Volume: 01, Issue: 04, Pages: 230-235 (2010)

A New Optimal Approach for evaluating the size of BDD for calculating the Reliability of a CCN

Manoj Singhal[#], R. K. Chauhan[##], Girish Sharma[###],
[#]Asso. Prof., Dept. Of MCA,
Academy of Business and Engineering Sciences, Ghaziabad, (U.P.), India,
Email: manoj_singhal@rediffmail.com
[##] Prof. & Chairman, Deptt. Of Computer Science and Applications,
Kurukshetra University, Kurukshetra, India,
Email: rkc.dcsa@gmail.com
[###] Asstt. Prof., Deptt. Of MCA, Bhai Parmanand Institute Of Business Studies, Delhi, India,
Email: gkps123@gmail.com

ABSTRACT

In this paper we adopted a new approach for evaluating the size of the BDD and also generated modified binary decision diagrams for calculating the reliability of the given directed computer communication network. We have also shown that these modified binary decision diagrams are of minimum size. Conclusively, we can say that more than one optimal variable ordering may exist for finding the reliability of particular networks.

Key Words Binary Decision Diagrams (BDD), Directed Acyclic Graph (DAG), Computer communication Network (CNN), Ordered Binary Decision Diagrams (OBDD).

Date of Submission: December 16, 2009 Date Accepted: February 07, 2010

1. 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 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. 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.**" There are so many exact methods for computation of network reliability. The network model is a directed stochastic graph G = (V, E), where V is the vertex set, and E is the set of directed edges. An incidence relation which associates with each edge of G a pair of nodes of G, called its end vertices. The edges represent components that can fail with known probability. In real problems, these probabilities are usually computed from statistical data.

The problem related with connection function is NP-hard [13]. The same thing is observed for planar graphs[12]. In the exact method there are two classes for the computation of the network reliability. The first class deals with the enumeration of all the minimum paths or cuts. A path is a subset of components (edges and/or vertices), that guarantees the source and the sink to be connected if all the components of this subset are functioning. A path is a minimal if a subset of elements in the path does not exist that is also a path. A cut is a subset of components (edges and/or vertices), whose failure disconnect the source and sink. A cut is a minimal if the subset of elements in the cut does not exist that is also a cut. The probabilistic evaluation uses the inclusion-exclusion, or sum of disjoint products methods because this enumeration provides non-disjoint events. Numerous works about this kind of methods have been presented in literature [14, 21, 22].

In the second class, the algorithms are based on graph topology. In the first process we reduce the size of the graph by removing some structures. These structures as polygon-to-chain [15] and delta-to-star reductions [11]. By this we will be able to compute the reliability in linear time and the reduction will result in a single edge. The idea is to decompose the problem in to one failed and another functioning. The same was confirmed by

Volume: 01, Issue: 04, Pages: 230-235 (2010)

Theologou & Carlier [18] for dense networks. Satyanarayana & Chang [4] and Wood [20] have shown that the factoring algorithms with reductions are more efficient at solving this problem than the classical path or cut enumeration methods.

2. BINARY DECISION DIAGRAM

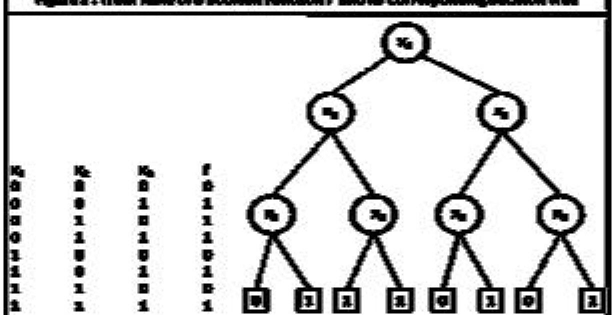
Akers [5] first introduced BDD to represent Boolean functions i.e. a BDD is a data structure used to represent a Boolean Function. Bryant [19] popularized the use of BDD by introducing a set of algorithms for efficient construction and manipulation of BDD structure. The BDD structure provides compact 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:

$$f = x.\, f_{x=1} + \bar{x}.\, f_{x=0}$$

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. BDD are used to work out the terminal reliability of the links. Madre and coudert [17] found BDD usefulness in reliability analysis is which was further extended by Rauzy [2, 3]. They are specially used to assess fault trees in system analysis. In the network reliability framework, Sekine & Imai [9], and Trivedi [26] have shown how to functionally construct the corresponding 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 f and its Corresponding Decision Tree



Note : A dashed (solid) tree branch denotes the case where the decision variable is 0 (1).

