

# COMPUTING NETWORK RELIABILITY WITH IMPERFECT NODES USING MODIFIED BINARY DECISION DIAGRAM

Manoj Singhal<sup>1</sup>, R. K. Chauhan<sup>2</sup>, Girish Sharma<sup>3</sup>

<sup>1</sup>Associate Professor, Department of Information Technology, Accurate Institute of Management & Technology, Greater Noida, (U.P.), India,

<sup>2</sup>Professor, Department of Computer Science and Applications, Kurukshetra University, Kurukshetra (Haryana), India,

<sup>3</sup> Associate Professor, Department of Computer Applications, Bhai Parmanand Institute of Business Studies, Delhi, India,

## **ABSTRACT**

*Since more and more enterprises become dependent upon computer communication network or networked computing applications particularly after the growth of local area network, wide area network and internet. Therefore failure of a single component may directly affect the functioning of the network. The nodes and the edges may fail in any real network. In this paper we have proposed an efficient algorithm to compute the reliability of a network (which has imperfect nodes) using modified binary decision diagram. After that we have compared our algorithm with an existing algorithm based on edge expansion diagram using ordered binary decision diagram. The size of the modified binary decision diagram obtained by our algorithm is much less than the size of the ordered binary decision diagram, however the reliability results are same.*

**Keywords:** *Binary Decision Diagrams (BDD), Directed Acyclic Graph (DAG), Computer communication Network (CCN), Modified Binary Decision Diagram (MBDD), Ordered Binary Decision Diagram (OBDD)*

## **I. INTRODUCTION**

Network reliability analysis receives considerable attention for the design, validation, and maintenance of many real world systems, such as computer, communication, or power networks. The components of a network are subject to random failures, as more and more enterprises become dependent upon computer communication network (CCN) or networked computing applications. Failure of a single component may directly affect the functioning of a network. So the probability of each component of a CCN is a crucial consideration while considering the reliability of a network. Hence the reliability consideration is an important factor in CCN [1]. The IEEE 90 standard defines the reliability as "*the ability of a system or component to perform its required functions under stated conditions for a specified period of time.*" Many algorithms have been presented to solve the problem of network reliability. These algorithms are based on the exact methods as well as the approximation methods [2, 3, 4]. Some of them are based on the min-paths/min-cuts methods. In these methods we first enumerate all the min-paths and min-cuts of the given CCN, then these min-paths/min-cuts are manipulated to get their counterpart in the sum of disjoint product form. Min-cuts methods have been used since 1960 to compute the reliability of a network [5]. The authors [6] have shown that min-cuts based algorithms are more efficient than the min-paths based algorithms only for the networks where number of min-cuts are less than the number of min-paths. However the number of min-paths/min-cuts increases exponentially as the size of the networks increases. It is impractical to enumerate all the min-paths/min-cuts of a very large network. Some of the others are based on the factoring theorem [7, 8]. Moskowitz was the first to use the factoring theorem directly to compute the reliability of a

network [9]. The factoring theorem divides the reliability problems into two sub problems and the formula is given below:

$$R(G) = P_e \cdot R(G/\text{edge } e \text{ functions}) + (1 - p_e) \cdot R(G/\text{edge } e \text{ fail}) \quad (1)$$

This factoring formula must be applied only when there is no reduction on the graph is possible. It has shown that the optimal binary structure of the factoring algorithm for undirected networks can be generated by means of pivoting. Before applying the factoring, we must apply the reduction techniques like polygon to chain or series-parallel [10]. If a network has imperfect nodes as well as imperfect links, then such type of structure will increase the complexity to compute the reliability of the network. The most commonly used method for nodes failure is incident edge substitution [11]. In this method, for an edge  $e_i$ , we put  $v_i e_i v_j$  in the min-path function for the perfect nodes, because if we consider an edge  $e_i$ , then this edge must contains two end vertices say  $v_i$  and  $v_j$ . So we have to simplify the Boolean function. The min-path function for the perfect nodes is the union of all min-paths from source to sink. By performing such type of operations we need large memory. One more feasible solution is to slightly change the probability function used in the factoring theorem and factor on links that have at least one end point [12]. The authors [13] have shown an efficient and exact method to compute the reliability of a network with imperfect vertices. One other method was shown by Xing with imperfect coverage and common cause failure [14].

