

# A New Approach for Network Topology Optimization

*T.Srinivasa Rao, S.K.Bose, K.R.Srivathsan*  
*Dept. of Elec. Engg. I.I.T. Kanpur - 208016*  
*INDIA.*

*Kalyanmoy Deb*  
*Dept. of Mech. Engg. I.I.T. Kanpur - 208016*  
*INDIA.*

## Abstract

We consider the problems of designing network topologies with minimum cost. In the first problem, basic optimum topology which guarantees only one path from one to the other node, is considered. This problem is extended in the second part to include one redundant path between two nodes. These two problems are solved in this work in three stages. In the first stage, modified version of random feasible topology generation algorithm is used to generate initial feasible solutions. In the second and third stages, Link Reduction Algorithm and Branch-X-Change algorithms are used sequentially to get a locally optimal solution. These three stages are required to repeat with different orderings of the nodes to get a global optimal solution. We tested the above approach with two problems.

## 1 Introduction

The central aim of network topology optimization is to minimise the total cost of the links subject to constraints on link utilization, redundant connections between any pair of nodes and the number of links emanating at each of the node. In a general optimization problem, the design variables are link decision variables,  $\{l_{mn}\}$ , defined as follows:

$$l_{mn} = \begin{cases} 1 & \text{if there is link between m and n} \\ 0 & \text{otherwise} \end{cases}$$

Here, the link decision variables,  $l_{mn}$  take an integer value (strictly either 0 or 1) and the capacity of the link is a continuous variable. This optimization problem is one of Mixed Integer Programming. In the objective function, the cost of a given link may depend linearly either on the link capacity or on the distance between the two nodes. If the objective function is also a linear cost function with respect to the design variables, then the optimization would be one of Mixed Integer Linear Programming.

Apart from communication system engineers, people from various engineering disciplines like transportation, production and computer systems engineering are also interested in this problem. The exact solution to this problem is not known as it is established that general network optimization is an NP- complete problem (Steiglitz, 1969). In the literature, various heuristic and approximate algorithms are proposed for solving such optimization problems. Unfortunately, most of these techniques converge to locally optimum solutions (Gersht,1990 and Magnanti,1984). Therefore one is required to solve this problem with different initial solutions to get locally optimum solutions for each case; we then select one among them as a globally optimum solution. Since it involves considerable amount of computational effort, this approach may not be practical for large sized networks.

In this report, we first look for a basic optimum topology. This means a minimum-cost topology which guarantees at least one path from one node to every other node. In the second part of the report, we concentrate on the minimum-cost topology which guarantees atleast two paths from one node to every other node in the network; this is so that the failure of any link will not make a node unreachable from other nodes.

To solve the network topology optimization problem, we use heuristic procedures in three stages. In the first stage, a feasible but not optimal topology is generated. In the second stage of the problem, we use the link reduction algorithm on the initial feasible topology to obtain a feasible topology with reduced cost. In the third and final stage, we use the Branch-X-Change algorithm to get a local optimum solution.

In Section 2, the formulation of the basic topology and its extension to a topology with one redundant path is discussed. The methodology to solve the above problem is presented in Section 3. In Section 4, the implementation details are given. The results are presented in Section 5. Fianlly in Section 6, Conclusions and future extensions are discussed.

## 2 Problem Formulation

We consider a communication network with  $N$  nodes, with each node transacting various types of packet traffic. In this design, however, we only consider one type of traffic. The other types of traffic will be considered implicitly by constraints on the link utilisation for this traffic. The nodes are proposed to be interconnected by voice grade telephone lines. The cost  $C_{ij}$  of link  $l_{ij}$  between node  $i$  and node  $j$  is the sum of a fixed installation cost  $A$  and a variable quantity which is proportional to the distance  $d_{ij}$  between two nodes. In our approach, we need to suggest a basic topology with minimum cost, subject to the following constraints:

1. Every node is reachable from every other node.
2. One of the nodes (say a given node  $K$ ), should be connected to every other node such that the shortest path for this has 4 hops or less.
3. The total number of links at some given nodes  $J, K$  and  $L$ , should not be more than 4 and at all other nodes the number of links should be either 1 or 2.
4. The link utilization factor should be in the range  $(0, u_f)$ .

