数理解析研究所講究録 1205 巻 2001 年 148-153

Topological Optimization Models for Communication Network with Multiple Reliability Goals

Baoding Liu[a] & Kakuzo Iwamura[b]
城西大学(理)数学 岩村覚三
[a] *Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China*
[b] *Department of Mathematics, Josai University, Sakado, Saitama 350-02, Japan*

Abstract

Network reliability models for determining optimal network topology have been presented and solved by many researchers. This paper presents some new types of topological optimization model for communication network with multiple reliability goals. A stochastic simulation-based genetic algorithm is also designed for solving the proposed models. Some numerical examples are finally presented to illustrate the effectiveness of the algorithm.

Keywords: network reliability, stochastic programming, genetic algorithm, simulation

1 Topological Optimization Models

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{P})$ be a communication network in which $\mathcal{V}$ and $\mathcal{E}$ correspond to terminals and links, and $\mathcal{P}$ is the set of reliabilities for the links $\mathcal{E}$. If there are n vertices (terminals), then the links $\mathcal{E}$ may also be represented by the link topology of $\boldsymbol{x} = \{x_{ij} : 1 \le i \le n-1, i+1 \le j \le n\}$, where $x_{ij} \in \{0, 1\}$, and $x_{ij} = 1$ means that the link (i, j) is selected, 0 otherwise.

If we assume that the terminals are perfectly reliable and links fail s-independently with known probabilities, then the success of communication between terminals in subset $\mathcal{K}$ of $\mathcal{V}$ is a random event. The probability of this event is called the $\mathcal{K}$-terminal reliability, denoted by $R(\mathcal{K}, \boldsymbol{x})$, when the link topology is $\boldsymbol{x}$. A network $\mathcal{G}$ is called $\mathcal{K}$-connected if all the vertices in $\mathcal{K}$ are connected in $\mathcal{G}$. Thus the $\mathcal{K}$-terminal reliability

$$R(\mathcal{K}, \boldsymbol{x}) = \Pr\{\mathcal{G} \text{ is } \mathcal{K}\text{-connected with respect to } \boldsymbol{x}\}. \tag{1}$$

Notice that when $\mathcal{K} \equiv \mathcal{V}$, the $\mathcal{K}$-terminal reliability $R(\mathcal{K}, \boldsymbol{x})$ is the overall reliability.

In addition, for each candidate link topology $\boldsymbol{x}$, the overall cost should be $\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} c_{ij} x_{ij}$, where c_{ij} is the cost of link (i, j), $i = 1, 2, \cdots, n-1$, $j = i+1, i+2, \cdots, n$, respectively.

We want to minimize the total cost subject to multiple reliability constraints, then we have

$$\left\{\begin{array}{l} \min \displaystyle\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} c_{ij}x_{ij} \\ \text{subject to:} \\ \qquad R(\mathcal{K}_k, \boldsymbol{x}) \geq R_k, \ \ k = 1, 2, \cdots, m \end{array}\right. \tag{2}$$

where $\mathcal{K}_k$ are target subsets of $\mathcal{G}$, R_k are predetermined minimum reliabilities called confidence levels, $k = 1, 2, \cdots, m$, respectively. This is clearly a type of chance-constrained programming.

2 $\mathcal{K}$-terminal Reliability

After a link topology $\boldsymbol{x}$ is given, we should estimate the $\mathcal{K}$-terminal reliability $R(\mathcal{K}, \boldsymbol{x})$ with respect to some prescribed target set $\mathcal{K}$. Estimating $\mathcal{K}$-terminal reliability has received considerable attention during the past two decades. It is almost impossible to design an algorithm to compute $R(\mathcal{K}, \boldsymbol{x})$ analytically. In order to handle larger network, we may employ the stochastic simulation (Monte Carlo simulation) which consists of repeating s-independently N times trials.

Step 1. Set counter $N' = 0$;

Step 2. Randomly generate an operational link set $\mathcal{E}'$ from the link topology $\boldsymbol{x}$ according to $\mathcal{P}$;

Step 3. If $(\mathcal{V}, \mathcal{E}')$ is $\mathcal{K}$-connected, then $N'++$;

Step 4. Repeat the second and third steps N times;

Step 5. $R(\mathcal{K}, \boldsymbol{x}) = N'/N$.