One other author has tried to convert an undirected network in to a directed network and then compute the reliability of a network. This algorithm generates result with minor errors within reasonable time. This algorithm also generates bad result for some networks. This has been shown by Y. Chen [15]. One of the others algorithm is the brute-force algorithm. It uses the path function and have presented by V. A. Netes [16].

The network model is an undirected stochastic graph  $G = (V, E)$ , where  $V$  is the vertex set, and  $E$  is the set of undirected edges. An incidence relation which associates with each edge of  $G$  a pair of nodes of  $G$ , called its end vertices. The edges and nodes are the components of a network that can fail with known probability. In real problems, these probabilities are usually computed from statistical data. The reliability of a network is the probability that at least one path is operational from source to sink.

This paper is organized as follows. First, we give a brief introduction to Binary decision diagrams (BDD) in section II. Then, in section III we have define three types of network reliability. In section IV, we have proposed the description of our method for computing network reliability by using BDD. In section V, we have shown the experimental results. Finally we draw some conclusions in section VI.

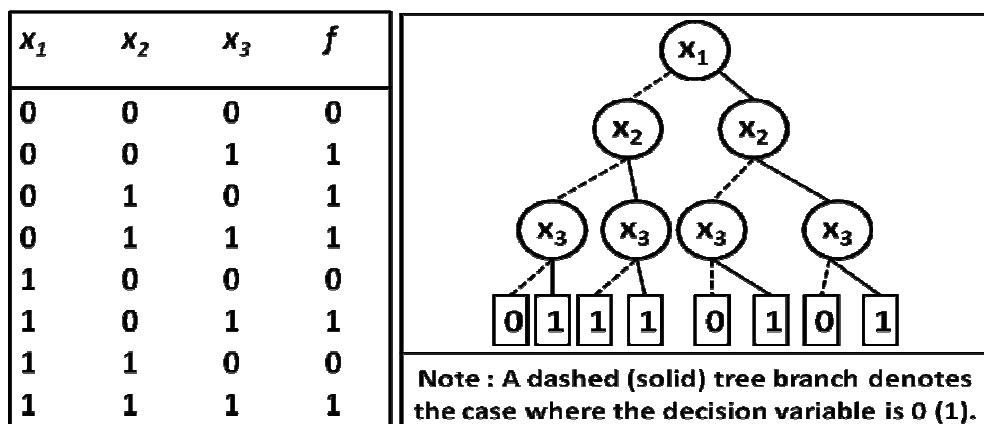
## II. BINARY DECISION DIAGRAMS

Akers [17] first introduced BDD to represent Boolean functions i.e. a BDD is a data structure used to represent a Boolean Function. Bryant [18] popularized the use of BDD by introducing a set of algorithms for efficient construction and manipulation of BDD structure. The BDD structure provides compact representations of Boolean expressions. A BDD is a directed acyclic graph (DAG) based on the Shannon decomposition. The Shannon decomposition for a Boolean function is defined as follows:

$$f = x \cdot f_{x=1} + \bar{x} \cdot f_{x=0} \quad (2)$$

where  $x$  is one of the decision variables, and  $f$  is the Boolean function evaluated at  $x = i$ .

By using Shannon's decomposition, any Boolean expression can be transformed in to binary tree. The authors [19] have shown a method to minimize Boolean expression with sum of disjoint product functions by using BDD. BDD are used to work out the terminal reliability of the links. In the network reliability framework, Sekine & Imai [20] have shown how to functionally construct the corresponding BDD. The authors [21] have shown an alternate approach to find the network reliability by using BDD. Figure 1 shows the truth table of a Boolean function  $f$  and its corresponding Shannon tree.

