Backbone Network Design with QoS Requirements

Hong-Hsu Yen and Frank Yeong-Sung Lin

Department of Information Management,
National Taiwan University, Taiwan
{d4725001, yslin}@im.ntu.edu.tw

Abstract. In this paper, we consider the backbone network design problem with a full set of QoS requirements. Unlike previous researches, we consider both the transmission line cost and the switch cost. And the QoS requirements that we considered include the average packet delay, end-to-end packet delay and node disjoint paths. We formulate the problem as a combinatorial optimization problem where the objective function is to minimize the total network deployment cost subject to the aforementioned QoS constraints. Besides the integrality constraints, the nonlinear and the nonconvex properties associated with the problem formulation make it difficult. Lagrangean relaxation in conjunction with a number of optimization-based heuristics are proposed to solve this problem. From the computational experiments, the proposed algorithms calculate creditable solutions in minutes of CPU time for moderate problem sizes.

1 Introduction

How to design a usually sophisticated backbone network with the minimum deployment and operation cost subject to various and often stringent QoS requirements is a common challenge faced by network designers and managers. Intensive research has been conducted to address this issue. However, most research tackles this backbone network design problem without considering a full set of QoS requirements.

Gavish model the network topological design problem as a nonlinear combinatorial optimization problem. The objective is to minimize the network installation cost and the queueing cost imposed on the network users. However, network installation cost and the queueing cost are two different concepts such that it is not appropriate to put them together in the objective function. In addition, end-to-end QoS constraints are not considered in [3].

A bunch of heuristics based on genetic algorithms (GA) and Tabu-Search algorithms are proposed to tackle the backbone network design problem [2, 5, 6]. However, in [2, 5, 6], the cost for network access point is not considered in the network installation cost. Furthermore, average and end-to-end delay constraints and reliability constraint are not jointly considered at the same time.

The delay QoS requirement is crucial to modern application services (e.g. VOD, tele-conferencing). And in particular, backbone network usually requires high availability, that is, redundant links and switching nodes are needed in case of failure. As a

P. Lorenz (Ed.): ICN 2001, LNCS 2094, pp. 148–157, 2001

result, unlike more of previous research, the QoS requirement considered in this paper can be classified into two parts. The first part is the delay QoS (including the average packet delay and end-to-end delay for each O-D pair) and the second part is the reliability and availability of services. On the other hand, the network construction cost includes both the switch and link installation cost to reflect the cost structure of real network. This problem is a well-known difficult NP-hard problem.

This paper is organized as follows. In Section 2, mathematical formulation of the backbone network design problem is proposed. In Section 3, the dual approach for the backbone network design problem based on the Lagrangean relaxation is presented. In Section 4, the getting primal heuristics are developed to get the primal feasible solutions from the solutions to the dual problem. In Section 5, the computational results are reported. In Section 6, the concluding remarks are presented.

2 Problem Formulation

This QoS based backbone network design problem is modeled as the graph where the users and switches are depicted as nodes and the communication channels are depicted as arcs. We show the definition of the following notation.

L	The set of candidate local loop links and backbone links in the communication network.
W	The set of origin-destination (O-D) pairs in the network.
λ_w	The traffic requirement for each O-D pair $w \in W$.
$\overline{C_l}$	a capacity upper bound in the candidate capacity configurations for link $l \in L$.
P_w	a given set of simple directed paths from the origin to the destination of O-D pair w.
U_k	a set of potential incoming links to switch k.
δ_{pl}	the indicator function which is one if link l is on path p and zero otherwise.
ε_{pk}	the indicator function which is one if switch k is on path p and zero otherwise.
g_l	the aggregate flow over link $l \in L$, which is $\sum_{p \in P_w} \sum_{w \in W} x_p \lambda_w \delta_{pl}$.
D_w	the maximum allowable end-to-end delay requirement for O-D pair w.
K	the maximum allowable average cross network delay requirement.
T_w	the minimum number of node disjoint paths required for O-D pair w.
O	the set of candidate locations for switches.
H	the set of link pairs that are with the same end points but in opposite directions.
A_l	the set of candidate capacity configurations for link l.
R_k	the set of admissible switching fabric configuration for switch at location k.

