

A GENETIC ALGORITHM APPROACH TO OPTIMAL TOPOLOGICAL DESIGN OF ALL TERMINAL NETWORKS

BERNA DENGIZ AND FULYA ALTIPARMAK

Department of Industrial Engineering
Gazi University, Ankara, TURKEY 06570

ALICE E. SMITH¹

Department of Industrial Engineering
University of Pittsburgh, Pittsburgh, PA 15261

ABSTRACT:

In the design of communication networks, one of the fundamental considerations is the reliability and availability of communication paths between all terminals. Together, these form the network system reliability. The other important aspect is the layout of paths to minimize cost while meeting a reliability criterion. In this paper, a new heuristic search algorithm based on Genetic Algorithms (GA) is presented to optimize the design of large scale network topologies subject to a reliability constraint. The search works with an improved Monte Carlo simulation technique to estimate the system reliability of a network topology.

INTRODUCTION

An important part of network design is to find the best way to layout the components (nodes and arcs) to minimize cost while meeting a performance criterion such as transmission delay, throughput or reliability. This design stage is called "Network Topological Optimization". In a topological network design problem, a main concern is to design networks which operate effectively and without interruption in the presence of component failures. Reliability is concerned with the ability of a network to carry out desired network operations.

Generally, a large scale network has a multilevel, hierarchical structure consisting of a backbone network and several local access networks (Boorstyn et al, 1977). Therefore, designing the topology of a large scale network can be divided into two problems, the backbone network design and the local network design. This study is mainly interested in large scale backbone network design. For backbone network design, an important connectivity measure is reliability. In a communication network, *all terminal* network reliability can be defined as the probability that every pair of nodes can communicate with each other (Jan et al, 1993; Colbourn, 1987).

Many studies have considered topological optimization with a network reliability criterion. For example, Jan et al (1993) used a decomposition method based on branch and bound to minimize total network cost under a system reliability

¹ Corresponding author.

constraint. Their method can only solve small networks because as the number of arcs increases, the number of possible layouts grows faster than exponentially. Because of this complexity, other existing methods are not computationally feasible for designing large scale network topologies with very confining assumptions. Therefore, a heuristic search algorithm based on Genetic Algorithms (GAs) is developed to find a network topology which has minimum cost, subject to a system reliability constraint.

A computer communication network can be modeled by a probabilistic graph $G = (N, L, p)$, in which N and L are the set of nodes and arcs that corresponds to the computer sites and communication links, respectively. The networks considered in this paper are assumed to have bi-directional links and therefore are modeled by graphs with non-oriented edges. We further assume that the graph under discussion has no redundant arcs. Any graph $G = (N, L)$ is said to be connected if there is at least one path between every pair of nodes. A sub-graph G_1 of G is a graph, of which all nodes and arcs are contained in G . i.e., $G_1 = (N_1, L_1)$ where $N_1 \subseteq N$ and $L_1 \subseteq L$. If $N_1 = N$, the sub-graph G_1 is called a “spanning sub-graph”. In a connected graph G of l arcs and n nodes, a tree T is a spanning tree consisting of $n-1$ arcs. The deletion of any edge from a tree results in a disconnected graph. Therefore a connected graph should be at least a spanning tree with $n-1$ edges. A communication network topology should be at least a spanning tree and communication network reliability must be greater than the required system reliability value, p_0 .

In addition to a simple network connectivity check (i.e., does a minimum spanning tree exist in the network), Roberts and Wessler (1970) proposed a “2-connectivity” measure in the design of communication network topologies. “2-connectivity” means that there are at least two paths between each pair of nodes, rather than one. In the literature, many studies considered this measure to be a reasonable constraint of reliability in the design of network topology. In our study, it is used to establish the initial population and to constrain subsequent populations. Therefore, the final network design will meet the system reliability constraint and contain at least two different paths between all pairs of nodes.

