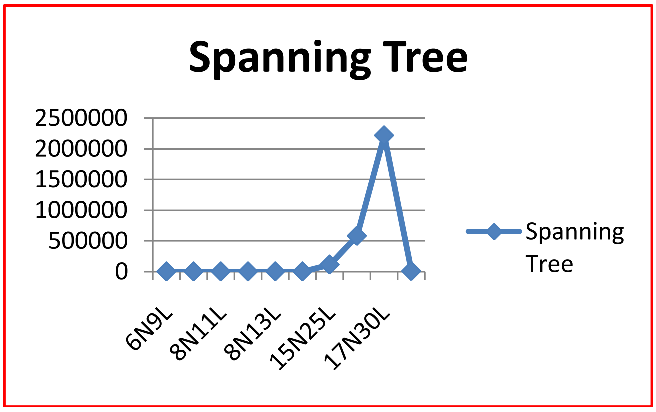


FIGURE 4: NUMBER OF SPANNING TREE VS NETWORK SIZE

Conclusion/Future Work

For a full utilization of facilities available in any system, a simple and efficient method for enumeration of all spanning trees has been proposed for evaluating the global reliability of communication networks. The proposed method is conceptually simple and computationally efficient for all-terminal reliability evaluation of large complex sensor networks and requires less computer memory and computational efforts as compared to other spanning trees based existing methods. Moreover, all methods of reliability computation are known to be computationally intractable or NP-hard, which make it difficult to compare the technique from the aspect of time or memory.

References

1. ain S. P. and K. Gopal, An Efficient Algorithm for Computing Global Reliability of a Network, IEEE Transactions on Reliability, Vol. 37, No. 5, pp. 488-492, **1988**.
2. Aggarwal A. and A. Satyanarayan, *An (O/E) time algorithm for computing the reliability of a class of directed networks*, Operation Research, Vol. 32, pp. 493-517, **1984**.
3. Wilkove R. S., *Analysis and design of reliable computer network*, IEEE Transactions on Communications. Vol. 20 (3), pp. 660-678, **1972**.
4. Provan J. S. and M. O. Ball, *Computing network reliability in time polynomial in the number of Cuts*, Operation Research, Vol. 32, pp 516-526, **1984**.
5. Denielson G., *On Finding the Simple Path Sets and Circuit in a Graph*, IEEE Transactions on Circuit Theory, Vol. CT-15, pp. 294-295, **1968**.
6. Fratta and U. G. Montanari, *A Vertex Elimination Algorithm for Enumerating All Simple Path Sets in a Graph*, Networks, Vol. 20, pp. 151-177, **1975**.

7. Aggarwal K. K., Gupta J.S. and K. B. Mishra, *A simple method for reliability evaluation of a communication system*, IEEE Trans. on Communications, pp. 563-566, **1975**.
8. Rai S. and K. K. Aggarwal, An efficient method for reliability evaluation of general network, IEEE Transactions on Reliability, Vol. R-273(3), pp. 206-356, **1978**.
9. Samad M. A., *An Efficient Method for Terminal and Multiterminal Path Set Enumeration*, Microelectronics and Reliability, Vol. 27, No. 3, pp. 443-446, **1987**.
10. Samad M. A., *Methods for Global Reliability Evaluation of any Large Complex System*, Reliability Engineering, Vol. 18, pp. 47-55, **1987.**
11. Aziz M. A., Sobhan M. A. and M. A. Samad, *Empirical Formulas and Global reliability Evaluation od Reliability Graphs*, Microelectronics & Reliability, Vol. 33, No. 9, pp. 1233-1236, **1993**.
12. Chaturvedi, S. K. and K. B. Misra, *A hybrid method to evaluate reliability of complex networks, International Journal of Quality and Reliability Management*, Vol. 19, No. 8, pp. 1098-1112, **2002**.
13. Rath D. and K. P. Soman, *A Simple Method for Generating k-Trees of a Network*, Microelectronics and Reliability, Vol. 33, No. 9, pp. 1241-1244, **1993**.
14. Chaturvedi S. K. and K. B. Misra, *An Efficient Multi-Variable Inversion Algorithm for Reliability Evaluation of Complex System using Path Sets*, International Journal of Reliability, Quality and Safety Engineering, Vol. 9, No. 3, pp. 237-259, **2002.**
15. Mishra R., Chaturvedi S. K., A Cutsets-based Unified Framework to Evaluate Network Reliability Measures, IEEE Trans. on Reliability, vol. 58-4, pp. 658-666,**2009**