In the second part of the work, this problem is extended to get a topology with redundant links. For this part, the following constraint is added to the constraint set:

5. There should be at least one redundant(arc-disjoint) path from one to every other node.

### 2.1 Optimization Problem:

The optimization problem considered here is formulated as follows—

Given:

$$\text{Requirement Matrix, } R = \begin{bmatrix} 0 & r_{12} & r_{13} & \dots & r_{1N} \\ r_{21} & 0 & r_{23} & \dots & r_{2N} \\ \dots & \dots & \dots & \dots & \dots \\ r_{N1} & r_{N2} & r_{N3} & \dots & 0 \end{bmatrix}$$

where  $r_{ij}$  = Traffic flow (Bytes/hr) between node  $i$  and node  $j$ .

$$\text{Distance Matrix, } D = \begin{bmatrix} 0 & d_{12} & d_{13} & \dots & d_{1N} \\ d_{21} & 0 & d_{23} & \dots & d_{2N} \\ \dots & \dots & \dots & \dots & \dots \\ d_{N1} & d_{N2} & d_{N3} & \dots & 0 \end{bmatrix}$$

where  $d_{ij}$  = Distance between node  $i$   
and node  $j$  in units of 100  
kilometers.

**Find :**

$$\text{Link Matrix} = \begin{bmatrix} 0 & l_{12} & l_{13} & \dots & l_{1N} \\ l_{21} & 0 & l_{23} & \dots & l_{2N} \\ \dots & \dots & \dots & \dots & \dots \\ l_{N1} & l_{N2} & l_{N3} & \dots & 0 \end{bmatrix}$$

**Such that :**

$$\text{Minimize } \bar{C} = \sum_{i,j} l_{ij}(A + d_{ij}B)$$

**Subject to :**

$$(a) \quad 0 \leq \frac{f_{ij}}{2400} \leq u_f \leq 1$$

where  $f_{ij}$  = traffic flow on link  $l_{ij}$  in bits  
per sec.

$$(b) \quad 1 \leq \sum_j l_{ij} \leq 4 \quad \text{if } i = j, K \text{ or } L$$

$$1 \leq \sum_j l_{ij} \leq 2 \quad \text{otherwise}$$

(d) The no. of hops in the shortest path from node  $K$  to every other node is also constrained to be in the range of (1,4).

The constraint (d) also implicitly ensures that every node is reachable from every other node. For the extended problem, one more constrain is added. This states that-

(e) There should be at least one redundant path between one node and every other node.

### 3 Heuristic Approach

As mentioned earlier, most heuristic algorithms for solving the topology problems, result in a locally optimum solution for a given initial solution. We need to use these heuristics repeatedly for different initial feasible solutions to obtain different local optimum topologies. In this section, we discuss an algorithm to generate an initial feasible topology. This algorithm is a version of the Randomised routin proposed in (Steiglitz,1969), which has been modified to suit our problem.

### 3.1 Randomised Starting Routine

The algorithm is proposed in (Steiglitz,1969) to generate initial feasible solution randomly in the absence of any constraint other than the redundancy paths. i.e. there should be at least  $s_{ij}$  paths between node  $i$  and node  $j$ . To generate an initial topology which satisfies the above constraint, the following steps are followed–

- 1: Order the nodes randomly.
- 2: A new link is added with the highest degree of requirement and a node with next highest degree of requirement, where–

Degree of requirement of node  $i$  = maximum of  $s_{ij}$  over all  $j$ .

Of all those nodes with the next highest degree of requirement, the node with least distance is chosen.

3. The degree of requirement of the nodes between which a link added is reduced by one for each.
4. If the degree of requirement of all the nodes are zero, the procedure terminates. Otherwise, repeat from (2).