**Figure 1:** Truth Table of a Boolean Function and Decision Tree of the given Boolean Function

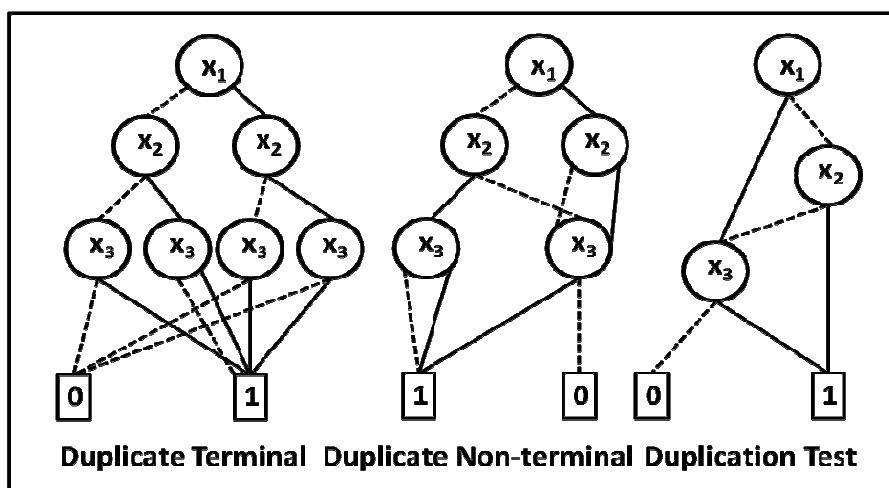
Sink nodes are labelled either with 0, or with 1, representing the two corresponding constant expressions. Each internal node  $u$  is labelled with a Boolean variable  $\text{var}(u)$ , and has two out-edges called 0-edge, and 1-edge. The node linked by the 1-edge represents the Boolean expression when  $x_i = 1$ , i.e.  $f_{x_i=1}$ ; while the node linked by the 0-edge represents the Boolean expression when  $x_i = 0$ , i.e.  $f_{x_i=0}$ . The two outgoing edges are given by two functions  $\text{low}(u)$  and  $\text{high}(u)$ . Indeed, such representation is space consuming. It is possible to shrink by using following three postulates.

**Remove Duplicate Terminals:** Delete all but one terminal vertex with a given label, and redirect all arcs into the deleted vertices to the remaining one.

**Delete Redundant Non Terminals:** If non terminal vertices  $u$ , and  $v$  have  $\text{var}(u) = \text{var}(v)$ ,  $\text{low}(u) = \text{low}(v)$ , and  $\text{high}(u) = \text{high}(v)$ , then delete one of the two vertices, and redirect all incoming arcs to the other vertex.

**Delete Duplicate tests :** If non terminal vertex  $v$  has  $\text{low}(v) = \text{high}(v)$ , then delete  $v$ , and redirect all incoming arcs to  $\text{low}(v)$ .

If we apply all these three rules then the decision tree can be reduced. The shrinking process is shown in figure 2.

**Figure 2:** Shrinking Process of the Decision Tree

## 2.1 Ordered Binary Decision Diagram

For an ordered BDD (OBDD), we impose a total ordering  $\prec$  over the set of variables and require that for any vertex  $u$ , and either non terminal child  $v$ , their respective variables must be ordered. The authors [22, 23] have shown two different methods to find the reliability of the network by using OBDD.

## 2.2 Dual Binary Decision Diagram

If two or more BDD have the same size and representing the same Boolean function, then these BDD are known as Dual BDD, because they are Dual of each other. The size of the BDD means the total

number of non terminal vertices and the number of non terminal vertices at particular level [24]. A particular sequence of variables is known as a variable ordering. It has been observed that the size of the BDD strongly depends on the ordering of variables [25]. There are three types of variable ordering (optimal, good and bad) depending on the size of the different BDD [26]. An ordering is said to be optimal if it generates the minimum size BDD. A new approach for finding various optimal variable ordering to generate minimum size BDD has shown by Singhal [27]. Herrmann has shown the process how to improve the reliability of a network by using augmented BDD [28, 29].

### 2.3 Modified Binary Decision Diagram

The modified binary decision diagram (MBDD) is a binary decision diagram which is either dual BDD or the smaller size BDD. [30].

## III. NETWORK RELIABILITY

The reliability of a network  $G$  is the probability that  $G$  supports a given operation. We distinguish three kinds of operation and hence three kind of reliability [31].

### 3.1 Two Terminal Reliability

It is the probability that two given vertices, called the source and the sink, can communicate. It is also called the terminal-pair reliability [32].

### 3.2 K Terminal reliability

When the operation requires only a few vertices, a subset  $k$  of  $N(G)$ , to communicate each other, this is  $K$  terminal reliability [33].