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 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. $f_{xi=1}$; while the node linked by the 0-edge represents the Boolean expression when xi = 0, i.e. fxi=0. The two outgoing edges are given by two functions low(u) and high(u).

Indeed, such representation is space consuming. It is possible to shrink by using following three postulates.

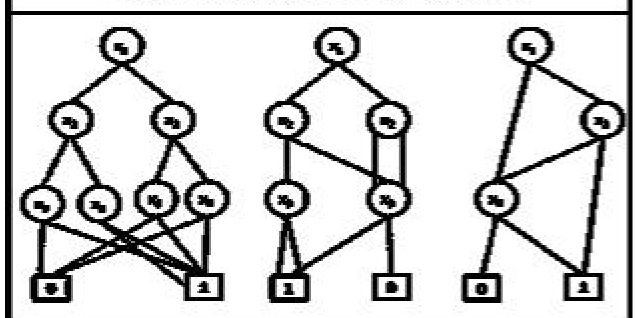
Remove Duplicate Terminals : Delete all but one terminal vertex with a given label, and redirect all arcs into the deleted vertices to the remaining one.

Delete Redundant Non Terminals : If non terminal vertices u, and v have var(u) = var(v), low(u) = low(v), and high(u) = 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 low(v) = high(v), then delete v, and redirect all incoming arcs to low(v).

If we apply all these three rules then the above decision tree can be reduced in to the diagrams given below in figure 2.

Figure 2 : Reduction of Decision Tree to BDD



Duplicate Terminals, Duplicate Non-Terminals, Redundant Tests

3. 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 [1, 10].

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 [25].

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 [7].

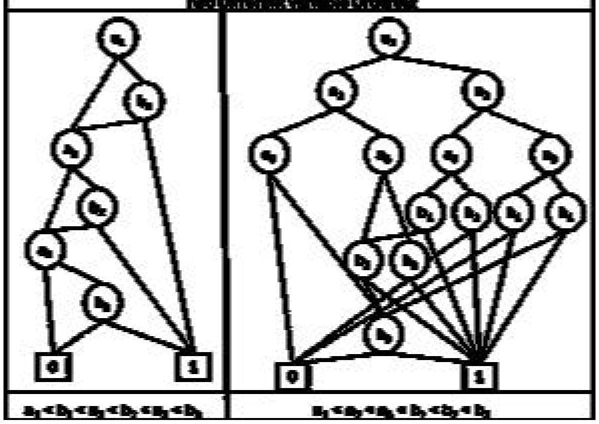
All Terminal Reliability : When the operation requires that each pair of vertices is able to communicate via at least one operational path, this is all terminal reliability. We can see that 2-terminal terminal reliability and all terminal reliability are the particular case of K-terminal reliability [8].

Effects of Variable Ordering: 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. The size of BDD means the total number of non-terminal nodes in the BDD and number of nodes in a particular level. An ordering is said to be optimal if it generates the minimum size BDD [23, 24]. Let us consider a Boolean function f given below depends on six different variables.

$$f = a_1.b_1 + a_2.b_2 + a_3.b_3 \quad \text{------------------} (1)$$

Where '.' and '+' denotes the AND & OR operation respectively. If the variables are ordered by "$a_1 < b_1 < a_2 < b_2 < a_3 < b_3$" and "$a_1 < a_2 < a_3 < b_1 < b_2 < b_3$" the BDD are shown below:

Figure 3: BDD Representation of a single Function $f = a_1.b_1 + a_2.b_2 + a_3.b_3$ for Two Different Variable Ordering



The researchers [ ]said that the ordering "$a_1 < b_1 < a_2 < b_2 < a_3 < b_3$" is good ordering, because it contains minimum number of non-terminal vertices. Since the function f is depend on six different variables so its BDD must have minimum six non-terminal vertices. So in our opinion the good ordering is the optimal ordering. The minimum number of non-terminal vertices of the BDD is 2n[ ]. Where n is the number of terms in the function. Here n = 3 , so for optimal ordering the minimum number of non-terminal vertices is 6 for the Boolean function f expressed in equation (1). We can generalize this function f as shown below:

$$f = a_1.b_1 + a_2.b_2 + a_3.b_3 + ...+ a_n.b_n \quad \text{----------} (2)$$

The BDD of this function f expressed in equation (2) has 2n non-terminal vertices one for each variable. The work carried out by [] has shown that there are 2n non-terminal vertices for the general function f represented in equation (2), but this will probably not work if the Boolean function f is different. Now let us consider a Boolean function f given below:

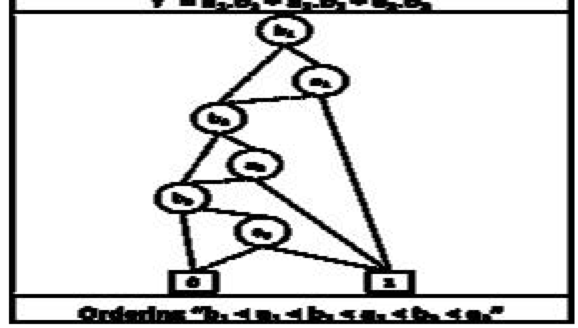
$$f = a_1.b_1.c_1 + a_2.b_2.c_2 + a_3.b_3 \quad \text{---------------} (3)$$

According to the researchers, there are only three terms, so the minimum number of non-terminal vertices of the BDD of the Boolean function f expressed in equation (3) is 2 * 3 = 6. But this Boolean function f has eight different variables, so it's BDD must contains minimum 8 non-terminal vertices.

Therefore the statement regarding 2n non-terminal vertices can be interpreted in a more general way by using the total number of different variables. We can say that the minimum number of non-terminal vertices of a Boolean function f is equal to the number of number of different variables used in the function. The researchers also said that the ordering "$a_1 < a_2 < a_3 < b_1 < b_2 < b_3$" is bad ordering.

Now let us try to find out another variable ordering for which the size of the BDD is also minimum. For this, we consider the function f represented in equation (1). If we take the ordering "$b_1 < a_1 < b_2 < a_2 < b_3 < a_3$", the BDD is shown below:

Figure : 4 BDD Representation of the Function $f = a_1.b_1 + a_2.b_2 + a_3.b_3$

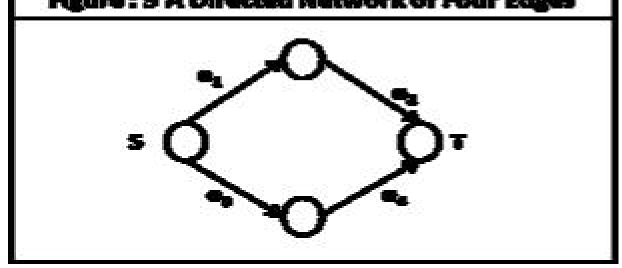


From the above BDD, we find that there are 6 non-terminal vertices. Therefore the size of the BDD is also minimum for "$b_1 < a_1 < b_2 < a_2 < b_3 < a_3$" ordering. We can also generate minimum size BDD for the generalize function expressed in equation (2) by taking the ordering "$b_1 < a_1 < b_2 < a_2 < b_3 < a_3 < ....< b_n < a_n$". Thus we can say that more than one optimal ordering may exist for generating the BDD of the Boolean function f. This may be possible only when the Boolean function f has different variables. These minimum size BDD is known as Modified Binary Decision Diagrams (MBDD).

