A Genetic Algorithm Approach for the Design of Minimum Cost Survivable Networks with Bounded Rings

B. Ombuki, M. Nakamura, Z. Nakao and K. Onaga
Faculty of Engineering
University of the Ryukyus
Okinawa, 903-0213 Japan
Phone +81-98-895-8715 Fax +81-98-895-8727
E-mail {betty,morikazu}@ads.ie.u-ryukyu.ac.jp

Abstract: We study the problem of designing at minimum cost a two-connected network topology such that the shortest cycle to which each edge belongs does not exceed a given maximum number of hops. This problem is considered as part of network planning and arises in the design of backbone networks. We propose a genetic algorithm approach that uses a solution representation, in which the connectivity and ring constraints can be easily encoded. We also propose a crossover operator that ensures a generated solution is feasible. By doing so, the checking of constraints is avoided and no repair mechanism is required. We carry out experimental evaluations to investigate the solution representation issues and GA operators for the network design problem.
Keywords: two-connected network design, Genetic Algorithm, Combinatorial Optimization

1 Introduction

The problem of how to cost effectively design a network topology so that some specified constraints are met is relevant to many real-world problems such as computer networking [1], telecommunications [2], oil and gas lines [3] and sewage systems. A typical computer network can be represented as a hierarchical structure, integrating two levels: backbone networks dedicated to the delivery of information from source to destination and local access networks.

Two main issues appear in the planning process of computer networks: economy and survivability. Economy refers to the construction cost, which is expressed as the sum of the edge costs. On the other hand, survivability refers to the restoration of service in the event of node or edge failure.

This paper focuses on the topological designing at minimum cost of Survivable two-connected (edge-connected or node-connected) network topologies such that the shortest cycle to which each edge belongs does not exceed a given maximum number of hops (ring constraints). Even though the problem of two-connected network design has been studied, the idea of two-connected network with bounded rings has only been recently studied by Fortz et al. [4]. This is a relevant extension: if a connection is broken, the flow which was routed using such connection needs to be re-routed. The extra ring constraints limit the region of influence of the traffic which is necessary to re-route. We shall henceforth refer to the problem discussed here as the Two-Connected Network With Bounded Rings Problem (2CNBR) as defined in [4].

The network design problem is classified as NP-hard combinatorial optimization problem [3], the search space growing exponentially with the increase in the number of nodes. Finding an optimal solution for this type of problems within reasonable time is computationally intractable. This problem has been studied in the literature, using both exact approaches (usually a variation of branch and bound [4]) and heuristic methods, such as, simulated annealing [2], genetic algorithms (GA) [3], and tabu search [4].

Reference [4] applied a branch and cut method for the 2CNBR and reported that the algorithm is only effective on small instances. Thus he presents a set of constructive heuristics and a Tabu search [4] to solve this problem. We suppose that GAs are also suited to the problem because of their ability to handle large search spaces besides carrying out multi-objective search. However, an important factor that has made application of GA to this problem a challenge is the issue of generating feasible initial population and effective GA operators. GA approaches applied previously to this problem employ repair mechanism.

In encoding the two-connected problem with bounded rings, we introduce a solution encoding representation in which connectivity and ring constraints are easily encoded along the chromosomes which are all feasible solutions. Further, a problem-specific crossover is designed based on this encoding such that checking of constraints and repair mechanism are avoided.

2 Problem Definition

Let $G = (V, E)$ be an undirected graph, where V represents a set of vertices (computer sites), and is E the set of possible edges (connection links) that can make a direct transmission link between two vertices.

Each edge $e = (i, j) \in E$, has a non negative cost C_{ij} and in this paper we assume is equivalent to the Euclidean distances between i and j.

The two-connected network with bounded rings problem (2CNBR) consists in designing a minimum cost network T mathematically formulated as:

$$T = min \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} C_{ij}\alpha_{ij}$$

Subject to

- $\alpha_{ij} \in \{0, 1\}$
- T contains at least two node-disjoint paths or arc-disjoint paths between every pair of nodes, *Two-connectivity constraint.*
- Each edge of T belongs to at least one cycle whose length is bounded by a given constant K,*Ring constraints.*

The following assumptions are made in dealing with this problem.:

- Each node location is given
- Only one bi-directional link is possible between i and j
- Links are either operational or failed
- No link repair is considered
- The failures of links are independent
- Each link cost is fixed and known

3 Proposed GA

This section describes in details our design strategy based on genetic algorithms. The objective is to convert each of the GA chromosomes into a network topology with two diverse paths (edge-connected and node-connected) and then subjecting each of these chromosomes into an evolutionary process until a minimal total connection cost is achieved. The initial population consists of only feasible solutions and the GA operators is applied while maintaining feasibility constraints. The following procedure gives an outline of our algorithm.