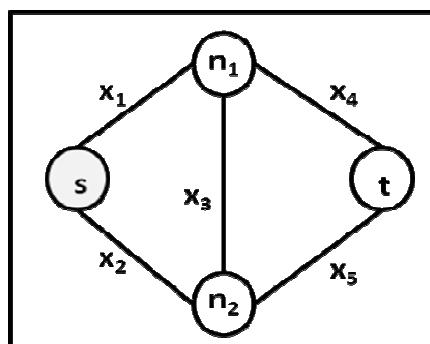
### 3.3 All Terminal Reliability

When the operation requires that each pair of vertices is able to communicate via at least one operational path, this is all terminal reliability. We can see that 2-terminal reliability and all terminal reliability are the particular case of  $K$ -terminal reliability.

## IV. COMPUTATION OF NETWORK RELIABILITY

Let us take an example of an undirected network  $G(V, E)$  with single source ( $S$ ) and single sink ( $T$ ) as taken by the authors [34], shown in figure 3. The graph has four nodes and five edges. There are only four path exist from source to sink. These min-paths are as follows:

$$H_1 = \{x_1, x_4\}, H_2 = \{x_2, x_5\}, H_3 = \{x_1, x_3, x_5\}, H_4 = \{x_2, x_3, x_4\}.$$



**Figure 3:** An Undirected Graph

In [34], the authors have generated the OBDD based on edge expansion diagram by considering that all the nodes are perfect and then compute the reliability by applying Shannon's decomposition. The edge expansion diagram and the respective OBDD are shown in figure 4 and figure 5 respectively. This OBDD (figure5) contains 8 non-terminal vertices. The authors also have generated the OBDD by considering the network with imperfect nodes as well as imperfect links. This is shown in figure 6. This OBDD (imperfect nodes) contains 16 non-terminal vertices.

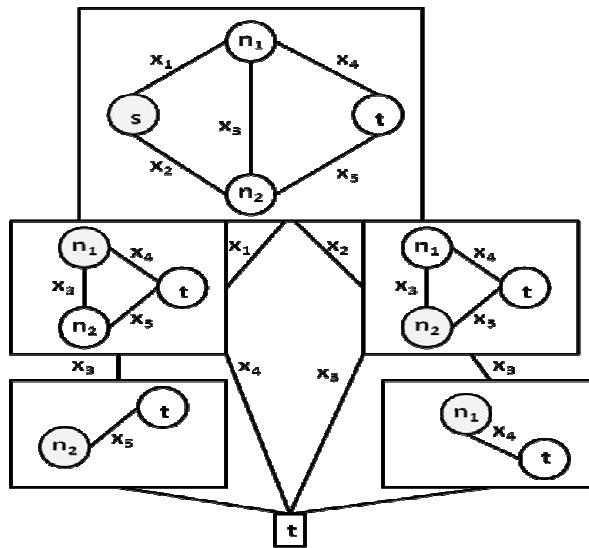


Figure 4: Edge Expansion Diagram

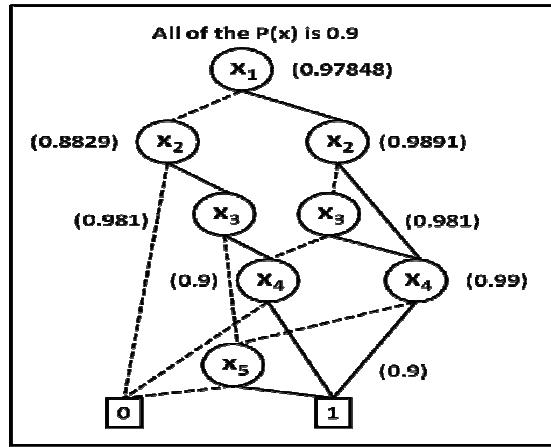


Figure 5: OBDD and its probability computation (perfect nodes)