Use of MBDD in Reliability Evaluation of a Directed CCN

Let us consider an example of a directed CCN given below in the form of a graph.

Figure : 5 A Directed Network of Four Edges



Int. J. Of Advanced Networking and Applications
Volume: 01, Issue: 04, Pages: 230-235 (2010)

The network consists of two min-paths from source to sink. These min-paths are $H_1 = \{e_1, e_2\}$ and $H_2 = \{e_3, e_4\}$.

Let H_1, H_2,-----H_n be the n min-paths then the network connectivity C can be represented as a logical OR of its min-paths.

$$C = H_1 U H_2 U \text{-------} H_n$$

So the point to point reliability is :

$$R_s = Pr\{C\} = Pr\{ H_1 U H_2 U \text{-------} H_n \} \quad \text{--------} (4)$$

So the network connectivity of the given network can be expressed as

$$C_{1\text{-}4} = e_1 e_2 U e_3 e_4 \quad \text{---} (5)$$

The probability of the union of non-disjoint events, as in Formula (4), can be computed by several techniques: Here we use the inclusion-exclusion principle.

Inclusion-exclusion Formula : One method of transforming a Boolean expression $\Phi(G)$ into a probability expression is to use Poincare's theorem, also called inclusion-exclusion method[7]. Let us consider an example with two minimal paths H_1 and H_2 and the Boolean expression $\Phi(G) = H_1 + H_2$, then the probability expression $E(\Phi(G))$ can be expressed as follows:

$$E(H_1 + H_2) = E(H_1) + E(H_2) - E(H_1 H_2)$$

Poincare's formula for m min-paths :

$$E(\Phi(G)) = \sum_{1 \le i \le m} E(H_i) - \sum_{1 \le i \le j \le m} H_i . H_j + (-1)^{m+1} . E(H_1 . H_2 . H_3 \ldots H_m)$$

Let P_i denote the probability of edge e_i of being working, by applying the Classical inclusion-exclusion formula for calculating the probability of given network (figure 5), we get

$$R_{1\text{-}4} = Pr\{C_{1\text{-}4}\} = p_1 p_2 + p_3 p_4 - p_1 p_2 p_3 p_4 \quad \text{---------}(6)$$

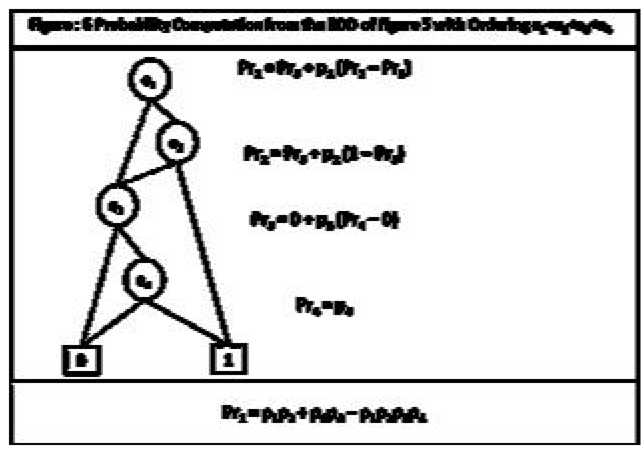
There are only two possible existing variable orderings to generate modified BDD of the given network (Figure 5) and we will show that the network reliability, which is obtained by Poincare theorem is equal to the network reliability, which is obtained recursively by modified BDD (all existing variable orderings) of the same network (Figure 5).