This algorithm will generate different initial feasible topologies with different orderings of the nodes.

### 3.2 Modified Starting Routine

The above algorithm is modified to suit our problem. The modified definition of degree of requirement is the maximum number of links emanating at the node. The terminating condition, Step 4, is also modified as follows:

4. If the highest and next highest degrees of requirement are greater than zero, then go to step 2. Otherwise terminate.

The topology so obtained from the above modified algorithm, will satisfy only the constraint on the number links at each node. Hence before accepting the topology as feasible, the algorithm is extended as follows with the addition of the following steps:

5. For every node  $i$  other than node  $K$ , find shortest distance to node  $K$ . If the shortest path from any node is more than 4 (as specified in the constraint), go to Step (8). Otherwise go to Step (6).
6. From every node  $i$  to every other node  $j$ , find the shortest path. If the shortest path is equal to zero, go to Step (8). Otherwise update the flow  $f_{mn}$  for every link  $l_{mn}$  in the shortest path between node  $i$  and node  $j$  by  $r_{ij}$ .

7. For every  $i$  and  $j$ , calculate the ratio of the flow  $f_{ij}$  and capacity of the link  $l_{ij}$ . If for any link, the ratio is more than  $u_f$  (as given in the constraints), go to Step (8). Otherwise terminate.
8. Permute the ordering of the nodes and rearrange distance matrix  $d_{ij}$  and flow requirement matrix  $r_{ij}$  accordingly. The repeat from Step (2).

### 3.3 Link Reduction Algorithm

This algorithm is proposed in (Gersht,1990) for improving (reducing) the cost of the topology.

1. Let the initial topology be  $T^0$  and set  $v = 1$ . Calculate the cost  $C^0$  of the topology  $T^0$ .
2. Pre-elimination of link  $l_{mn}$ :  
Generate topology  $T_{mn}^v$  by eliminating a new link  $l_{mn}$  from  $T^{v-1}$ .
3. check  $T_{mn}^v$  to see whether the constraints are satisfied or not. If not, go to step 2. Otherwise include  $T_{mn}^v$  in  $T^v$ , the set of retained topologies for iteration  $v$ .
4. Compute  $\Delta C_{mn}^v = C^{v-1} - C_{mn}^v \rightarrow \Delta C^v$ .

If every feasible link is reduced then go to step 4. Otherwise go to step 2.

5. Get cost set  $\Delta C^v$  for topologies  $T_v$ .

Find  $\max_{k,l} \Delta C_{kl}^v \rightarrow (\Delta C_{mn}^v, T_{mn}^v)$ .

Then  $T^v = T_{mn}^v$  ;  $C^v = C_{mn}^v$  .

6. If  $C^{v-1} - C^v > 0$ , then  $v = v + 1$ , and go to step 2. Otherwise, terminate.

### 3.4 Branch-X-Change Algorithm

The topology obtained from Link Reduction Algorithm is used as the initial topology in Branch-X-Change algorithm (Steiglitz,1969). This algorithm is another popular algorithm for network optimization. The basic idea in this algorithm is that in the topology  $T$ , link  $l_{ij}$  and link  $l_{mn}$  are replaced by links  $l_{im}$  and link  $l_{mj}$ , if the topology  $T'$  so obtained, is feasible and of lower cost compared to that topology  $T$ .

The Algorithm is:

1. Let the initial topology be  $T_0$  with cost  $C_0$ . And set  $v = 1$ .
2. select links  $l_{ij}$  and  $l_{mn}$ . Remove links  $l_{ij}$  and  $l_{mn}$  add links  $l_{in}$  and  $l_{mj}$  to get a topology  $T'_1$ .
3. Calculate the cost  $C'_1$  of  $T'_1$ . If  $C'_1 < C_0$ , go to Step 4. otherwise go to Step 5.
4. Check for feasibility of  $T'_1$ . If feasible,  $T_1 = T'_1$  and  $C_1 = C'_1$ . Otherwise  $T_1 = T_0$  and  $C_1 = C_0$ .
5. If all links are checked terminate. Otherwise, set  $v = v + 1$ , go to Step 2.