Under the following assumptions: (1) the location of each network node is fixed and given, (2) each c_{ij} and the p are fixed and known, where c_{ij} is the cost of link in the network between nodes i and j , and p, q are link reliability and unreliability ($p + q = 1$), (3) each link is bi-directional, (4) there are no redundant links in the network, and (5) the probability of failure of a link is independent of the states of the other links, the main problem can be stated mathematically as follows:

$$\text{Minimize } z = \sum c_{ij} x_{ij}$$

$$\text{Subject to : } f(x) \geq p_0$$

$x_{ij} \in \{0,1\}$ are the decision variables and $f(x)$ is the network reliability. The all terminal system reliability of a network is defined to be the probability that every pair of nodes can communicate with each other. At any instant of time, only some arcs of G may be operational. A state of G is a sub-graph (N, L') with $L' \subseteq L$, where L' is the set of operational arcs. An operational state is generally called a pathset, and a minimal operational state is a min-path. A failed state L' is called $L \setminus L'$ (a cutset) and when L' is a maximal failed state, $L \setminus L'$ is a min-cut (Colbourn et al, 1991). The reliability of G , $Rel_K(G)$, is the k -terminal reliability: If $K=N$, this is the all

terminal reliability, $Rel(G)$. It is easy to formulate a network in state $L' \subseteq L$, with reliability as follows:

$$\prod_{e \in L'} p_e \prod_{e \in (L \setminus L')} q_e \quad \text{where } L' \text{ is the set of operational arcs.}$$

Summing this state occurrence probability over all operational states gives the network system reliability. There are basically two approaches to network system reliability calculation; simulation and analytic. All known analytic methods have worst case computation times which grow exponentially with the size of the network (Aggarwal et al, 1982; Nakazawa, 1981; Cavers, 1975; Rai, 1982; Aggarwal and Rai, 1981). Monte Carlo simulation methods, for which computation time grows only slightly faster than linear with network size, have been the method of choice for more than trivial sized networks. In this research, we used a Monte Carlo simulation technique to predict the network reliability which substantially reduces the variance of the estimator when compared to “crude” Monte Carlo (Yeh et al, 1994). This reduced variance Monte Carlo is based on a two tiered hierarchical approach to sampling which makes use of how many arcs fail during a given simulation.

SOLUTION APPROACH BASED ON GENETIC ALGORITHMS

A GA is developed as a solution methodology for network topological optimization with a reliability constraint. GAs were pioneered by Holland (1975) and Goldberg (1989) for continuous non-linear optimization, and later extended (Cohon et al, 1991; Biegel et al, 1990; Muhlenbein et al, 1988) for combinatorial problems. In GA, the search space is composed of possible solutions to the problem, each represented as some convenient data structure, referred to as the chromosome. Each chromosome has an associated objective function value, called the fitness value. A good chromosome is the one that has a high fitness value. A set of chromosomes together with their associated fitness values is called the population. This population, at a given stage of the GA, is referred to as a generation.

In a conventional GA, candidate solutions are represented by strings of numbers using a binary or non-binary alphabet. The present algorithm uses a binary coding structure for representing candidate solutions. A binary set is used to represent arcs, where the maximum number of non-redundant, undirected arcs for a network of n nodes is given by $(n-1)n/2$. For example, a simple network whose base graph consists of 5 nodes and 10 possible links can be represented by:

$$[1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1]$$

$$[x_{12}, x_{13}, x_{14}, x_{15}, x_{23}, x_{24}, x_{25}, x_{34}, x_{35}, x_{45}]$$

where x_{ij} represents a link connecting two nodes i and j . If x_{ij} is equal to 1, there is a connection between these two nodes. If x_{ij} is equal to 0, then there is no connection.

The initial population which consists of a set of feasible solutions (2-connected networks) is generated in a random fashion. For determining this initial population, a number of experiments were carried out. A candidate network consists of some randomly selected arcs between nodes. The selection of the probability values which are used in deciding whether an arc exists or not was an important step to generate the initial population. In an experimental design with 10, 20 and 30 nodes, the following characteristics were systematically controlled.

- i) Arc probabilities between $[0,1]$, which determines the existence of an arc between nodes, are selected,

- ii) The system reliability value of each connected network is estimated using Monte Carlo simulation,
- iii) The probability values of the existence of arcs and the corresponding network reliability values are compiled.

The aim was to determine the intervals of the probability values which result highly reliable networks. Any initial population can then be generated by using probabilities within these intervals. Table 1 shows the resulting probability intervals from the experiments described above which were used for the initial populations.

