Available online at www.sciencedirect.com

SciVerse ScienceDirect

ELSEVIER

Procedia Technology 6 (2012) 196 – 205

Procedia Technology

2nd International Conference on Communication, Computing & Security [ICCCS-2012]

A Genetic Algorithm Based Approach for Topological Optimization of Interconnection Networks

P. K. Tripathy[a*], R. K. Dash[b], C. R. Tripathy[c]

[a]Department of CSE, Silicon Institute of Technology, Bhubaneswar
[b]Department of CSA, College of Engineering & Technology, Bhubaneswar
[c]Department of CSE, VSS University of Technology, Burla
[a]E-mail: pradyumnatripathy@gmail.com
[b]E-mail: drranjandash@gmail.com

Abstract

The paper addresses the two terminal reliability while designing the interconnection networks. Thus a topological optimization problem is defined as the existence of at least a reliable path between a pair of nodes satisfying the predefined cost of the network. A new method based on Genetic Algorithm is proposed to solve the above said problem. In the proposed method the chromosome as well as the genes are efficiently encoded so that the cross over provides the optimal solution with better convergence rate. The reliability of some benchmark interconnection networks are evaluated by the proposed method. The population size and the computational time of the said networks as reported in this paper ensures that the proposed method converges to it's optimal solution in very few cpu secondss, while maximizing the value of the reliability of the said network to a greater extent.

Keywords : Interconnection Network; Genetic Algorithm; Reliability; Topological Optimization

1. Introduction

One of the most important components of a computer system is its Interconnection Network, as it defines the physical layout of such systems. An Interconnection Network may consist of nodes or stations representing computers, switches, terminals, or printers, and edges denoting communication links. These links can be conventional telephone lines, optical fiber cables or microwave channels. Therefore, the suitability of such a network for various scientific and engineering applications can be assessed by careful analysis of its performance in terms of its reliability. So, there is always a challenge to design a highly reliable system, which is quite economical. An important stage of the design of such systems is to find the

* Pradyumna Kumar Tripathy. Tel.: +91-9437141874
E-mail address: pradyumnatripathy@gmail.com

doi:10.1016/j.protcy.2012.10.024

best layout of components to minimize cost while meeting a performance criterion, such as the reliability. The reliability is the probability of reaching other stations, despite the breakdown of one or several stations. Intuitively, we note that more links between stations imply different routes between each node pair, and consequently the network is more reliable; on the other hand, the network is more expensive. In general this type of design goal is to optimize the topology of the interconnection network, so as to either maximizing the reliability or minimizing the cost. A general definition of the topological design problem for a large hierarchical network is found in [1].

In literature, the topological optimization problems can be solved by either enumerative methods or heuristic methods. Some of the important enumerative methods are branch and bound [2], enumeration of spanning trees [3], shrink and search algorithm [4]. In branch and bound algorithm, the optimization problem can be decomposed into multiple sub problems for an efficient search of the exact optimal solution. But this method is only able to find the solution of the fully connected networks of maximum size of 12 nodes (60 links). The methods based on enumeration of the spanning trees can be applied only for the small networks, as for large sized networks the computational complexity in enumerating all the spanning trees prohibits using such techniques. To a different extend, Shao et al. [7] proposed the shrinking and searching algorithm (SSA) for maximizing the network reliability of a distributed access network with constraint total cost. SSA jointly considers the upper bound of the network reliability for each branch in the combinatorial tree, and the cost constraint on the possible solutions**.** But this algorithm is not as such applicable for large size networks, such as networks having more than 20 nodes, because the size of the generated combinatorial tree is too large, which prohibits the exact calculation of network reliability.

The enumerative methods discussed so far can be applied only to small networks, but for larger networks the heuristic methods viz. simulated annealing [5][6][7][8], Artificial Neural Network [9], Genetic Algorithm [18][19] are preferred because of their strength and ability to find near optimal solutions. Pierre and et al. [6] used simulated annealing to find optimal designs for packet switched networks where delay and capacity were considered, but reliability was not. AboElFotoh et al [9] used an optimize neural network for solving the network topological optimization problem. Similar kinds of works are reported in [10] and [11]. However, mapping of the topological optimization problem onto a neural network is quite cumbersome. Further, literature also records that the heuristic methods based on Artificial Neural Network may not always lead to optimal solutions.