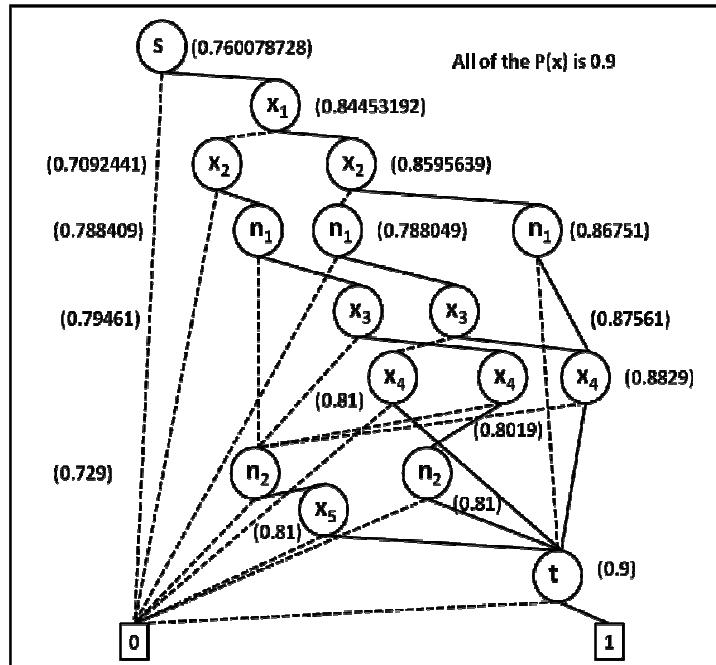


Figure 6: OBDD and its Probability computation (Imperfect Nodes)

Here we have tried to reduce the size of the OBDD (imperfect nodes) by applying an efficient

algorithm. Our algorithm executes in three main steps.

1. Order the link by applying a good heuristic approach.
2. Generate the BDD by min-path function for the perfect vertices as well as for imperfect vertices.
3. Apply Shannon's decomposition to compute the reliability.

**Heuristic Approach:** The heuristic approach is given below:

(i) Traverse the graph from source S to sink T. Find all the min-paths from source to sink. These are  $H_1 = \{x_1, x_4\}$ ,  $H_2 = \{x_2, x_5\}$ ,  $H_3 = \{x_1, x_3, x_5\}$  and  $H_4 = \{x_2, x_3, x_4\}$ .

(ii) Check whether these paths are disjoint or not. If all the paths are disjoint then we can select any one disjoint path then second then third and so on.

(iii) If all min-paths are not disjoint then find only those min-paths which are disjoint. We have found that the min-paths  $H_1$  and  $H_2$  are disjoint. Since the degree of nodes  $n_1$  and  $n_2$  are same, then we can move from source S via min-path  $H_1$  or  $H_2$ . To choose either  $H_1$  or  $H_2$ , we will give preference to min-path  $H_1$ , then middle edge  $x_3$  and then min-path  $H_2$ .

A particular sequence of variables is known as a variable ordering. It has been observed that the size of the BDD strongly depends on the ordering of variables. An ordering is said to be optimal if it generates the minimum size BDD. An ordering may be good or bad depending on the size of the BDD.

By applying our heuristic we have found the variable ordering

$$x_1 < x_4 < x_3 < x_2 < x_5$$

Now we have to build a reliability function (min-path function) for perfect nodes as well as imperfect nodes.

The min-path function R for perfect nodes is the union of all min-paths from source to sink and is given below:

$$R = H_1 \cup H_2 \cup H_3 \cup H_4 \quad (3)$$

Again we have to build the reliability function by considering that the nodes and the links in the CCN may fail. It means the CCN has imperfect vertices as well as imperfect links. To generate the reliability function for imperfect nodes, we consider an undirected edge  $e_i$  as shown in figure 7, then it must contain two end vertices say  $v_i$  and  $v_j$ . Therefore its incident edge function is  $e_i = v_i e_i v_j$ . So by substituting  $v_i e_i v_j$  for  $e_i$  in the path function of perfect nodes. We get the reliability function with imperfect nodes.

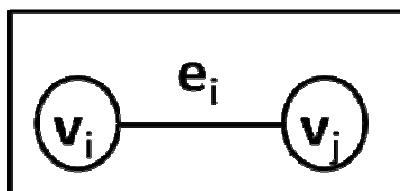


Figure 7: An undirected Edge  $e_i$  with End Vertices

The computation of the probability of the BDD of can be calculated recursively by resorting to the Shannon decomposition.

$$\Pr\{F\} = p_1 \Pr\{Fx_1=1\} + (1 - p_1) \Pr\{Fx_1=0\} = \Pr\{Fx_1=0\} + p_1 (\Pr\{Fx_1=1\} - \Pr\{Fx_1=0\}) \quad (4)$$

where  $p_1$  is the probability of the Boolean variable  $x_1$  to be true and  $(1 - p_1)$  is the probability of the Boolean variable  $x_1$  to be false.

The BDD and its probability computation for imperfect vertices are shown in figure 8.