Table 1. Probability values used to generate the initial population.

Number of Nodes (<i>n</i>)	Probability of an Arc
10	(0.15-0.60)
20	(0.15-0.50)
30	(0.10-0.30)

The choice of parameters for GAs can have a significant effect on performance of the algorithm. Parameter values were investigated by running the GA with different population sizes (10, 20, 30), crossover rates (0.55, 0.65, 0.75, 0.85, 0.95) and mutation rates (0.01, 0.05, 0.09, 0.10). It was found that the best results were: population size = 20, crossover rate = 0.95 and mutation rate = 0.05.

The objective function is the sum of the total cost for all arcs in the network plus a quadratic penalty function, which is applied when the network reliability prediction does not meet the network reliability requirement (i.e., infeasible). The objective of the penalty function is to lead the optimization algorithm to feasible solutions. It was important to allow infeasible solutions into the population because good solutions are often the result of breeding between feasible and infeasible solution (Coit and Smith, 1994; Smith and Tate, 1993). The objective function is,

$$Z = \sum_i \sum_j c_{ij}x_{ij} + \delta(\epsilon(\text{Rel}(G) - p_0))^2, \quad i = 1, \dots, n-1 \quad ; \quad j = i+1, \dots, n$$

where c_{ij} , x_{ij} and p_0 were previously defined, $\text{Rel}(G)$ equals $f(x)$ (network reliability), ϵ is the maximum value of c_{ij} and $\delta = 0$ if $\text{Rel}(G) \geq p_0$ and $\delta = 1$ if $\text{Rel}(G) < p_0$. The fitness is chosen to be $(Z_{\max} - Z)$ where Z_{\max} is a constant, which is the largest penalized cost of all networks in the current population. This subtraction translates the minimization problem to a maximization problem.

The reduced variance Monte Carlo estimation of system reliability (Yeh et al, 1994) is used to minimize computational effort. To further speed up the search, Jan's upper bound formulation (Jan, 1993) of network reliability is used. If this upper bound reliability value exceeds the required system reliability value, then the Monte Carlo simulation is used as a subroutine. Otherwise, the candidate network is considered to be infeasible. While it is possible that some networks which are truly feasible are discarded at this point, the probability of this occurring is very small. Use of the upper bound considerably reduces the number of network requiring simulation.

Roulette wheel selection is used for each generation of our algorithm. In this mechanism, a candidate network is selected with probability equal to its relative

fitness with respect to the whole population. Classic crossover and mutation operators (Goldberg, 1989) are used to obtain the new candidate networks for the next population. After crossover and mutation, new candidate networks are checked for connectivity using the “Set Merging Algorithm” (Hopcroft and Ullman, 1973). Then all new candidate networks replace their parents. Additionally, an elitist strategy appends the best performing candidate network of the previous generation to the current population. This strategy ensures that the candidate network with the best objective function value always survives to the next generation.

A GA continues until a pre-determined stopping criterion has been met. The criterion is often based on the total number of generations. Our termination generation is determined according to the size of the network under study.

A summary of the GA is as follows,

- 1:** Generate the initial population randomly using arc probabilities based on the number of nodes, n . Ensure that each network satisfies the 2-connectivity rule. Calculate the fitness value of each feasible network in the population.
- 2:** Select two networks from current population using selection mechanism.
- 3:** To obtain two children candidate networks, apply the genetic operators to the selected networks from Step 2.
- 4:** Control the connectivity of each new candidate network. If the network is not connected then discard and go to Step 2, else calculate the upper bound of the network reliability using Jan’s procedure. If this upper bound is greater than or equal to the required system reliability value, then estimate the network reliability value (Rel(G)) using Monte Carlo. Compute the objective function value by summing arc cost and any penalty applied.
- 5:** Use the replacement strategy to create the new population.
- 6:** Check the number of new networks. If this number is less than the population size go to Step 2, else go to Step 7.
- 7:** Calculate the fitness value of each network in the new population.
- 8:** If the termination condition is satisfied then stop, else go to Step 2.