E_k	the set of candidate port configuration for switch at location *k*.
$\varphi_l(C_l)$	the cost for installing capacity C_l on link *l*, including the fixed and variable cost.
ξ_l	the fixed link installation cost for link *l*.
$Q_k(J_k, S_k)$	the cost for installing a switch at location *k* with switching fabric capacity J_k and number of ports S_k.
$F_l(f_l, C_l)$	the average delay on link $l \in L$, which is a function of f_l and C_l.
$B_l(f_l, C_l)$	the average number of packets on link $l \in L$, which is a function of f_l and C_l, and by the Little's results, which is equal to $\lambda_w * F_l(f_l, C_l)$.

And the decision variables are depicted as follows.

x_p	1 when path $p \in P_w$ is used to transmit the packets for O-D pair $w \in W$ and 0 otherwise.
z_p	1 when path $p \in P_w$ is the node disjoint path for O-D pair $w \in W$ and 0 otherwise.
y_{wl}	1 when link $l \in L$ is on the path chosen for O-D pair $w \in W$ and 0 otherwise.
f_l	the estimated aggregate flow on link $l \in L$.
M_l	1 when a link is installed at location $l \in L$ and 0 otherwise.
C_l	the capacity assignment for link $l \in L$.
J_k	the switching fabric capacity assignment for switch at location *k*.
S_k	the number of ports for switch at location *k*.

We formulate the QoS based backbone network design problem as a nonlinear and nonconvex combinatorial optimization problem, as shown below.

$$\min Z_{IP} = \sum_{l \in L} \varphi_l(C_l) + \sum_{k \in O} Q_k(J_k, S_k) - \sum_{(l,\bar{l}) \in H} \xi_l M_l M_{\bar{l}} \quad \text{(IP)}$$

subject to:

$$\frac{1}{\sum_{w \in W} \lambda_w} \sum_{l \in L} B_l(f_l, C_l) \le K \quad (1)$$

$$\sum_{l \in L} y_{wl} F_l(f_l, C_l) \le D_w \qquad \forall w \in W \quad (2)$$

$$\sum_{p \in P_w} x_p = 1 \qquad \forall w \in W \quad (3)$$

$$x_p = 0 \text{ or } 1 \qquad \forall p \in P_w, w \in W \quad (4)$$

$$\sum_{p \in P_w} x_p \delta_{pl} \le y_{wl} \qquad \forall w \in W, l \in L \quad (5)$$

$$y_{wl} = 0 \text{ or } 1 \qquad \forall w \in W, l \in L \quad (6)$$

$$g_l = \sum_{w \in W} \sum_{p \in P_w} x_p \delta_{pl} \lambda_w \le f_l \qquad \forall l \in L \qquad (7)$$

$$f_l \le C_l \qquad \forall l \in L \qquad (8)$$

$$C_l \in A_l \qquad \forall l \in L \qquad (9)$$

$$M_l = 0 \text{ or } 1 \qquad \forall l \in L \qquad (10)$$

$$C_l \le \overline{C_l} M_l \qquad \forall l \in L \qquad (11)$$

$$J_k \in R_k \qquad \forall k \in O \qquad (12)$$

$$\sum_{l \in U_k} M_l \le S_k \qquad \forall k \in O \qquad (13)$$

$$S_k \in E_k \qquad \forall k \in O \qquad (14)$$

$$\sum_{w \in W} \sum_{p \in P_w} x_p \varepsilon_{pk} \lambda_w \le J_k \qquad \forall k \in O \qquad (15)$$