In Step 3 we have to check if the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}')$ is $\mathcal{K}$-connected. In fact, the n-node graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}')$ can be described by its adjacency matrix, which is the $n \times n$ matrix $A = (a_{ij})$ with entries

$$a_{ij} = \begin{cases} 1, & \text{if the link } (i,j) \in \mathcal{E}' \\ 0, & \text{otherwise.} \end{cases}$$

Let I be an $n \times n$ unit matrix, and t be the smallest integer such that $2^t \geq n - 1$. If all entries a'_{ij} of $(I + A)^{2^t}$ are positive, then the graph $\mathcal{G}$ is connected. Moreover, if the entry a'_{ij} of $(I + A)^{2^t}$ is positive for any given indexes $i, j \in \mathcal{K}$, then the graph $\mathcal{G}$ is $\mathcal{K}$-connected.

3 Stochastic Simulation-based Genetic Algorithm

In this section, we present a stochastic simulation-based genetic algorithm for solving the topological optimization models for communication network reliability.

3.1 Representation Structure

Now we use an $n(n-1)/2$-dimensional vector $V = (y_1, y_2, \cdots, y_{n(n-1)/2})$ as a chromosome to represent a candidate link topology $\boldsymbol{x}$, where y_i is taken as 0 or 1 for $1 \leq i \leq n(n-1)/2$. Then the relationship between a link topology and a chromosome is

$$x_{ij} = y_{(2n-i)(i-1)/2+j-i}, \quad 1 \leq i \leq n-1, \ i+1 \leq j \leq n. \tag{3}$$

3.2 Initialization Process

We set y_i as a random integer from $\{0, 1\}$, $i = 1, 2, \cdots, n(n-1)/2$, respectively. If the generated chromosome $V = (y_1, y_2, \cdots, y_{n(n-1)/2})$ is proven to be feasible, then it is accepted as a chromosome, otherwise we repeat the above process until a feasible chromosome is obtained. We may generate *pop_size* initial chromosomes $V_1, V_2, \cdots, V_{pop\_size}$ by repeating the above process *pop_size* times.

3.3 Evaluation Function

The evaluation function, denoted by $eval(V)$, assigns a probability of reproduction to each chromosome V so that its likelihood of being selected is proportional to its fitness relative to the other chromosomes in the population, that is, the chromosomes with higher fitness will have a greater chance of producing offspring through roulette wheel selection.

Let $V_1, V_2, \cdots, V_{pop\_size}$ be the *pop_size* chromosomes in the current generation. At first we calculate the objective values of the chromosomes. According to the objective values, we can rearrange these chromosomes $V_1, V_2, \cdots, V_{pop\_size}$ from good to bad (i.e., the better the chromosome, the smaller the ordinal number). Now let a parameter $a \in (0, 1)$ in the genetic system be given, then we can define the so-called *rank-based evaluation function* as follows,

$$eval(V_i) = a(1-a)^{i-1}, \qquad i = 1, 2, \cdots, pop\_size. \tag{4}$$

We mention that $i = 1$ means the best individual, $i = pop\_size$ the worst individual.

3.4 Selection Process

The selection process is based on spinning the roulette wheel *pop_size* times, each time we select a single chromosome for a new population using *rank-based evaluation function.*

3.5 Crossover Operation

We define a parameter P_c of a genetic system as the probability of crossover. In order to determine the parents for a crossover operation, let us repeat the following process from $i = 1$ to *pop_size*: Generate a random real number r from the interval $[0, 1]$, then the chromosome V_i is selected as a parent if $r < P_c$.