Study of literature reports the use of genetic algorithm to solve many difficult engineering problems as well as many optimization problems with large, complex search spaces. However, the use of genetic algorithm in the field of reliability engineering is quite limited [12]-[15].

Different from the previous studies, this paper uses the genetic algorithm in maximizing the two terminal reliability subjected to some total cost of interconnection networks while all the nodes are least connected with each other. In this regard, a new method based on Genetic Algorithm is proposed in Section 3; where a different and efficient chromosome as well as gene encoding and cross over is used for better convergence towards the optimal solution. The rest of the paper is organized as follows.

The application of the proposed method on some sample interconnection networks is presented in Section 4. Section 5 concludes the paper with future scope.

2. Formulation Of Optimization Problem

In this section, Notations and Assumptions are first introduced; then the targeted constrained optimization problem is formulated.

Notations	
ICN	Completely connected interconnection network.
G	Probabilistic Graph of the network
N	Number of nodes
L	Number of Links
s , t	Source node and destination node respectively
Cross_mat	Cross over matrix
Cost_mat	Cost of the links of the network
Rel_mat	Reliability of the links of the network
$Cost_{max}$	Maximum permissible Cost/Budget Constraint
T_i	Minimal Spanning Tree i .
ST1 and ST2	Selected s to t paths for gene level cross over
Child1, Child2	Paths generated after gene level cross over
P1, P2	Selected two networks for chromosome level cross over
C1 and C1	Generated networks after chromosome level cross over
C_{ij}	Cost of the link connecting node i and j
R_{ij}	Reliability of the link connecting node i and j
x_{ij}	A decision variable.(1 if a link exist between i and j, 0 otherwise)
K	The set of nodes having degree less than 2.
$\|K\|$	The number of nodes in K set.

Assumptions

1. Each C_{ij} and R_{ij} is fixed and known.
2. Nodes are perfectly reliable.
3. Links are either operational or failed.
4. Failure of links are s -independent

2.1 Problem Statement

An interconnection network can be viewed as an equivalent probabilistic graph G (N, L) where, N represents the nodes (Stations, terminals or computer sites), while L is the communication links among them. If the nodes of the network are fixed, then the main design decision is to maximize the reliability connecting the source vertex s to the destination vertex t satisfying the minimum degree constraint of 2 under maximum permissible cost ($Cost_{\max}$) constraint.

Objective function
Maximize $R(G)$

Subject to $$\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} C_{ij}\, x_{ij} \le Cost_{\max}$$

and degree of each node of $G \ge 2$

3. Proposed Method

3.1 Definitions:

Definition 1. Spanning Tree is a maximal set of edges of G that contains no cycle, or as a minimal set of edges that connect all vertices.

Definition 2. Minimum Spanning Tree is a spanning tree with weight less than or equal to the weight of every other spanning tree.

Definition 3. A steady state is defined as the state of a system that does not change over time or in which any one change is continually balanced by another.

3.2. Encoding

For a network with N number of nodes and L number of links, the length of the chromosomes is the number of nodes of the network and the chromosomes are encoded by a sequence of 1's and 0's where 1 represents a direct connection from the current node to its neighbors while 0 represents no such direct connections. As in general the number of nodes of an interconnection network is less than the number of links, the length of the chromosomes by using the proposed encoding method is very less than that using[14][18]. A gene is a sequence of chromosomes such that there exist a path from the source node s to destination node t.

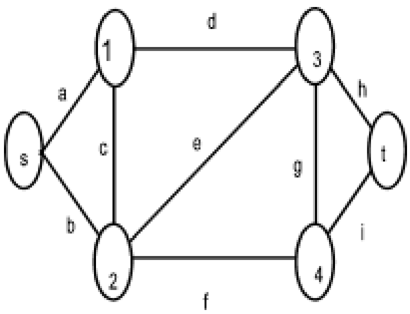

Fig-1. An example Interconnection Network

Example:
Chromosome Encoding of the example Interconnection Network (Fig-1) is

Node s 011000
Node 1 101100
Node 2 110110
Node 3 011011
Node 4 001101
Node t 000110

Gene encoding

011000 101100 110110 001101 000110 (s -1-2-4- t)
011000 110110 011011 000110 (s -2-3- t)

3.3 Initial Population

The initial population Cross_mat is generated in 3 phases.