$$\sum_{p \in P_w} z_p = T_w \qquad \forall w \in W \qquad (16)$$

$$\sum_{p \in P_w} z_p \delta_{pl} \le M_l \qquad \forall w \in W, l \in L \qquad (17)$$

$$z_p = 0 \text{ or } 1 \qquad \forall p \in P_w, w \in W\ . \qquad (18)$$

The objective is to minimize network installation cost. There are three terms in the objective function. The first term is to compute the total link installation cost, including the fixed cost and the variable cost. The second term is to compute the total switch installation cost. The third term is to compute one fixed cost for each installed opposite links. The necessity of subtracting the third term is to ensure that only one rather than two fixed cost is calculated for two links with the same attached nodes but in opposite direction. Constraint (1) enforce the average cross network delay constraint. Constraint (2) enforce the end-to-end packet delay for each O-D pair. Constraints (3) and (4) require that the all the traffic for each O-D pair should be transmitted over exactly one path. The decision variable y_{wl} in Constraint (6) is an auxiliary decision variable, which is equal to $\sum_{p \in P_w} x_p \delta_{pl}$. Hence, the equality in Constraint (5) is replaced by inequality due to the ease use of the Lagrangean relaxation. Constraints (7) and (8) are the link capacity constraints. Constraint (9) determines the possible capacity configurations of all links. Constraints (10) and (11) require that the link must be installed first before link capacity assignment. Constraints (12) and (14) determine the possible switching fabric and number of ports of all switches. Constraint (13) is the switch termination constraint, which means the number of incoming links to the switch should not exceed the number of ports on that switch. Constraint (15) is the switch capacity constraint Constraints (16) and (18) are the path diversity (node disjoint) requirement for each O-D pair. Constraint (17) guarantees that link must be installed first before it could be adopted on the node disjoint path for each O-D pair.

3 Lagrangean Relaxation

We dualize Constraints (1), (2), (5), (7), (8), (11), (13), (15) and (17) of Problem (IP) to get the following Lagrangean relaxation problem (LR).

$$\min\ Z_D = \sum_{l\in L}\varphi_l(C_l) + \sum_{k\in O} Q_k(J_k, S_k) - \sum_{(l,\bar{l})\in H} \xi_l M_l M_{\bar{l}} + a[\frac{1}{\sum_{w\in W}\lambda_w}\sum_{l\in L} B_l(f_l, C_l) - K]$$

$$+ \sum_{w\in W} b_w[\sum_{l\in L} y_{wl} F_l(f_l, C_l) - D_w] + \sum_{w\in W}\sum_{l\in L} c_{wl}[\sum_{p\in P_w} x_p\delta_{pl} - y_{wl}] + \sum_{l\in L} h_l[f_l - C_l] +$$

$$\sum_{l\in L} d_l[\sum_{w\in W}\sum_{p\in P_w} x_p\delta_{pl}\lambda_w - f_l] + \sum_{l\in L} e_l[C_l - \overline{C}_l M_l] + \sum_{k\in O} n_k[\sum_{w\in W}\sum_{p\in P_w} x_p\varepsilon_{pk}\lambda_w - J_k] +$$

$$+ \sum_{k\in O} m_k[\sum_{l\in U_k} M_l - S_k] + \sum_{w\in W}\sum_{l\in L} q_{wl}[\sum_{p\in P_w} z_p\delta_{pl} - M_l] \qquad \text{(LR)}$$

subject to:

$$\sum_{p\in P_w} x_p = 1 \qquad \forall w\in W \qquad (19)$$

$$x_p = 0 \text{ or } 1 \qquad \forall p\in P_w, w\in W \qquad (20)$$

$$y_{wl} = 0 \text{ or } 1 \qquad \forall w\in W, l\in L \qquad (21)$$

$$C_l \in A_l \qquad \forall l\in L \qquad (22)$$

$$M_l = 0 \text{ or } 1 \qquad \forall l\in L \qquad (23)$$

$$J_k \in R_k \qquad \forall k\in O \qquad (24)$$

$$S_k \in E_k \qquad \forall k\in O \qquad (25)$$

$$\sum_{p\in P_w} z_p = T_w \qquad \forall w\in W \qquad (26)$$

$$z_p = 0 \text{ or } 1 \qquad \forall p\in P_w, w\in W\ . \qquad (27)$$

We can decompose (LR) into five independent subproblems.