The topology so obtained is a locally optimum solution. Since there is no guarantee that this topology would be globally optimum, we need to compute different locally optimum solutions by starting with different initial solutions. As discussed above, we can get different initial feasible topologies from the modified starting routine by varying the ordering of the nodes every time.

## 4 Implementation

The above discussed algorithm was implemented in C for solving the network topology problems presented in Section 2. The module level flow chart is given in Fig. 1.

The code is developed in an object-specific modular fashion so that the same code can be used for different kinds of network topologi problems with little changes like addition and deletion of particular mudules. In this work, we have solved both the problems presented in Section 2 using essentially same code. This has been done by appropriately adding submodules in order to take care of additional constraints in the second problem.

## 5 Results and Discussion

Most of the topology optimization algorithms proposed in the literature are based on the searching of the feasible topology space. The efficiency of any such search algorithm depends on the range of the search space. For a given initial topology, this varies from algorithm to algorithm. In the Link Reduction algorithm, the search space is such that the cost of the feasible topologies reduces monotonically. The disdvantage of this is that it searches for an optimal topology in only one narrow direction. In other words, in the initial feasible topology, if there is no link between node  $i$  and node  $j$ , there will not be a link between node  $i$  and node  $j$  in the locally optimum topology obtained from the Link Reduction Algorithm. Hence, the range of search space for this algorithm is very small. This drawback is partly overcome in our approach by using the Branch-X-Change

algorithm. As explained earlier, in this algorithm, the link  $l_{ij}$  and link  $l_{mn}$  are replaced by links  $l_{in}$  and  $l_{mj}$ , if the topology so obtained is feasible and at improved cost. In other words, the Branch - X - Change algorithm searches for a better solution in neighborhood regions and hence results in an increase in the range of the search space.

We have tested the code on an HP 9000/300 workstation with two network topology problems. As discussed earlier, like most of the search algorithms, the above approach also converges to a locally optimum topology which varies from one initial topology to another. It may not be practical from a computational point of view, to generate locally optimum solutions for every initial feasible solution; the code is tested for the generation of fixed but not exhaustive number of locally optimum solutions. In this report, we have tested the code for generating thirty locally optimum solutions for each instance of the problem. The results are given below:

## 5.1 Problem 1

Here we have a 6- node network, whose distance Matrix D is given by,

$$D = \begin{bmatrix} 0.0 & 3.8 & 8.0 & 10.3 & 14.2 & 13.7 \\ 3.8 & 0.0 & 11.2 & 10.6 & 16.0 & 16.3 \\ 8.0 & 11.2 & 0.0 & 14.2 & 13.6 & 11.7 \\ 10.3 & 10.6 & 14.2 & 0.0 & 7.7 & 9.6 \\ 14.2 & 16.0 & 13.6 & 7.7 & 0.0 & 2.7 \\ 13.7 & 16.3 & 11.7 & 9.6 & 2.7 & 0.0 \end{bmatrix}$$

Flow requirement Matrix R, is given by,

$$R = \begin{bmatrix} 0 & 1885 & 1071 & 13097 & 1402 & 870 \\ 6736 & 0 & 2425 & 25547 & 3252 & 1919 \\ 2146 & 2399 & 0 & 18581 & 1870 & 1070 \\ 64892 & 61328 & 50346 & 0 & 74692 & 35490 \\ 2403 & 3116 & 1759 & 37391 & 0 & 1455 \\ 2264 & 2770 & 1551 & 27185 & 2047 & 0 \end{bmatrix}$$

Also  $J = 2, K = 4, L = 5$  and  $u_f = 0.5$ .

The installation cost of link = Rs. 40,000 (fixed) The incremental cost of link = Rs. 75,000 per unit distance.