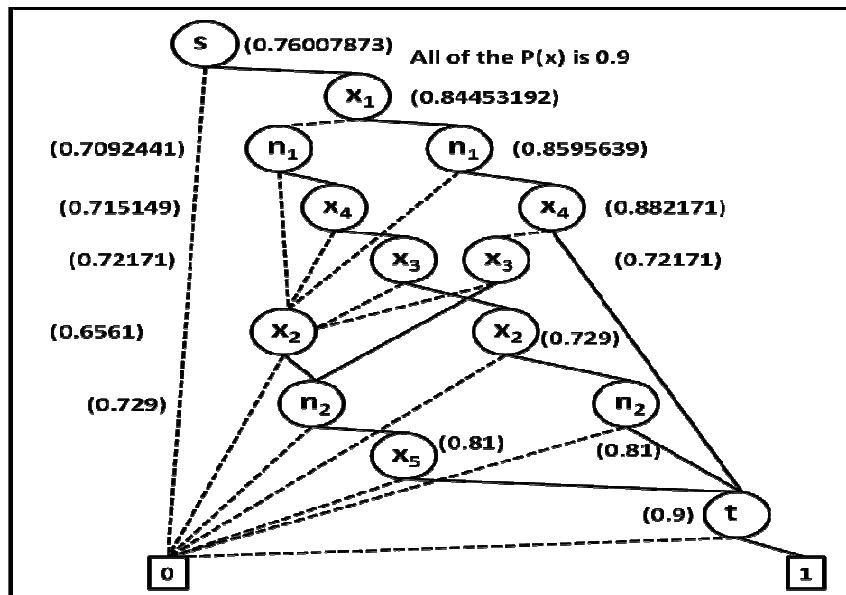


Figure 8: MBDD and its Probability Computation (Imperfect Nodes)

This BDD (imperfect nodes) contains only 14 non-terminal vertices, while the OBDD shown by [34] contains 16 non-terminal vertices. So the size of this BDD is much less than the size of the existing OBDD. Therefore this BDD (14 non-terminal vertices) is known as MBDD because the size of the MBDD is less than the size of the existing OBDD. However the reliability result obtained by OBDD and MBDD is same.

The BDD obtained for perfect nodes of the example network (figure 4) is shown in figure 9. This BDD has the same size as shown by [34] for perfect vertices and the reliability result is same. Therefore the BDD shown in figure 9 is the dual BDD of the existing OBDD (figure 5) for perfect nodes. We can also say that the BDDs shown in figure 5 and figure 9 are dual of each other.

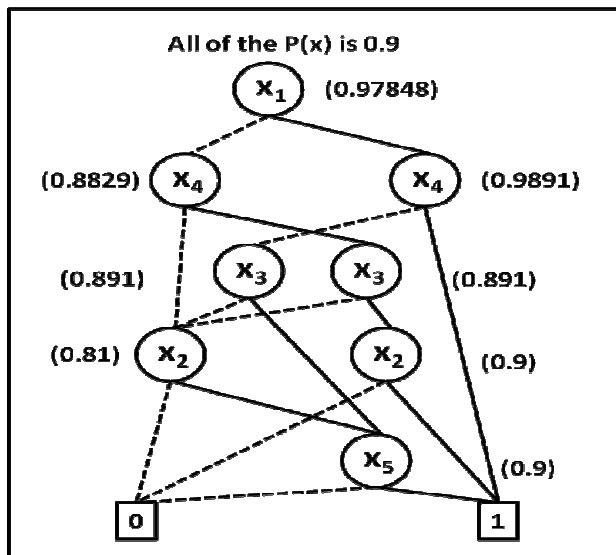


Figure 9: Dual BDD of figure 5

## V. EXPERIMENTAL RESULTS

Our program is written in the C language and computations are done by using a Pentium 4 processor with 512 MB of RAM. The computation speed heavily depends on the variables ordering because the size of the BDD heavily depends on the variable ordering. The size of BDD means the total number of nodes in the BDD and number of nodes in a particular level. There are several variables ordering may be possible for constructing the dual BDD (perfect nodes) of the given CCN. We have constructed

only one dual BDD of the given CCN and compute the reliability of the given CCN by using dual BDD. We have found that the reliability obtained by dual BDD and OBDD are same. Here the size of the OBDD (perfect nodes) is same as the size of dual BDD (perfect nodes). But the size of the MBDD (imperfect nodes) is much less than the size of the existing OBDD (imperfect nodes).