Subproblem 1: for x_p:

$$\min \sum_{w\in W}\sum_{p\in P_w}[\sum_{l\in L}(c_{wl} + d_l\lambda_w)x_p\delta_{pl} + \sum_{k\in O} n_k\lambda_w x_p\varepsilon_{pk}] \qquad \text{(SUB1)}$$

subject to: (19) and (20).

Subproblem 2 : for C_l, y_{wl} and f_l:

$$\min \sum_{l\in L}\varphi_l(C_l) + a\frac{1}{\sum_{w\in W}\lambda_w}\sum_{l\in L} B_l(f_l, C_l) + \sum_{w\in W} b_w \sum_{l\in L} y_{wl} F_l(f_l, C_l) - \sum_{w\in W}\sum_{l\in L} c_{wl} y_{wl} -$$

$$\sum_{l\in L} d_l f_l + \sum_{l\in L} e_l C_l + \sum_{l\in L} h_l f_l - \sum_{l\in L} h_l C_l \qquad \text{(SUB2)}$$

subject to: (21) and (22).

Subproblem 3: for M_l:

$$\min\ -\sum_{l\in L} e_l \overline{C}_l M_l + \sum_{k\in O}\sum_{l\in U_k} m_k M_l - \sum_{(l,\bar{l})\in H}\xi_l M_l M_{\bar{l}} - \sum_{w\in W}\sum_{l\in L} q_{wl} M_l \qquad \text{(SUB3)}$$

subject to: (23).

Subproblem 4: for J_k and S_k:

$$\min \sum_{k\in O}[Q_k(J_k,S_k)-n_kJ_k-m_kS_k] \tag{SUB4}$$

subject to: (24) and (25).

Subproblem 5: for z_p:

$$\min \sum_{w\in W}\sum_{l\in L} q_{wl}\sum_{p\in P_w} z_p\delta_{pl} \tag{SUB5}$$

subject to: (26) and (27).

In order to deal with the nodal weight of (SUB1), the node spiltting technique [4] is used. As a result, (SUB1) could be further decomposed into $|W|$ independent shortest path problem with nonnegative arc weights. It can be easily solved by the Dijkstra's algorithm. (SUB2) could also be decomposed into $|L|$ independent sub-problems. For each link $l\in L$,

Subproblem 2.1: for C_l, y_{wl} and f_l:

$$\min \quad \varphi_l(C_l)+a\frac{1}{\sum_{w\in W}\lambda_w}B_l(f_l,C_l)+\sum_{w\in W}b_w\ y_{wl}F_l(f_l,C_l)-\sum_{w\in W}c_{wl}\ y_{wl}-d_lf_l+e_lC_l+ h_lf_l-h_lC_l \tag{SUB2.1}$$

subject to: $y_{wl}=0$ or $1\ \forall w\in W$ and $C_l\in A_l$.

(SUB2.1) is a complicated problem due to the coupling of three decision variables, C_l, y_{wl} and f_l. Since the possible capacity configurations of links are finite, such as 64kbps, 128kbps, 256kbps, 512kbps, T1 and T3 for example. We can exhaustive search all different possible link configuration by finding the best y_{wl} and f_l. In [1], Lin proposed an efficient algorithm to solve y_{wl} and f_l at a given link capacity under *M/M/1* queuing model. Therefore, the algorithm to solve (SUB2.1) under *M/M/1* queuing model is proposed as bellow. In addition, the formulation could be extended to any non *M/M/1* model with monotonically increasing and convexity performance metrics.