### 5.1.1 Basic Optimal Topology

For one particular ordering of the nodes, the initial feasible topology is shown in Fig. 2. Its total cost is Rs. 54,37,500. After Link Reduction Algorithm, the improved topology is given in Fig. 3. Its cost is Rs. 41,97,500. After Branch - X - Change Algorithm, the



local optimum topology is as shown in Fig. 4. The cost of this topology is Rs. 35,97,500. We have run this program and generated thirty locally optimum topologies (among these, some may be identical). Among these topologies, the topology with the lowest cost is considered as global optimal topology. The global optimal topology is as shown in Fig. 5 and its cost is Rs. 26,60,000. The execution time of the code for obtaining these thirty local optimal topologies is 0.16 seconds.

**5.1.2 Optimal Topology with One Redundant Path**

The global optimal topology (best of thirty locally optimal topologies) is shown in Fig. 6 and its cost is Rs. 35,77,500. To generate these thirty topologies, the execution time required is 0.23 seconds.

We have also obtained optimum topologies for different value of  $u_f$  (i.e.  $u_f = 0.4, 0.3, 0.2$ ). It is observed that for the given traffic flow requirements, the optimal topology remained unchanged if  $u_f$  is as low as 0.3.

**5.2 Problem 2**

In this problem we considered an 8 - node network whose distance matrix is given by D given below,

$$D = \begin{bmatrix} 0.0 & 3.8 & 8.0 & 10.3 & 14.2 & 13.7 & 9.3 & 8.5 \\ 3.8 & 0.0 & 11.2 & 10.6 & 16.0 & 16.3 & 11.2 & 12.0 \\ 8.0 & 11.2 & 0.0 & 14.2 & 13.6 & 11.7 & 9.7 & 1.0 \\ 10.3 & 10.6 & 14.2 & 0.0 & 7.7 & 9.6 & 5.5 & 15.4 \\ 14.2 & 16.0 & 13.6 & 7.7 & 0.0 & 2.7 & 4.7 & 14.4 \\ 13.7 & 16.3 & 11.7 & 9.6 & 2.7 & 0.0 & 5.0 & 12.6 \\ 9.3 & 11.2 & 9.7 & 5.5 & 4.7 & 5.0 & 0.0 & 11.0 \\ 8.5 & 12.0 & 1.0 & 15.4 & 14.4 & 12.6 & 11.0 & 0.0 \end{bmatrix}$$

The flow requirement matrix for this problem is chosen as,

$$R = \begin{bmatrix} 0 & 1885 & 1071 & 13097 & 1402 & 870 & 870 & 1071 \\ 6736 & 0 & 2425 & 25547 & 3252 & 1919 & 1919 & 2425 \\ 2146 & 2399 & 0 & 18581 & 1870 & 1070 & 1070 & 3000 \\ 63892 & 61328 & 50346 & 0 & 74692 & 35490 & 35490 & 50346 \\ 2403 & 3116 & 1759 & 37391 & 0 & 1455 & 1455 & 1759 \\ 2264 & 2770 & 1551 & 27185 & 2047 & 0 & 3000 & 1551 \\ 2264 & 2270 & 1551 & 27185 & 2047 & 5000 & 0 & 2500 \\ 2146 & 2399 & 5000 & 18581 & 1870 & 1070 & 2000 & 0 \end{bmatrix}$$

Also  $J = 2, K = 4, L = 5$  and  $u_f = 0.5$ .

### 5.2.1 Basic Optimal Topology

The global optimal topology obtained from our program is shown in Fig. 7. Its cost is Rs. 32,65,000. The time for execution to generate thirty local optimal topologies is 0.33 seconds.

### 5.2.2 Optimal Topology with One Redundant Path

The global optimal topology resulting from our program is given in Fig. 8 and its cost is Rs. 39,57,500. The time for the generation of the thirty optimal topologies is 0.35 seconds.