3.1 Chromosome Representation and GA Operations

(1) Chromosome Representation
An assumption is made that any possible links given by $L = N(N-1)/2$ (N= Number of nodes) can be used. We adopt a variable length integer chromosomes representation. A chromosome is represented as a connection of cyclic rings. Example 3.1 illustrates an example of a chromosome representation for a 10 node network with

```
procedure NetGA;
begin
  Read cost matrix C(G) and node locations;
  Set GA parameters;
  Generate initial population pre-processing
  POP of size POPSIZE;
  Convert each chromosome to a network
  topology;
  for gen = 1 to MAXGEN do
    Evaluate fitness of the individuals of
    POP;
    Rank the individuals of POP and select
    new population;
    Retain the best individual;
    Apply genetic operators;
  endfor;
end;
```

Figure 1: An outline of NetGA

ring bound = 4. The initial population consists of individuals of this nature. To generate an initial population, we apply a pre-processing function to generate the chromosomes in the initial population.

We investigate three types of initial populations. The first is based on random generation and the two are generated using the pre-processor. The pre-preprocessing strategy is based on a sort of neighborhood construction criteria. The pre-processor carries out a greedy algorithm to generate a chromosome based on shortest distance from the current node to another node to the generate second type of population. However, since this restricts the solution search space, we try to balance between randomness and greedy allocation and thus the pre-processor is applied to generate a chromosome by choosing a neighbor to the current node from a fixed set of neighbors. A chromosome is represented as connection of cyclic rings. In order to generate a feasible chromosome, the pre-processor must satisfy the following conditions.

1. Ring length is less or equal to given bound
2. Every node must appear in at least one ring
3. No parallel rings are allowed

[Example 3.1] This example illustrates a chromosome representation for a network of node size 10 and ring bound, $K = 4$.

$$\overbrace{4\ 5\ 2\ 8\ 4}^{R1}\ \overbrace{6\ 9\ 7\ 1\ 6}^{R2}\ \ldots\ \overbrace{2\ 6\ 3\ 10\ 2}^{R6}$$

For easier understanding, we attach the ring numbers to their respective positions. Otherwise, the ring numbers are not part of the chromosomes. □

(2) Evaluation of chromosome's fitness

The total cost of the edge connections of the network topology developed from a chromosome is returned as the fitness value. That is, an individual x is assigned the fitness value.

$$F(x) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} C_{ij} \alpha_{ij}$$

where C_{ij} is the cost of connecting link (i, j) and $\alpha_{ij} \in \{0, 1\}$ is decision variable.
(3) Reproduction

The roulette wheel selection incorporating the elite strategy is used to generate a new population for the next generation. We investigate the cases where the best chromosome at each generation is allowed to reproduce and when it is only allowed a lifespan until a better offspring is born.

(4) Crossover Operator
We choose randomly two chromosomes to be crossed as parents according to the crossover rate. We proposed a problem specific crossover operator and investigate its performance by comparison with a standard crossover Uniform order crossover and the two points crossover with swapping applied which [5] applied to the three-connected network design.

For every one ring (consisting of genes representing nodes) on a chromosome to be crossed, a "0" or "1" is randomly assigned, making a 0-1 mask pattern of length equal to the number of rings in the respective chromosome. Since the number of rings for two crossing chromosomes may not be necessarily the same, we make a mask pattern-for each crossing parent. Next, the contents of the rings (s) corresponding to a "1" in the mask pattern is copied directly to their respective offsprings.

A depth first search is carried out to determine the ring we call "root net" which forms the starting subnetwork from the copied rings in an offspring. The remaining nodes: not in the root net are assigned to a non-assigned list. By using edges from the other crossing parents and while checking feasibility constraints, the unassigned nodes are used to build up the root net. Once a node which is in a ring that had been copied to the offspring is met from the unassigned, the whole ring containing it is added to the net root naturally. The process completes when all nodes are used to build up a feasible network. At each start of a new ring, an edge must leave from the root net, and return to another node of the root net once the ring bound is satisfied. Figure 3 describes the proposed crossover in details.

4 Experimental Evaluation

Random problem instances with 20 to 50 nodes were generated. We carried out experiments for different bounds for the rings. The GA parameters were: population size of 100/150, generation span of 100/200 and the crossover rate was set at 1.0.

Table 1: Average Connection Cost

Data	n	k	*Cross*	2pcws	UOX
dat22	22	4	471	593	627
dat22	22	8	327	423	430
dat22	22	10	322	407	407
dat26	26	4	643	748	784
dat26	26	8	391	511	501
dat26	26	10	324	448	469
dat30	30	4	786	915	897
dat30	30	8	469	570	586
dat30	30	10	432	545	550
dat30	30	16	380	476	479