Step 1. For each possible link capacity configuration, applying the algorithm developed in [1] to solve (SUB2.1) as to find the optimal y_{wl} and f_l.

Step 2. Finding the minimum objective value of (SUB2.1) from the objective value associated with each possible link capacity configuration. Then y_{wl} and f_l can be determined from the optimal link capacity.

(SUB3) can be decomposed into $|H|$ independent subproblems. For each pair of bi-directional links $(l,\bar{l})\in H$,

Subproblem 3.1: for M_l and $M_{\bar{l}}$:

$$\min \qquad -e_l\overline{C_l}M_l-e_{\bar{l}}\overline{C_{\bar{l}}}M_{\bar{l}}+G_1M_l+G_2M_{\bar{l}}-\xi_lM_lM_{\bar{l}}-\sum_{w\in W}q_{wl}M_l-\sum_{w\in W}q_{w\bar{l}}M_{\bar{l}} \tag{SUB3.1}$$

subject to: $M_l=0$ or 1 and $M_{\bar{l}}=0$ or 1.

In the above formulation, the G_1 and G_2 are calculated as follows.

1. If the link l is the incoming link to any potential switch, say k_1, then assign G_1 to m_{k_1}, else assign G_1 to zero.
2. If the link $\bar{l}$ is the incoming link to any potential switch, say k_2, then assign G_2 to m_{k_2}, else assign G_2 to zero.

In (SUB3.1), two opposite direction links are considered at the same time. As a result, the algorithm to optimally solve (SUB3.1) is proposed as follows.

Step 1. Let $N_1 = 0$, $N_2 = -e_l\overline{C_l} + G_1 - \sum_{w\in W} q_{wl}$, $N_3 = -e_{\bar{l}}\overline{C_{\bar{l}}} + G_2 - \sum_{w\in W} q_{w\bar{l}}$, $N_4 = -e_l\overline{C_l} + G_1 - e_{\bar{l}}\overline{C_{\bar{l}}} + G_2 - \xi_l - \sum_{w\in W} q_{wl} - \sum_{w\in W} q_{w\bar{l}}$.

Step 2. Identify the N_i with the minimum value, where i = 1, 2, 3, 4.

Step 3. If i = 1, then assign $M_l = 0$ and $M_{\bar{l}} = 0$, else if i = 2, then assign $M_l = 1$ and $M_{\bar{l}} = 0$, else if i = 3, then assign $M_l = 0$ and $M_{\bar{l}} = 1$, else if i = 4, then assign $M_l = 1$ and $M_{\bar{l}} = 1$.

(SUB4) can be further decomposed into $|O|$ independent subproblems. For each independent subproblem, due to the number of possible switch configurations (including number of ports and switching fabric) is finite and manageable within computational time, we can exhaustively search all possible combination of switch configurations as to find the optimal J_k and S_k.

(SUB5) can be further decomposed into $|W|$ independent node disjoint shortest path problem with nonnegative arc weights. Suurballe propose an efficient algorithm to optimally solve link disjoint path problem [7]. Hence, (SUB5) could be optimally solved by the Suurballe's algorithms in conjunction with the node splitting technique.

According to the algorithms developed above to solve each subproblem, we could successfully solve the Lagrangean relaxation problem optimally. By using the weak Lagrangean duality theorem (for any given set of non-negative multipliers, the optimal objective function value of the corresponding Lagrangean relaxation problem is a lower bound on the optimal objective function value of the primal problem), Z_D is a lower bound on Z_{IP}. We could construct the dual problem to calculate the tightest lower bound and solve the dual problem by using the subgradient method.