## 6 Conclusions and Future Extensions

In this work, we studied some heuristic algorithms for network topology optimisation. The first is for generating an initial feasible topology. This is a modified version of the algorithm presented in (Steiglitz,1969). The other two algorithms are for optimising these initial feasible topologies. They are Link Reduction Algorithm (Gersht,1990) and Branch - X - Change Algorithm (Steiglitz,1969). In this work, we observed that the topology obtained from Link Reduction Algorithm may not always be optimum. This is because of the fact that the link reduction algorithm searches only in the direction of the initial feasible topology. Hence improvement in the solution was observed when we execute the Branch - X - Change Algorithm with the topology obtained from Link Reduction Algorithm, as i.e. the initial feasible topology for the Branch - X - Change Algorithm. We have developed the code for these algorithms in C and tested it with two different problems. An increase in range of the search space, due to the application of Branch - X - Change Algorithm preceded by Link Reduction Algorithm, is observed. The range of search space can be improved further by applying both Link Reduction Algorithm and Branch - X - Change Algorithm in alternative iterations. The initial starting routine can be modified so that the ordering of the nodes is generated randomly.

## 7 References

- Gerla,M. and Kleinrock,L.(1977) *On the Topological Design of Distributed Computer Networks*, IEEE Trans. on Communications, Vol. COM-25, No. 1, Jan 1977
- Gersht,A. and Weihmayer,R.(1990) *Joint Optimization of Data Network Design and Facility Selection*, IEEE Jl. on S.A.C., Vol. 8, No. 9, Dec 1990

Magnanti, T.L. and Wong, R.T. (1984) *Network Design and Transportation Planning: Models and Algorithms*, Transport Science, Vol. 18, No. 1, Feb 1984

Steiglitz, K., Weiner, P. and Kletman, D.J. (1969) *The Design of Minimum-cost Survivable Networks*, IEEE Trans. on Circuits Theory, Vol. CT-16, No. 4, Nov 1969

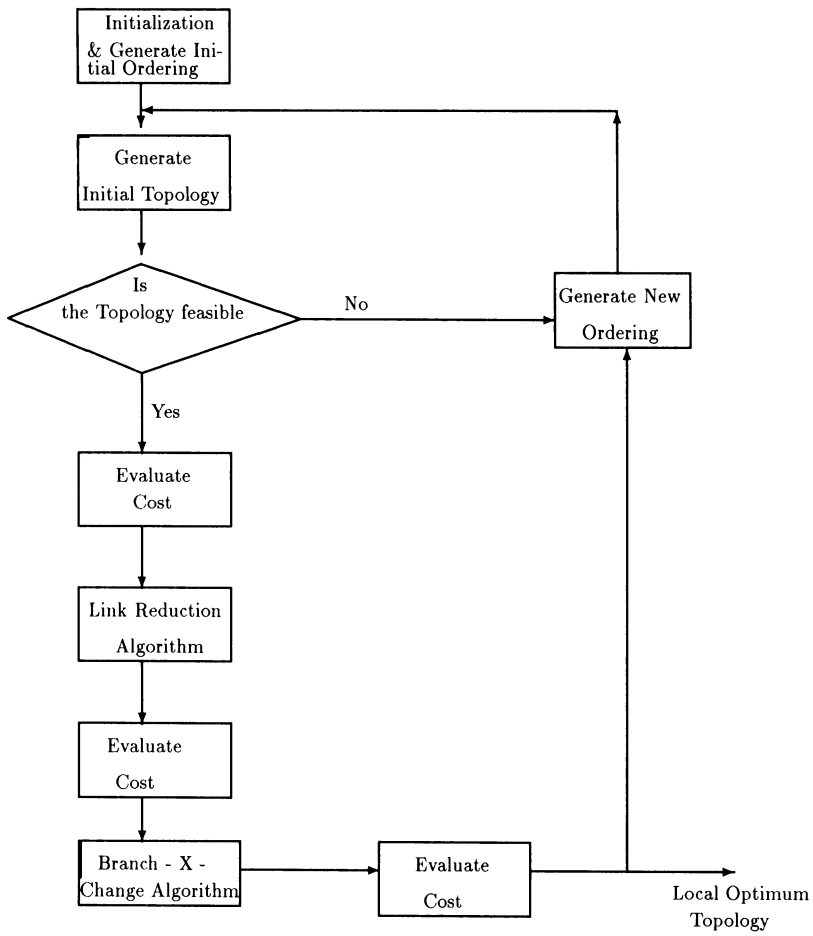
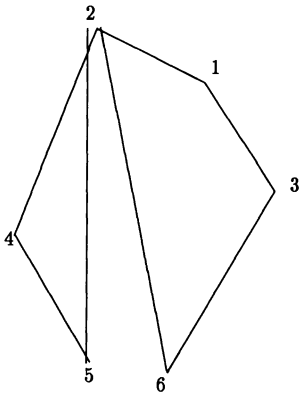
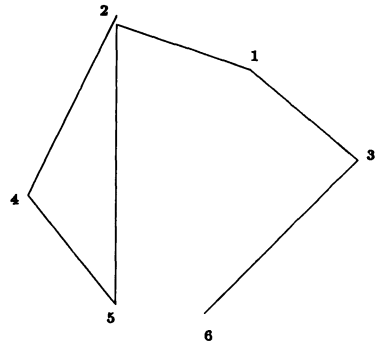