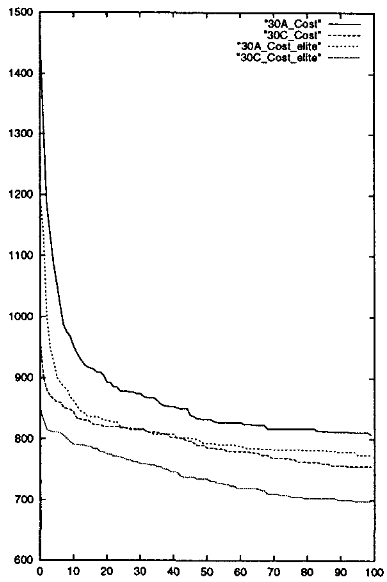

Figure 2: Fitness Value graph for a 30 Nodes Graph

Table 1 reports the size of problem instances given by column labeled n, the ring bound is shown in the column labeled k. Results are given for the averages of 5 runs for the total link connection costs per a network. Costs are given for the three crossovers: proposed crossover in column labeled *cross*, two points crossover with swapping labeled *2pcws* and the uniform order crossover labeled *uox*. *2pcws* was applied in [5] for the three-connected network design problem. Table 1 depicts that the standard crossovers are not effective here. The minimal costs obtained by the proposed crossover shows that a problem specific type of crossover is more suitable for the 2CCNBR.

We carried out experiments to investigate the perfomance of the three types of initial populations described in Section 3. Since the second type of initial population gave results of much poorer quality, we only report those of types one and three. Type A is where the chromosome is generated randomly and produces infeasible chromosomes. Type C we applied a pre-processor as indicated in Section 3. We also investigate the case when the best chromosome (elite) in each generation is allowed to reproduce and when it is allowed to pass on to next generation without alteration and only dies when a better offspring is found. The graph in Figure 2 shows the results obtained for a problem instance of node = 30 and K = 4. The networks produced based on initial population C obtains consists of lower cost than those by random generation. Even though generation type C gives lower costs, we need to improve on it to make it more effective in terms of reducing the costs further and also to speed up convergence rate.

5 Concluding Remarks

This paper discusses a genetic algorithm approach for the two connected problem with bounded rings. A simple GA solution representation method is presented with a crossover operator based on the encoding. The encoding allows the required constraints to be easily encoded long the chromosomes thus producing only feasible chromosomes. Furthermore, no repair mechanism is required after the crossover. Experimental results have shown that the proposed crossover is much more suitable compared to the standard crossovers used. Further work is to improve on the crossover operator and also introducing a hybrid GA that permits another heuristic to be incorporated into the proposed GA to improve on its performance.

References

[1] K.K Aggarwal, Y.C Chopra and J.S Bajwa, "Topological layout of links for optimizing the overall reliability in a computer Communication system," *Micro-electronics and Reliability*, Vol. 22, No.3, pp 347-351.

[2] M.M Atiqullah, S.S Rao, "Reliability optimization of communication networks using simulated Annealing," *Micro-electronics and Reliability*, Vol.33, No.9 , pp 1303-1319, 1993.

[3] G. Walters, D. K. Smith,"Evolutionary design algorithm for optimal layout of tree networks, "*Engineering Optimization*, Vol. 24, pp 261-281, 1995.

[4] Bernad Fortz , "Design of Survivable Networks With Bounded Rings,", *PhD Thesis* Universite Libre de Buxelles, 1999

[5] R. Huang, J. Ma, D. F. Hsu, "A Genetic Algorithm for Optimal 3-Connected Telecommunication Network Designs, "*Proceedings of the 1997 International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN'97)*, pp 344-350, 1997.

procedure Crossover;
begin
  select two chromosomes P_1 and P_2;
  let P_1 be $R_1^1 \mid R_1^2 \mid \cdots \mid R_1^{l_1}$;
  Generate a binary mask-pattern M of length l_1;
  $C_1 := \emptyset$;
  for $i := 1$ to l_1 **do**
    if $M(i) = 1$ **then**
      $C_1 := C_1 \mid R_1^i$;
  Find largest connected component, L_1 in C_1 by Depth First Search(DFS);
  for $i := 1$ to $NODES$ do
    if $i \in L_1$ **then**
      $Root\_Net[i] := TRUE$;
    else
      $L\_Not := L\_Not \cup \{i\}$;
  while $L\_Not \neq \emptyset$ **do begin**
    choose c randomly such that
    $Root\_Net[c] := TRUE$;
    $R := \emptyset$;
    while $length(R) \ll Bound$
    do begin
      find j such that (c, j) is an edge in
      P_2 by some criteria;
      if such j can be found **do**
        $R := R \mid (c, j)$;
        $Root\_Net[j] := 1$;
        make $TRUE$ in $Root\_Net$ all nodes connected to j;
        delete j from L_Not;
        delete from L_Not all nodes
        connected j or in a ring containing j
      else
        find j from L_Not such that (c, j)
        has minimum cost;
        $R := R \mid (c, j)$;
        $Root\_Net[j] := 1$;
        make $TRUE$ in $Root\_Net$ all nodes connected to j;
        delete j from L_Not;
        delete from L_Not all nodes
        connected j or in a ring containing j
    endwhile
      Let c be first node in ring R;
      $R := R \mid (j, c)$;
      $C_1 := C_1 \mid R$;
  endwhile
  By exchanging the roles of P_1 and P_2 each other, C_2 is constructed likewise;
end

Figure 3: Procedure NetCrossover