1) Find two minimal spanning trees [17] T_1 and T_2 of ICN on the basis of Cost to Reliability ratio and $(1-R)$.
2) Generate all the paths p_i, i=1,2,… from s to t of ICN.
3) Merge each p_i with T_1 and T_2 separately till the cost constraint of the network is not satisfied. Add the resultant network into Cross_mat.

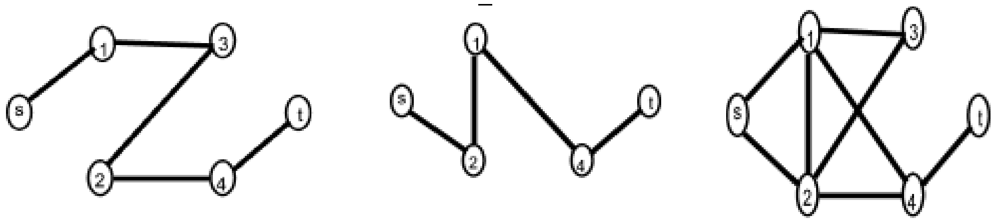

Fig-2. (a) Spanning Tree ; (b) $s \sim t$ path ; (c) *Spanning Tree after merge with* $s \sim t$ *path*

3.4 Gene level Cross Over

The gene level cross over is made to generate $s \sim t$ paths of the network. It makes the convergence rate of the algorithm faster. The gene level cross over is carried on ST1 and ST2 in four steps.

1) Perform a single point cross over at any common node so that it ensures the resultant child paths Child1 and Child2 are also $s \sim t$ paths.
2) Select best two paths on the basis of minimum cost to reliability ratio among ST1, ST2, Child1 and Child2 and merge with each of the candidate networks of the population until the cost constraint $Cost_{max}$ is not reached.
3) Eliminate the cycle(s) from the path (if any).
4) Add each merged networks into the population for next generation.

Example:

Gene level cross over

ST1 : 011000 101100 110110 001101 000110 (s -1-2-4- t)

ST2: 011000 110110 011011 000110 (s -2-3- t)

Cross over at node 2 resulting child nodes as:
Child1:011000 101100 110110 011011 000110 (s -1-2-3- t)
Child2:011000 110110 001101 000110 (s -2-4- t)

Cycle Elimination:
If the resultant gene is { s -1-2-3-4-2-7- t }, then eliminating the outer cycle will produce the resultant gene as { s -1-7- t }.

3.5 Chromosome level Cross Over

A random single point cross over is applied on the chromosome P1 and P2 to obtain the child networks C1 and C2 which is exemplified below.

Network1:

0	0	1	1	0	0
0	0	1	1	1	0
1	1	0	0	1	1
1	1	0	0	1	0
0	1	1	1	0	0
0	0	1	0	0	0

Network2:

0	1	0	0	1	1
1	0	1	0	1	0
0	1	0	1	0	1
0	0	1	0	0	1
1	1	0	0	0	0
1	0	1	1	0	0

After Cross Over:

Network C1:

0	0	1	1	1	1
0	0	1	1	1	0
1	1	0	0	0	1
1	1	0	0	0	1
0	1	1	1	0	0
0	0	1	0	0	0

Network C2:

0	1	0	0	0	0
1	0	1	0	1	0
0	1	0	1	1	1
0	0	1	0	1	0
1	1	0	0	0	0
1	0	1	1	0	0

3.6 Check_For_Validity Operator

The resultant chromosomes C1 and C2 after cross over may lead to some invalid networks. A check is made to validate the child networks as well as to adjust the bits of some chromosomes so that a valid network can be formed.

The check is made in the following way:

Check the chromosome bit pattern from node 1 to N.

i) If the ith bit of jth node is 1 then,
the jth bit of the ith node is set to 1 (if it is not).

ii) If the ith bit of jth node is 0 then
the jth bit of the ith node is set to 0 (if it is not).