## VI. CONCLUSIONS

In a real network the nodes and link may fail. . The complexity of the reliability analysis will grow exponentially if the nodes of the network are imperfect. To overcome this problem, our algorithm based on MBDD is proposed to compute the reliability of a network with imperfect nodes. We found that the size of the MBDD is much less than the size of the existing OBDD and the reliability results are equal. So the complexity of the MBDD is less than the complexity of the OBDD by an existing edge expansion approach. In future we extend our algorithm to compute the K-terminal reliability with imperfect nodes.

## REFERENCES

- [1] A. Rauzy (1993), "New algorithms for fault tolerant trees analysis", Reliability Engineering and System Safety, vol. 5, no. 59, pp. 203–211.
- [2] Andrea Bobbio, Caterina Ferraris, Roberta Terruggia (2006), "New Challenges in Network Reliability Analysis", Technical Report, pp. 1-8, TR-INF-UNIPMN.
- [3] C. Lucet and J.-F. Manouvrier (1999), "Exact methods to compute network reliability", in Statistical and Probabilistic Models in Reliability, D. C. Ionescu and N. Limnios, Eds. Birkhauser Boston, pp. 279–294.
- [4] D. Torrieri (1994), "Calculation of node-pair reliability in large networks with unreliable nodes," IEEE Trans. Reliability, vol. 43, no. 3, pp. 375–377.
- [5] P. A. Jensen and M. Bellmore (1969), "An algorithm to determine the reliability of a complex system," IEEE Trans. Reliability, vol. 18, no. 4, pp. 169–174.
- [6] Y. G. Chen and M. C. Yuang (1996) , "A cut-based method for terminal-pair reliability," IEEE Trans. Reliability, vol. 45, no. 3, pp. 413–416.
- [7] A. Satyanarayana and M. K. Chang (1983), "Network reliability and the factoring theorem," Networks, vol. 13, pp. 107–120.
- [8] L.B. Page and J. E. Perry (1989), "Reliability of directed networks using the factoring theorem", IEEE Trans. Reliability 38: 556-562.
- [9] F. Moskowitz (1958), "The analysis of redundancy networks," AIEE Trans. Communications and Electronics, vol. 39, pp. 627–632.
- [10] M. Resende (1986), "A program for reliability evaluation of undirected networks via polygon-to-chain reductions," IEEE Trans. Reliability, vol. 35, no. 1, pp. 24–29.
- [11] K. K. Agarwal (1975), J.S. Gupta and K. B. Mishra, "A simple method for evaluation of a communication system", IEEE Trans. Communication 23:563-566.
- [12] O. Theologou and J. Carlier (1991), "Factoring and reductions for networks with imperfect vertices", IEEE Trans. Reliability, vol. R-40, pp. 210–217.
- [13] Sy-Yen Kuo, Fu-Min Yeh, and Hung-Yau Lin (2007), "Efficient and Exact Reliability Evaluation for Networks With Imperfect Vertices", IEEE Transactions on Reliability, vol. 56, no. 2, pp.288-300.
- [14] Liudong Xing (2008), "An Efficient Binary Decision Diagram Based Approach for Network Reliability and Sensitivity Analysis", IEEE Trans. On Systems, Man and Cybernetics, pp. 105-115.
- [15] Y. Chen, A. Q. Hu, K. W. Yip, X. Hu, and Z. G. Zhong (2003), "A modified combined method for computing terminal-pair reliability in networks with unreliable nodes," in Proceedings of the 2nd Int'l Conference on Machine Learning and Cybernetics, , pp. 2426–2429.
- [16] V. A. Netes and B. P. Fillin (1996), "Consideration of node failure in network reliability calculations", IEEE Trans. Reliability 45: 127-128.
- [17] B. Akers (1992), "Binary decision diagrams", IEEE Trans. Computers, vol. C-27, pp.509–516, 1978.
- [18] R. E. Bryant, "Symbolic Boolean manipulation with ordered binary-decision diagrams", ACM Computing Surveys, vol. 24, no. 3, pp. 293–318.
- [19] D. Sensarma, S. Naskar and S. S. Sarma (2012), "On An Optimization Technique using Binary Decision Diagram", IJCSEA, Vol.2 No. 1, pp73-86
- [20] H. Imai, K. Sekine, and K. Imai (1999), "Computational investigations of all terminal network reliability via BDDs", IEICE Transactions on Fundamentals, vol. E82-A, no. 5, pp.714–721.
- [21] Manoj Singh, R. K. Chauhan, Girish Sharma(2010) , "An alternate approach to compute the reliability of a computer communication network using Binary Decision Diagrams", Communications in Computer and Information Science, pp. 160-170, IC3 2010, Springer-Verlag Berlin Heidelberg.