We denote the selected parents as $V_1', V_2', V_3', \cdots$ and split them into the following pairs:

$$(V_1', V_2'), \quad (V_3', V_4'), \quad (V_5', V_6'), \quad \cdots$$

Let us illustrate the crossover operation on each pair by (V_1', V_2'). We denote

$$V_1' = \left(y_1^{(1)}, y_2^{(1)}, \cdots, y_{n(n-1)/2}^{(1)}\right), \qquad V_2' = \left(y_1^{(2)}, y_2^{(2)}, \cdots, y_{n(n-1)/2}^{(2)}\right).$$

First, we randomly generate two crossover positions n_1 and n_2 between 1 and $n(n-1)/2$ such that $n_1 < n_2$, and exchange the genes of V_1' and V_2' between n_1 and n_2, thus produce two children by the crossover operation as follows,

$$\begin{aligned} V_1'' &= \left(y_1^{(1)}, \cdots, y_{n_1-1}^{(1)}, y_{n_1}^{(2)}, \cdots, y_{n_2}^{(2)}, y_{n_2+1}^{(1)}, \cdots, y_{n(n-1)/2}^{(1)}\right), \\ V_2'' &= \left(y_1^{(2)}, \cdots, y_{n_1-1}^{(2)}, y_{n_1}^{(1)}, \cdots, y_{n_2}^{(1)}, y_{n_2+1}^{(2)}, \cdots, y_{n(n-1)/2}^{(2)}\right). \end{aligned}$$

We note that the two children are not necessarily feasible, thus we must check the feasibility of each child and replace the parents with the feasible children.

3.6 Mutation Operation

We define a parameter P_m of a genetic system as the probability of mutation. Similarly with the process of selecting parents for a crossover operation, we repeat the following steps from $i = 1$ to *pop_size*: Generate a random real number r from the interval $[0, 1]$, then the chromosome V_i is selected as a parent for mutation if $r < P_m$.

For each selected parent, denoted by $V = (y_1, y_2, \cdots, y_{n(n-1)/2})$, we mutate it in the following way. We randomly generate two mutation positions n_1 and n_2 between 1 and $n(n-1)/2$ such that $n_1 < n_2$, and regenerate the sequence $\{y_{n_1}, y_{n_1+1}, \cdots, y_{n_2}\}$ at random from $\{0, 1\}$ to form a new sequence $\{y'_{n_1}, y'_{n_1+1}, \cdots, y'_{n_2}\}$. We thus obtain a new chromosome

$$V' = (y_1, \cdots, y_{n_1-1}, y'_{n_1}, \cdots, y'_{n_2}, y_{n_2+1}, \cdots, y_{n(n-1)/2}).$$

Finally, we replace the parent V with the offspring V' if it is feasible. If it is not feasible, we repeat the above process until a feasible chromosome V' is obtained.

3.7 Genetic Algorithm Procedure

We summarize the genetic algorithm for solving the topological optimization models for the communication network reliability as follows.

Input parameters: pop_size, P_c, P_m*;*
Initialize pop_size chromosomes with the Initialization Process;
REPEAT
  Update chromosomes by crossover and mutation operators;
  Compute the evaluation function for all chromosomes;
  Select chromosomes by the sampling mechanism;
UNTIL(*termination_condition*)
  Report the best chromosome as the optimal link topology.

4 Numerical Examples

The computer code for the stochastic simulation-based genetic algorithm to topological optimization models has been written in C language. In order to illustrate the effectiveness of genetic algorithm, a lot of numerical experiments have been done and the result is successful. Here we give two numerical examples performed on a personal computer with the following parameters: the population size is 30, the probability of crossover P_c is 0.3, the probability of mutation P_m is 0.2, the parameter a in the rank-based evaluation function is 0.05. Each simulation in the evolution process will be performed 2000 cycles.

Example 1. Let us consider a 10-node, fully-connected network. Suppose that the cost matrix is

Nodes	1	2	3	4	5	6	7	8	9	10
1	-									
2	30	-								
3	43	26	-							
4	45	76	38	-						
5	50	45	17	35	-					
6	62	25	30	28	15	-				
7	25	46	30	16	25	38	-			
8	15	45	13	20	37	40	36	-		
9	51	15	45	10	34	10	46	42	-	
10	45	25	45	15	37	40	16	24	45	-

We suppose that the reliabilities of all links are 0.90. We also suppose that the total capital available is 250. Thus we have a constraint,

$$\sum_{i=1}^{9} \sum_{j=i+1}^{10} c_{ij} x_{ij} \leq 250.$$

We may set the following target levels and priority structure:

Priority 1: For the subset of nodes $\mathcal{K}_1 = (1, 3, 6, 7)$, the $\mathcal{K}_1$-terminal reliability $R(\mathcal{K}_1, \boldsymbol{x})$ should achieve 99%, thus we have

$$R(\mathcal{K}_1, \boldsymbol{x}) + d_1^- - d_1^+ = 99\%$$

where d_1^- will be minimized.