Example:
The Network C1 and Network C2 after applying Check_For_Validity operator :

Network C1

0	0	1	1	1	1
0	0	1	1	1	0
1	1	0	0	0	1
1	1	0	0	0	1
1	1	0	0	0	0
1	0	1	1	0	0

Network C2

0	1	0	0	0	0
1	0	1	0	1	0
0	1	0	1	1	1
0	1	1	0	1	0
0	1	1	1	0	0
0	0	1	0	0	0

3.7 *Mutation Operator*

A mutation operator is used here to add some additional links to the generated networks so as to satisfy the minimum 2-connectivity criteria (if it does not satisfy). The updating is applied in the following steps:

1- If $|K|$=1 then join that node with a node having least cost.

2- If $|K|$=2 say N_1 and N_2 then

2.1 Compute C_1=The minimum cost between N_1 and any other node N_k of the network.
If N_k=N_2 then join N_1 and N_2.
else

2.2 Compute C_2= The minimum cost between N_2 and any other node N_L of the network.
If N_L=N_1 the join N_1 with N_2
else

2.3 Compute C=Cost between N_1 and N_2
If C< (C_1+C_2) then join N_1 and N_2 otherwise join N_1 with N_k and N_2 with N_L.

3- If $|K| \geq 2$ then

Assign a link from ICN to the existing network by joining any two nodes of K -set with minimum cost.

Example:

The network C1 satisfy the 2-connectivity criteria where as network C2 does not satisfy. So the Mutation operator is applied on C2.

In network C2, node 1 and 6 has degree less than 2, so k={1,6} satisfying step 3 of mutation operator algorithm.

Here N_1=1 and N_2=6. C_1=30 and N_k=N_2, therefore join N_1 and N_2.

Now the repaired network satisfying the minimum 2-connectivity criteria is:

0	1	0	0	0	1
1	0	1	0	1	0
0	1	0	1	1	1
0	1	1	0	1	0
0	1	1	1	0	0
1	0	1	0	0	0

3.8 *Genetic Algorithm*

Input: (ICN, G , Cost_mat, Rel_mat, $Cost_{\max}$, s , t)

1. Generate the initial population.
2. While (the steady state is not reached)

Sl. No.	(N, L)	Cost max	Mutation Rate	Population Size	Computed Cost	Computed Reliability	CPU Time (sec)	Optimal Network
1	(6N,8L)	355	0.0280	72	350	0.9576215	45.064	{{1,4},{1,5},{2,3},2,5},{3,4},{3,6},{4,6},{5,6} }
2	(6N,9L)	380	0.0285	150	365	0.9655204	117.71	{{1,2},{1,5},{2,3},{2,4},{3,4},{3,6},{4,5},{4,6},{5,6}}
3	(6N,11L)	445	0.0440	213	440	0.9670229	192.31	{{1,3},{1,4},{1,6},{2,3},{2,5},{2,6},{2,4},{3,4},{3,5},{4,5},{5,6}
4	(7N,10L)	425	0.0237	155	420	0.9688972	105.47	{{1,2},{1,4},{2,3},{2,5},{3,6},{3,7},{4,6},{4,7},{5,6},{5,7}}
5	(7N,11L)	455	0.0206	108	446	0.9713799	191.79	{{1,3},{1,5},{2,3},{2,4},{2,7},{3,6},{4,6},{4,7},{5,6},{5,7},{6,7}
6	(8N,11L)	470	0.0117	93	450	0.9510128	216.88	{{1,2}, {1,3} ,{2,3}, {2,4}, {2,7}, {3,6}, {4,6}, {4,7},{5,6}, {5,7}, {6,7}}
7	(8N,12L)	511	0.0221	123	500	0.9657424	237.58	{{1,3},{1,8},{2,3},{2,4},{2,5},{3,4},{4,6},{5,6},{5,7},{6,7},{6,8},{7,8}}
8	(8N,13L)	555	0.0216	165	541	0.9889690	548.07	{{1,3},{1,7},{1,8},{2,3},{2,5},{2,6},{2,7},{3,4},{4,5},{4,7},{5,6},{5,8},{6,8}
9	(8N,14L)	625	0.0220	408	605	0.9891628	1701.3	{{1,3},{1,4},[1,8},{2,3},{2,5},{2,6},{2,8},{3,5},{4,6},{4,7},{5,7},{5,8},{6,7},{6,8}}
10	(9N,12L)	465	0.0086	159	425	0.9693728	262.78	{{1,2},{1,9},{2,8},{3,4},{3,7},{3,9},{4,5} {4,6},{5,7},{5,8},{6,7},{6,8}}
11	(9N,14L)	571	0.0226	264	532	0.0972199	2201.2	{{1,2},{1,3},{2,4},{2,6},{3,5},{3,6},{4,7},{4,9},{5,6},{5,8},{6,8},{7,9},{7,8},{8,9}}
12	(9N,15L)	603	0.0091	498	580	0.9867127	1116.6	{{1,2},{1,3},{1,4},{2,8},{2,9},{3,5},{3,6},{4,5},{4,9},{5,7},{6,7},{6,8},{6,9},{7,8},{8,9}}
13	(10N,13 L)	600	0.0038	1788	550	0.9718334	7323.6	{{1,2},{1,10},{2,3},{3,4},{3,9},{4,5},{4,8},{5,6 },{5,7},{6,7},{7,8},{8,9},{9,10}}
14	(10N,14 L)	560	0.0020	381	545	0.9563464	859.24	{{1,2},{1,3},{2,4},{2,9},{3,5},{3,10},{4,5},{4,7 },{5,8},{6,7},{6,8},{6,10},{8,9},{9,10}}