4 Getting Primal Feasible Solutions

To obtain the primal solutions to the (IP), solutions to the (LR) are considered. We develop sophisticated getting primal heuristics to getting the primal feasible solutions. This getting primal heuristic start with the routing assignment obtained from the (SUB2.1). From the routing assignment in (SUB2.1), the aggregate flow on each link can be calculated. In order to satisfy the end-to-end delay requirement for each O-D pair, the tightest end-to-end delay for all O-D pairs is located by searching the minimum end-to-end delay requirement among all O-D pairs. From the tightest end-to-end delay, the tightest link delay can be calculated by dividing the tightest end-to-end delay to the maximum hop number in any routing path. The maximum hop number for any O-D pair is equal to the number of potential switches plus one, since the source node must home to the switch first, and then route to the other switches, and finally route to the destination node. From the tightest link delay, we can determine the minimum link capacity in order to satisfy the tightest link delay requirement. From the above statement, we could satisfy the delay requirements.

In order to satisfy the node disjoint requirement for each O-D pair, the node disjoint path assignment from (SUB5) is used. If the associated link on any node disjoint path did not install at the above procedure, the minimum nonzero capacity is installed on that link. After the link capacity is determined, the number of links incoming to each potential switch can be determined. Also from the aggregate flow on each link, the total aggregate flow incoming to each potential switch can also be determined. As a result, the minimum cost switch configuration in order to satisfy the number of ports and switch fabric constraints can be determined as well.

5 Computational Experiments

The network planning algorithms developed in Section 3 and 4 are coded in C++ and performed at PC with INTEL™ PIII-800 CPU. The input parameters include the locations for the users and the potential switches, admissible configurations and cost structures of potential switches and links, traffic requirements and survivability/connectivity requirements. And the output parameters include the switch and link configuration assignment, routing assignment, node disjoint paths assignment, average end-to-end delay and individual end-to-end delay for each O-D pair.

The maximum number of iterations for the proposed dual Lagrangean algorithm developed above are 1000, and the improvement counter is 30. The step size for the dual Lagrangean algorithm is initialized to be 2 and be halved of its value when the objective value of the dual algorithm does not improve for 30 iterations.

Two sets of computational experiments are performed. The computational time for these two sets of computational experiments are all within fifteen minutes under the network size of 30 user/switch nodes. Hence, the proposed algorithms are efficient in time complexity.

In these computational experiments, the cost of the link assignment is divided into two parts, fixed cost and variable cost. The fixed cost is calculated from the Euclidean distance between two end points that the link connected, and the variable cost is based on the link capacity configuration. There are fifteen discrete potential link capacity configurations, from 0 to 500, for the computational experiments. And the cost associated with these potential capacity configurations is a concave function to reflect the economy-of-scale effect. On the other hand, the switch installation cost is based upon the switching fabric and the number of ports on the switch. There are nineteen discrete potential switch configurations in the computational experiments. And the cost associated with these potential switch configurations is also a concave function.

In the first set of computational experiment, we want to test the solution quality when the input delay requirements are loose as compared to the output of the delay requirement. And node disjoint path requirement is not considered. In the second set of computational experiment, we want to test the solution quality when the input delay requirements are tight as compared to the output of the delay requirement. And the 2-connected node-disjoint-path requirement for each O-D pair is considered.

Table 1 depicts the computational results for the various network sizes and traffic demand without node disjoint requirement and loose delay requirements. The first column is the network size. The location, x-axis and y-axis, of user nodes and potential switch nodes are randomly distributed between the 0 and 500. The second column is the traffic demand of the user nodes, are randomly distributed between 30 to 400. The third column reports the lower bound of the primal problem. The forth column reports the upper bound of the primal problem. The fifth column reports the error gap between the lower bound and upper bound. The seventh column reports the average network delay requirement, and the sixth column reports the average network delay calculated by the proposed algorithms. As could be seen from the fifth column in Table 1, the more number of user/switch nodes the looser the error gap. And we have a tighter error gap for heavy traffic in the same network topology.