- [22] F. Yeh, S. Lu, and S. Kuo (2002), "OBDD-based evaluation of k-terminal network reliability", IEEE Trans. Reliability, vol. R-51, no. 4, pp. 443-451.
- [23] F. M. Yeh, and S. Y .Kuo (1997), "OBDD based network reliability calculation", IEEE Electronics Letter 33(9) : 759-760.
- [24] Manoj Singhal, R. K. Chauhan, , Girish Sharma (2010), "A New Optimal Approach for evaluating the size of BDD (Binary Decision Diagram) for calculating the Reliability of a CCN (Computer Communication Network)". International Journal of Advanced Networking and Applications. Vol. 1 issue 4. pp. 230-235.
- [25] Manoj Singhal, R. K. Chauhan, , Girish Sharma (2010), "Effects of Variable Ordering on Binary Decision Diagrams for Computation of Reliability of a Computer Communication Network", Journal of Computer Science, Vol. 4, issue 6, pp. 1837-1845.
- [26 ] Manoj Singhal, R. K. Chauhan, Girish Sharma (2010), "Network Reliability Computation using Different Binary Decision Diagrams", International Journal of Distributed and Parallel Systems, Vol. 1, No. 1, pp. 82-91.
- [27] Manoj Singhal, R. K. Chauhan, Girish Sharma (2011), "A New approach for finding the Various Optimal Variable Ordering to generate the Binary Decision Diagrams (BDD) of a Computer Communication Network", International Journal of Computer Applications, Vol. 31, No.3, pp. 1-8.
- [28] J. U. Herrmann (2009), "On Augmented OBDD and Performability for Sensor Networks," Int'l J. Performability Engineering, accepted for publication 2009.
- [29] J. U. Herrmann (2010), "Improving Reliability Calculation with Augmented Binary Decision Diagrams", IEEE International Conference on Advanced Information Networking and Applications, pp. 328-333.
- [30] Manoj Singhal, R. K. Chauhan, Girish Sharma (2009), "Use of Modified Binary Decision Diagrams in Reliability Evaluation of a Directed Computer Communication Network", The Icfai University Journal of Computer Sciences, Vol. III No. 3, pp. 22-30.
- [31] G. Hardy, C. Lucet, and N. Limnios (2005), "Computing all-terminal reliability of stochastic networks with binary decision diagrams", in Proc.11th International Symposium on Applied Stochastic Models and Data Analysis, pp. 1468-1473.
- [32] S. Kuo, S. Lu, F. Yeh (1999), "Determining terminal pair reliability based on edge expansion diagrams using OBDD", IEEE Trans. Reliability, Vol. 48, no. 3, pp. 234-246.
- [33] Gary Hardy, Corinee Lucet and Nikolaos Limnios (2007), "K-Terminal Network reliability Measures with Binary Decision Diagrams", IEEE Tran. On Reliability Vol.56, No. 3.
- [34] F. M. Yeh, H. Y. Lin and Sy-Yen Kuo (2002), "Analyzing network reliability with imperfect nodes using OBDD", IEEE proceedings of PRDC'02.

## Authors

**Manoj Singhal** is currently working as an Associate Professor, Department of Information Technology in AIMT, Greater Noida (U.P.), India. He had done M.Sc. (Mathematics), MCA and M. Tech (Computer Engg.). He has written one book on Discrete Mathematics. His research interest include design and development of algorithms



**R. K. Chauhan** is currently working as a Professor, Department of Computer Science and Application in Kurukshetra University, Kurukshetra. He has published more than 70 research papers in International/National Journal/Conference. He has guided 7 Ph.D. His research interest includes Data Mining, DBMS and Algorithm.



**Girish Sharma** is currently working as an Asso. Professor, Department of Computer Application in Bhai Parmanand Institute of Business Studies, New Delhi. He has published more than 15 research papers in International/National journal / Conference. His research interest includes Mathematical Modelling, Distributed Systems and Parallel Architecture.