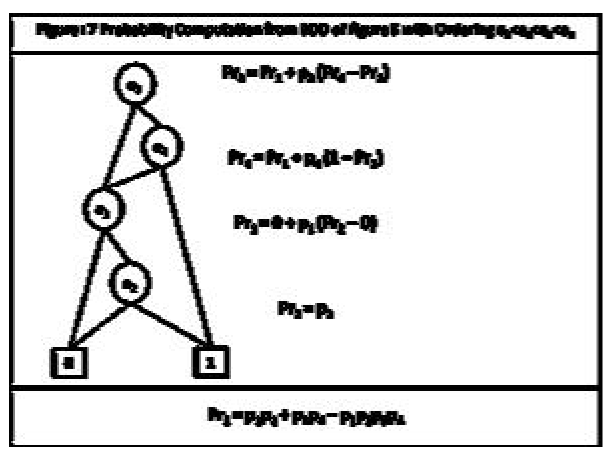
We apply the Shannon's decomposition to the Boolean connectivity function of the directed network expressed as the union of the min-paths in Formula (5).
The computation of the probability of the BDD of figure 5 can be calculated recursively by resorting to the Shannon decomposition.

$$Pr\{F\} = p_1 Pr\{Fx_1 = 1\} + (1 - p1) Pr\{Fx_1 = 0\} = Pr\{Fx_1 = 0\} + p_1 (Pr\{Fx_1 = 1\} - Pr\{Fx_1 = 0\}) \text{ ---}(7)$$

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 computation of the probability from the BDD of the given network is shown in figure 6 and in figure 7 for two possible ordering by applying Shannon's decomposition.

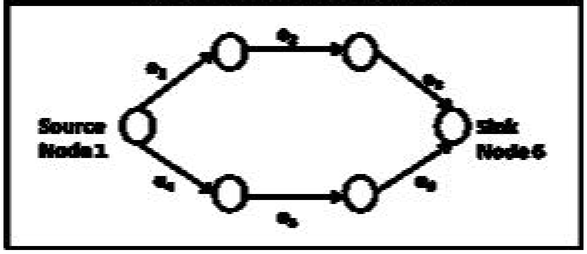




From figure 6 and figure 7, we found that more than one optimal variable ordering exists for finding the reliability of a particular directed CCN. The meaning of particular directed CCN is a network that has all its min-paths disjoint. If we consider the network given below

Volume: 01, Issue: 04, Pages: 230-235 (2010)

Figure : 8 A Directed Network



The authors[16] already shown that more than one optimal ordering exist for finding the reliability of the above CCN shown in figure 8.

4. 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 only two variable orderings are possible for constructing the modified BDD of the given CCN. We have constructed these two modified BDD of the given CCN and compute the reliability of the given CCN by using these modified BDD. We found that the reliability obtained in each case by using BDD is same as the reliability obtained by inclusion-exclusion formula. We also found that the size of the BDD is minimum only in all the cases.

5. Conclusion

We found that more than one optimal ordering may possible for finding the reliability of the directed CCN. The generated BDD are called the Modified BDD. Our future work will focus on computing other kinds of reliability and reusing the BDD structure in order to optimize design of network topology.

REFERENCES

[1] Andrea Bobbio, Caterina Ferraris, Roberta Terruggia, "New Challenges in Network Reliability Analysis", *Technical Report*, pp. 1-8, TR-INF-UNIPMN 2006.

[2] A. Rauzy, "New algorithms for fault tolerant trees analysis", *Reliability Engineering and System Safety*, vol. 5, no. 59, pp. 203–211, 1993.

[3] A. Rauzy, "A new methodology to handle Boolean models with loops", *IEEE Trans. Reliability*, vol. R-52, no. 1, pp. 96–105, 2003.

[4] A. Satyanarayana and M. K. Chang, "Network reliability and the factoring theorem", *Networks*, vol. 13, pp. 107–120, 1983.

[5] B. Akers, "Binary decision diagrams", *IEEE Trans. Computers*, vol. C-27, pp.509–516, 1978.

[6] C. Lucet and J.-F. Manouvrier, "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, 1999.

[7] F. Yeh, S. Lu, and S. Kuo, "OBDD-based evaluation of k-terminal network reliability", *IEEE Trans. Reliability*, vol. R-51, no. 4, pp. 443-451, 2002.

[8] G. Hardy, C. Lucet, and N. Limnios, "Computing all-terminal reliability of stochastic networks with binary decision diagrams", in *Proc.11th International Symposium on Applied Stochastic Models and Data Analysi*, 2005, pp. 1468-1473.