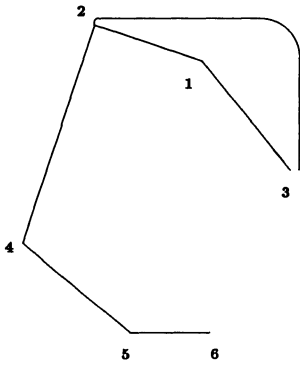
Fig. 1 Flow Chart



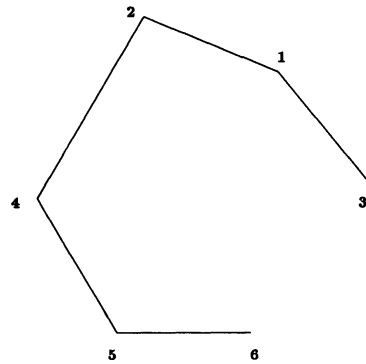
**Fig. 2 Initial Feasible Topology**



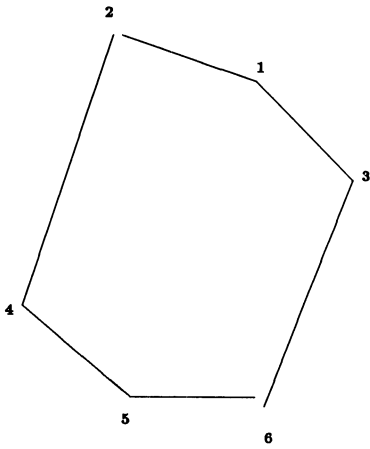
**Fig. 3 Reduced Cost Topology, from Link Reduction Algorithm**



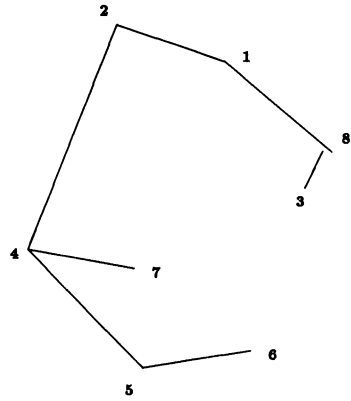
**Fig. 4 Locally Optimum Topology**



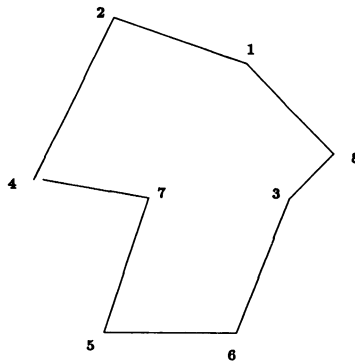
**Fig. 5 Global Optimal Topology**



**Fig. 6 Optimal Topology With One Redundant Path**



**Fig. 7 Basic Global Optimal Topology**



**Fig. 8 Global Topology With One Redundant Path**