Table-I. Application of proposed method on some sample Networks

2.1 Select any two *s~t* paths ST1 and ST2 from ICN within the cost constraint.

2.2 Perform gene level cross over on ST1 and ST2.

2.3 Pop two networks P1 and P2 from the population pool.

2.4 Apply chromosome level cross over on P1 and P2 to obtain the child networks C1 and C2.

2.5 Apply Check_For_Validity operator on C1 and C2.

2.6 Apply mutation operator on C1 and C2.

3. Compute the reliability using the method [16] of each chromosome of Cross_mat and find the chromosome with maximum reliability.

4. Result and Discussion

The algorithm is implemented on some sample networks [15] and the results are recorded along with the CPU time. The Source node (s) is 1 and destination node (t) is 6 in all the networks.

The reliability of some benchmark networks are estimated by using the proposed method. The reliability of the interconnection networks having up to 10 numbers of nodes are estimated under different cost constraint. The cost and reliability of the links are set as per [15]. The different mutation rates taken during estimating the reliability of the networks are within a range from 0.002 to 0.044, which is in average smaller than the maximum value of mutation rate, which ensures the preservation of the original network. The cost of the network thus computed by the proposed method is also found to be less than the maximum predefined cost. The proposed method estimates a reliability value of 0.97 within 191 cpu time for a network having 7 number of nodes and 11 number of links within a maximum cost of 455. For network with 10 number of nodes and 13 number of links, it can be observed from the table that the proposed method generates an initial population of 1788 and evaluates it's reliability of 0.97 within 7323 cpu time. From the above table and the different observations as discussed earlier, it can be ensure that the proposed method estimates a good measure of reliability value by generating reasonable size of initial population within very few cpu ses.

5. Conclusion

The genetic algorithm returns an optimal or near optimal solutions on every run regardless of problem instance, problem size, or random number seed. This paper presents a genetic algorithm based approach to addresses a topological optimization problem of interconnection network to optimize terminal reliability with some predefined cost constraint. The numerical results have shown that the genetic algorithm can solve this complicated problem. The algorithm also confirmed the efficiency to produce good solutions even for large-sized networks. The algorithm uses two level cross over for better convergence rate. The cross over points are randomly selected and an operator (Check_For_Validity) is used to repair the invalid generated networks. An efficient mutation operator is implemented to ensure a 2-connected interconnection network. The proposed algorithm is applied on various different kinds of interconnection networks and found to be efficiently computing the reliability of the given interconnection network in few secs. The proposed algorithm can be used in all types networks like communication network, distributed access network, computer network etc.. The method is quite flexible and can be used for alternative objectives (eg. minimize the cost subject to reliability constraint). The chromosome and gene encoding, cross over, the operator Check_For_Validity, and the special mutation operator always ensures producing a feasible 2-connected network topology. Solving a network design problem of additional complexity by including more number of objectives and constraints is an area of further investigation.