[9] H. Imai, K. Sekine, and K. Imai, "Computational investigations of all terminal network reliability via BDDs", *IEICE Transactions on Fundamentals*, vol. E82-A, no. 5, pp.714–721, 1999.

[10] J. Carlier and C. Lucet, "A decomposition algorithm for network reliability evaluation", *Discrete Applied Mathematics*, vol. 65, pp. 141–156, 1996.

[11] J. P. Gadani, "System effectiveness evaluation using star and delta transformations", *IEEE Trans. Reliability*, vol. R-30, no. 1, pp. 43–47, 1981.

[12] J. S. Provan, "The complexity of reliability computations on planar and acyclic graphs", *SIAM J. Computing*, vol. 15, no. 3, pp. 694–702, 1986.

[13] M. O. Ball, "Computational complexity of network reliability analysis: An overview", *IEEE Trans. Reliability*, vol. R-35, no. 3, pp. 230–239, 1986.

[14] M. O. Locks, "A minimizing algorithm for sum of disjoint products", *IEEE Trans. Reliability*, vol. R-36, no. 4, pp. 436–445, 1987.

[15] M. S. Choi and C. H. Jun, "Some variant of polygon-to-chain reductions in evaluating reliability of undirected network", *Microelectron. Reliab.*, vol. 35, no. 1, pp. 1–11, 1985.

[16] Manoj Singhal, R. K. Chauhan, Girish Sharma, "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, July 2009.

[17] O. Coudert and J. C. Madre, "Implicit and incremental computation of primes and essential primes of Boolean functions", in *Proc. of the 29th ACM/IEEE Design Automation Conference*, 1992, pp. 36–39.

[18] O. Theologou and J. Carlier, "Factoring and reductions for networks with imperfect vertices", *IEEE Trans. Reliability*, vol. R-40, pp. 210–217, 1991.

[19] R. E. Bryant, "Symbolic Boolean manipulation with ordered binary -decision diagrams",*ACM Computing Surveys*, vol. 24, no. 3, pp. 293–318, 1992.

[20] R. K.Wood, "A factoring algorithm using polygon-to-chain reductions for computing K-terminal network reliability", *Networks*, vol. 15, pp. 173–190, 1985.

[21] S. Hariri and C. S. Raghavendra,"SYREL: A symbolic reliability algorithm based on path and cut set methods", *IEEE Trans. Computers*, vol. C-36, no. 10, pp. 1224–1232, 1987.

[22] S. H. Ahmad, "Simple enumeration of minimal cut sets of acyclic directed graph", *IEEE Trans. Reliability*, vol. R-27, no. 5, pp. 484–487, 1988.

[23] S. J. Friedman and K. J. Supowit, "Finding the optimal variable ordering for binary decision diagrams", in *Proc. 24th ACM/IEEE Conference on Design Automation*, 1987, pp.348–356.

[24] S. J. Friedman and K. J. Supowit, "Finding an optimal variable ordering for binary decision diagrams", *IEEE Trans. Computers*, vol. C-39, no. 5, pp. 710–713, 1990.

[25] S. Kuo, S. Lu, F. Yeh, "Determining terminal pair reliability based on edge expansion diagrams using OBDD", IEEE Trans. Reliability, Vol. 48, no. 3, pp. 234-246, 1999.

[26] X. Zang, H. Sun, and K. Trivedi, "A BDD-based algorithm for reliability evaluation of phased mission systems", *IEEE Trans. Reliability*, vol. R-48, no. 1, pp. 50–60, 1999.

Authors Biography

Mr. Manoj Singhal is currently working as an Associate Professor, Department of Computer Application (MCA) in ABES Engineering College, Ghaziabad (U.P.).

Prof. (Dr.) R. K. Chauhan is currently working as a Chairman, 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 5 Ph.D. His research interest includes Data Mining, DBMS and Algorithm.

Dr. Girish Sharma is currently working as an Asstt. 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.