Priority 2: For the subset of nodes $\mathcal{K}_2 = (2, 4, 5, 9)$, the $\mathcal{K}_2$-terminal reliability $R(\mathcal{K}_2, \boldsymbol{x})$ should achieve 95%, thus we have

$$R(\mathcal{K}_2, \boldsymbol{x}) + d_2^- - d_2^+ = 95\%$$

where d_2^- will be minimized.

Priority 3: For the subset of nodes $\mathcal{K}_3 = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$, the $\mathcal{K}_3$-terminal reliability $R(\mathcal{K}_3, \boldsymbol{x})$ (here the overall reliability) should achieve 90%, thus we have

$$R(\mathcal{K}_3, \boldsymbol{x}) + d_3^- - d_3^+ = 90\%$$

where d_3^- will be minimized.

Then we obtain the following topological optimization model for communication network reliability,

$$\left\{ \begin{array}{l} \text{lexmin} \left\{ d_1^-, d_2^-, d_3^- \right\} \\ \text{subject to:} \\ \qquad R(\mathcal{K}_1, \boldsymbol{x}) + d_1^- - d_1^+ = 99\% \\ \qquad R(\mathcal{K}_2, \boldsymbol{x}) + d_2^- - d_2^+ = 95\% \\ \qquad R(\mathcal{K}_3, \boldsymbol{x}) + d_3^- - d_3^+ = 90\% \\ \qquad \sum_{i=1}^{9} \sum_{j=i+1}^{10} c_{ij} x_{ij} \leq 250 \\ \qquad x_{ij} = 0 \text{ or } 1, \quad \forall i, j \\ \qquad d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3. \end{array} \right.$$

A run of stochastic simulation-based genetic algorithm with 100 generations shows that the optimal link topology is

$$\boldsymbol{x}^* = \begin{pmatrix} - & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ & - & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ & & - & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ & & & - & 0 & 0 & 0 & 1 & 1 & 0 \\ & & & & - & 1 & 1 & 0 & 0 & 0 \\ & & & & & - & 0 & 0 & 1 & 0 \\ & & & & & & - & 0 & 0 & 1 \\ & & & & & & & - & 0 & 0 \\ & & & & & & & & - & 0 \\ & & & & & & & & & - \end{pmatrix}$$

which can satisfies the three goals. Moreover, the terminal reliability levels are

$$R(\mathcal{K}_1, \boldsymbol{x}^*) = 0.991, \quad R(\mathcal{K}_2, \boldsymbol{x}^*) = 0.956, \quad R(\mathcal{K}_3, \boldsymbol{x}^*) = 0.938,$$

and the total cost is 242. Additional computational results are given in Liu and Iwamura[5].

Acknowledgment

This work was supported by National Natural Science Foundation of China Grant No.69804006.

References

[1] Goldberg D. E., *Genetic Algorithms in Search, Optimization and Machine Learning,* (Addison-Wesley, 1989).

[2] Holland J.H., *Adaptation in Natural and Artificial Systems*, (University of Michigan Press, Ann Arbor, 1975).

[3] Iwamura K. and B. Liu, A genetic algorithm for chance constrained programming, *Journal of Information & Optimization Sciences*, Vol.17 (1996), No.2, 409-422.

[4] Liu B. and K. Iwamura, Chance constrained programming with fuzzy parameters, *Fuzzy Sets and Systems*, Vol.94(1998), No.2, 227-237.

[5] Liu B. and K.Iwamura, Topological Optimization Models for Communication Networks with Multiple Reliability Goals, to appear in *Computers and Mathematics with Applications.*

[6] Liu B., *Uncertain Programming*, (John Wiley & Sons, New York, 1999).

[7] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs,* 3nd ed., (Springer-Verlag, New York, 1996).