Table 1. Solution quality obtained by various network sizes and traffic demand without node disjoint requirement

# of Users / switches	Traffic Demand	Lower bound	Upper bound	Error Gap(%)	Average network delay	K
9	30~200	943.4	1639.9	73.8	0.168	10
9	60~400	1852.4	2778.0	49.9	0.068	10
12	30~200	1639.8	2664.8	62.5	0.079	10
12	60~400	3048.0	4353.1	42.8	0.069	10
15	30~200	1895.1	2918.9	54.0	0.294	10
15	60~400	3695.3	5291.5	43.2	0.131	10
30	30~200	2496.9	6042.3	141.9	0.397	10
30	60~400	4442.8	7936.1	78.6	0.117	10

Table 2. Solution quality obtained by various network sizes and traffic demand with two node disjoint requirement

# of Users / switches	Traffic Demand	Lower bound	Upper bound	Error Gap(%)	Average network delay	K
9	30~200	941.5	1980.3	110.3	0.195	1
9	60~400	1852.6	2909.0	57.0	0.054	1
12	30~200	1636.2	3220.0	90.7	0.087	1
12	60~400	3048.0	4353.1	42.8	0.069	1
15	30~200	1887.3	3369.6	78.5	0.203	1
15	60~400	3694.9	5667.0	53.4	0.130	1
30	30~200	2472.0	7098.8	187.2	0.452	1
30	60~400	4388.6	8628.5	96.6	0.130	1

Table 2 depicts the computational results for the various network sizes and traffic demand with two node disjoint requirement and loose delay requirements. As compared to Table 1, the average cross network delay and the end-to-end delay requirements are more stringent, delay = 1 instead of 10. On the other hand, there are two node disjoint requirement for each user nodes. As could be seen from the third column of Table 2, we have a looser lower bound in Table 2. However, we still have a reasonable good upper bound in Table 2.

6 Concluding Remarks

In this paper, the mathematical formulation and algorithms for the backbone network design which considers the system and user specified QoS requirements are proposed. The objective of this backbone network design problem is to minimize the total installation cost of link and switch installation cost. The system QoS requirement is the average cross network delay requirement. The user specified QoS requirements include the end-to-end delay requirement and node disjoint path requirements. Besides integrality constraints, the non-convexity of the delay performance metric makes this problem difficult. By using the Lagrangean relaxation method and the subgradient method to construct the dual problem and calculate the tightest lower bound, we provide getting primal feasible solution heuristic to obtain the primal feasible solution based on the solutions to the dual problem. Based on the solution quality and the computational time, we propose effective and efficient algorithms for this problem.

References

1. Cheng, K. T. and F. Y-.S. Lin, "Minimax End-to-End Delay Routing and Capacity Assignment for Virtual Circuit Networks," Proc. IEEE Globecom, (1995) 2134-2138.
2. Cheng, S. T., "Topological Optimization of a Reliable Communication Network", IEEE Trans. on Reliability, Vol. 47, No. 3, Sep. (1998) 225-233.
3. Gavish, B., "Topological design of computer communication networks – The overall design problem", European J. of Operational Research, 58, (1992) 149-172.
4. Lin, F. Y-.S., "Link Set Sizing for Networks Supporting SMDS", IEEE/ACM Trans. on Networking, Vol. 1, No. 6, Dec. (1993) 729-739.
5. Ombuki, B., M. Nakamura, Z. Nakao and K. Onaga, "Evolutionary Computation for Topological Optimization of 3-Connected Computer Networks", IEEE International Conference on Systems, Man, and Cybernetics, Vol. 1, (1999) 659-664.
6. Pierre, S. and A. Elgibaoui, "A Tabu-Search Approach for Designing Computer-Network Topologies with Unreliable Components", IEEE Trans. on Reliability, Vol. 46, No. 3, Sep. (1997) 350-359.
7. Suurballe, J. W. and R. E. Tarjan, "A Quick Method for Finding Shortest Pairs of Disjoint Paths", Networks, Vol. 14, (1984) 325-336.