NUMERICAL EXAMPLES

The performance of the algorithm is illustrated using three different problems. The first and the third problems were solved (Jan et al, 1993) using a decomposition method based on branch and bound. That study was the only one that used an analytical method to calculate reliability. There are no analytical results available for fully connected networks with more than 12 nodes. As a second test problem, a network with 7 nodes and 21 arcs was solved using the GA to identify the optimal network topology. The results (Table 2) indicate that the GA finds optimal nearly all the time, typified by the low coefficient of variation over 10 runs of each problem.

Table 2. Comparison Table Between Analytical and GAs Results

	# Nodes	# Arcs	p	p_0	Optimal Cost	Results of GA		
						Best Cost	Average Cost	Coeff. of Variation
Prob.1	5	10	0.80	0.9	255	255	255	0
Prob. 2	7	21	0.90	0.9	720	720	720	0
Prob. 3	20	30	0.95	0.9	596	596	598.7	0.0052

CONCLUSIONS

In this study, a heuristic search algorithm based on GAs was developed to solve network topology design with minimum cost subject to a reliability constraint. Based on the results obtained from the three test problems, it is concluded that complex design problems can be solved with high accuracy.

REFERENCES

- Aggarwal, K. K., Rai, S., (1981). Reliability evaluation in computer communication networks, *IEEE Transactions on Reliability*, R-30 (1).
- Aggarwal, K. K., Chopra, Y. C., Bajwa, J. S., (1982). Reliability evaluation by network decomposition, *IEEE Transactions on Reliability*, R-31 (4), 355-358.
- Biegel, J. E., Davern, J. J., (1990). Genetic algorithms and job shop scheduling, *Computers and Industrial Engineering*, 19 (1-4), 81-91.
- Boorstyn, R. R., Plank, H., (1977). Large scale network topological optimization, *IEEE Transactions on Communications*, Com-25 (1), 29-37.
- Cavers, J. K., (1975). Cutset manipulations for communication network reliability estimation, *IEEE Transactions on Communications*, Com-23 (6).
- Coit, D. W., Smith, A. E., (1994). Use of genetic algorithm to optimize a combinatorial reliability design problem, *Proceeding of the Third IIE Research Conference*, 467-472.
- Colbourn, C. J., (1987). *The Combinatorics of Network Reliability*, Oxford University Press.
- Cohon, J., Hedge, S. U., Martin, W. N., Richards, D. S., (1991). Distributed genetic algorithms for the floorplan design problem, *IEEE Transactions on Computer Aided Design*, 10 (4), 483-492.
- Goldberg, D. E., (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley.
- Holland, J., (1975). *Adaptation in Natural and Artificial Systems*, U of Michigan Press, Ann Arbor.
- Hopcroft, J., Ullman, J., (1973). Set merging algorithms, *SIAM Journal of Computers*, 2, 296-303.
- Jan, R. H., (1993). Design of reliable networks, *Computers and Operations Research*, 20, 25-34.
- Jan, R. H., Hwang, F. J., Cheng, S. T., (1993). Topological optimization of a communication network subject to a reliability constraint, *IEEE Transactions on Reliability*, 42 (1), 63-70.
- Muhlenbein, H., Schleuter, M. G., Kramer, O., (1988). Evolution algorithms in combinatorial optimization, *Parallel Computing*, 7, 65-85.
- Nakawaza, H., (1981). Decomposition method for computing the reliability of complex networks, *IEEE Transactions on Reliability*, R-30 (3).
- Rai, S., (1982). A cutset approach to reliability evaluation in communication networks, *IEEE Transactions on Reliability*, R-31 (5).
- Roberts, L. G., Wessler, B. D., (1970). Computer network development to achieve resource sharing, *AFIPS Conference Proceedings*, 36. Montvale, NJ: AFIPS Press, 543-599.
- Smith, A. E., Tate, D. M., (1993). Genetic optimization using a penalty function, *Proceedings of the Fifth International Conference on Genetic Algorithms*, 499-505.
- Yeh, M. S., Lin, J. S., Yeh, W. C., (1994). A new Monte Carlo method for estimating network reliability, *Proc 16th International Conference on Computers & Industrial Engineering*, 723-726.