References

[1] R. R Boorstyn and H. Frank, Large-scale network topological optimization, IEEE Transaction Communication, 25(1), (1977), 29-47.

[2] R. -H. Jan, Design of reliable networks, Computer and Operation Research, 20(1), (1993), 25-34.

[3] K. K. Aggarwal, Y. C. Chopra , and J. S. Bajwa, Topological layout of links for optimizing the overall reliability in a computer communication system, Microelectronics and Reliability, 22(3), (1982), 347-351.

[4] Fang-Ming Shao, Xuemin(Sherman) Shen, and Pin-Han Ho, Reliability optimisation of Distributed Access Networks With Constrained Total Cost, IEEE Transactions on Reliability, 54(3), (2005), 421-430.

[5] B. A. Coan, W. E. Leland, M. P. Vecchi, A.Wwnrib, and L. T. Wu, Using distributed topology update and preplanned configurations to achieve trunk network survivability, IEEE Transactions on Reliability, 40(4), (1991), 404-416.

[6] S. Pierre, M.-A. Hyppolite, J. –M. Bourjolly, and O. Dioume, Topological design of computer communication networks using simulated annealing, Eng. Applicat. Artif. Intell., 8(1), (1995), 61-69.

[7] C. Rose, Low mean internodal distance network topologies and simulated annealing, IEEE Transaction on Communications, 40(8), (1992), 1319-1326.

[8] P. J. M. Van Laarhoven, and E. H. L. Aarts, Simulated Annealing: Theory and Applications: Dordrecht, Holland: D. Reidel, 1987.

[9] Hosam M. F. AboElFotoh and Loulwa S. Al-Sumait, A neural Approach to Topological Optimization of Communication Networks, with Reliability Constraints, IEEE Transactions on Reliability, 50(4), (2001), 397-408.

[10] F. Altiparmak, B. Dengiz and A. E. Smith, Reliability Estimation of Computer Communication Networks: ANN Models, Proc. Eighth IEEE international Symposium on Computers and Communication (ISCC'03), (2003).

[11] M. K. M. Ali and F. Kamoun, Neural Networks for shortest path computation and routing in computer networks, IEEE Transactions on Neural Networks, 4(6), (1993), 941-955.

[12] D.W. Coit and A.E.Smith, Reliability Optimization of Series–Parallel Systems using a Genetic Algorithm, Computers and Operation Research, 23, (1996), 515-526.

[13] L.A.Cox, L. Davis, and Y. Qiu, Dynamic anticipatory routing in circuit-switched telecommunication networks, handbook of Genetic Algorithms: New York: Van Nostrand Reinhold, 1991.

[14] B. Dengiz, F. Altiparmak, and A. E. Smith, Local Search Genetic Algorithm for Optimal Design of Reliable Networks, IEEE Transactions on Reliability, 1(3), (1997), 179-188.

[15] B. Dengiz, F. Altiparmak, and A. E. Smith. Efficient Optimization of All-Terminal Reliable Networks, Using an Evolutionary Approach, IEEE Transactions on Reliability, 46(1), (1997), 18-26.

[16] P. K. Tripathy, R. K. Dash and C. R. Tripathy. A self generating disjoint minimal cut-set method for reliability evaluation of Interconnection networks. International Conference on Signal Processing and Communications (SPCOM), (2010),.1-5.

[17] T. H. Cormen, C. E. Leiserson and R. L. Rivest, Introduction to Algorithms: MIT Press, 1990.

[18] Sheng-Tzong Cheng, Topological Optimization of a Reliable Communication Network, IEEE Transactions on Reliability, 47(3), (1998), 225-233.

[19] A. Kumar, R. M. Pathak, and Y. P. Gupta, Genetic-algorithm-based reliability optimization for computer network expansion, IEEE Transactions on Reliability, 44(1) , (1995), 63–72.