JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 15, 71-89 (1999)

VP Layout Planning in Survivable ATM Networks

CHENG-SHONG WU AND SHI-WEI LEE
Department of Electrical Engineering
National Chung Cheng University
Chiayi, Taiwan 621, R.O.C.

In this paper, survivable VP planning is considered for preplanned restoration in ATM networks. Four restoration schemes are evaluated and compared. Both link protection and node protection are studied. Problems are formulated as several combinatorial optimization problems in which the objective is to minimize the bandwidth usage and the constraints are required to satisfy the survivability, end-to-end hop number and physical capacity limitations. Lagrangian relaxation and subgradient methods are used to obtain heuristic solutions and provide a lower bound to assess the quality of the solutions. Numerical results are reported, and the advantages and disadvantages of different protection schemes are discussed.

Keywords: ATM, virtual path, survivability, subgradient optimization, lagrangian relaxation

1. INTRODUCTION

Due to the fast development of high speed ATM networks, the volume of traffic should increase greatly in the future. Failures of facilities become much more serious than the existing networks. Therefore, it is very important to develop rapid and highly reliable restoration mechanisms to minimize the impact of facility failure.

ATM technology that adopts the Virtual Path (VP) concept has several advantages over STM technology in designing a survivable network [13]. The non-hierarchical structure of the VP layer can simplify path management and improve link resource utilization. Moreover, the path bandwidth and the path route are defined independently. As a result, a virtual path route can be established without any bandwidth being assigned to it. Therefore, it is possible to establish a zero bandwidth backup VP to reduce bandwidth usage.

Basically, there are two methods for restoring a failure path: the preplanned and the dynamic restoration methods. In the preplanned method, each working path prearranges its backup routes in advance [6, 10, 11]. On the other hand, broadcast messages will be generated to search for restoration paths when a failure is detected in the dynamic method [3, 4]. Fig. 1 and 2 illustrate the restoration procedures for the preplanned and dynamic restoration methods. Generally speaking, the preplanned method has faster restoration but requires a larger data base. In addition, the backup paths need to be replanned when the network topology changes in the preplanned method. A prototype for implementing the VP based preplanned self-healing scheme was reported in [6].

Received November 15, 1996; accepted December 8, 1997.
Communicated by Jean-Lien C. Wu.

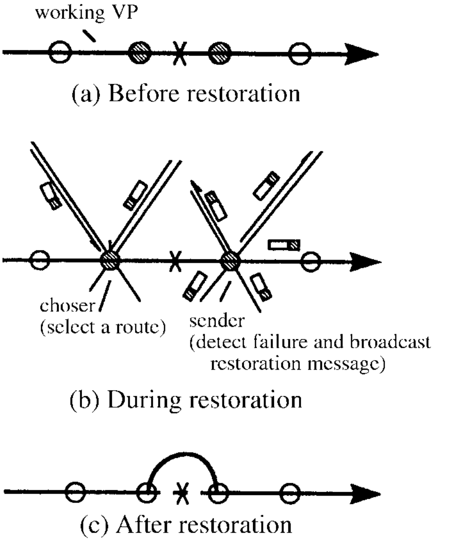


Fig. 1. Restoration procedure for the dynamic restoration method.

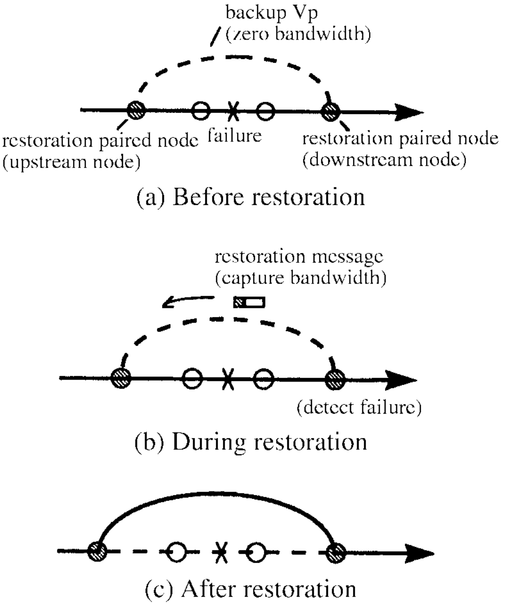


Fig. 2. Restoration procedure the for preplanned restoration method.

The dynamic method searches for a backup path, captures the bandwidth only when a failure happens and shares spare bandwidth resources automatically. In the ATM VP based preplanned method, a backup VP can be established without any bandwidth in advance and the bandwidth is assigned to the corresponding backup VP only when a failure happens. In this way, the preplanned method can share the spare capacity as efficiently as the dynamic method does. In order to guarantee success in restoration and to minimize the spare resource reserved for restoration, careful resource management is needed in both the dynamic and preplanned methods.

In [11], a heuristic spare resource assignment algorithm was proposed under the condition that the working paths are already deployed in the networks. However, the quality of the heuristic was not discussed in their paper. Murakami and Kim [10] modeled the survivable virtual path problem with link failure protection as a multicommodity flow problem and adopted the continuous flow approach by assuming that the bandwidth of each VP is considerably smaller than the link capacity. This assumption can be viewed as integer relaxation to original problem. However, this assumption is not realistic in practice because the VP to link capacity ratio cannot usually be neglected. In [5], working and backup routes were planned to protect any single link failure. However, only the path restoration scheme is considered to protect single link failure. The objective function was to minimize the average packet delay under normal and failure states.

In this paper, we consider the working VP assignment and backup VP routing problems jointly and employ the integer programming based approach to maximize system resource utilization. Unlike the packet switch network used in [5] to minimize the average delay, the ATM network must reserve enough network resources to guarantee quality of service. Therefore, we attempt to minimize the bandwidth usage under constraints that satisfy the survivability, end-to-end hop number and physical capacity limitations. Two different survivability requirements are considered: link failure survivability and node failure survivability. Link failure survivability requires that the network is guaranteed to survive any single link failure whereas the node failure requires that the network can survive any node failure. The solutions to the problems can be used not only to determine the working VP layout and the backup VP routing in the preplanned method, but also to reserve enough spare bandwidth for restoration when failure occurs in the dynamic method.

Four restoration schemes are considered in this paper: two path based schemes, one link based scheme and one hop node scheme. Although the path based and link based schemes are conventional, there are no information in the literature that can be used to compare their performance in the ATM VP environment. The newly proposed hop-node scheme can protect against node failures.

Due to the combinatorial nature of the formulations, trying to find exact optimal solutions to these problems is unrealistic. Lagrangian relaxation is very useful for approximately solving large integer programming problems and has been applied to solve various network design and resource allocation problems [1, 5, 7, 9]. Therefore, Lagrangian relaxation is used here to obtain heuristic solutions and provides a lower bound to assess the quality of the solution in this paper.

The remainder of this paper is organized as follows. In section 2, schemes for link protection and node protection are discussed. In section 3, problem formulations for different schemes are presented. The Lagrangian relaxation solution procedure used to obtain lower bounds and heuristic upper bounds for the problems are discussed in section 4. In section 5, we present computational results and draw comparisons between the different schemes. Finally, we close the paper with a brief conclusion in the last section.

2. THE LINK AND THE NODE PROTECTION SCHEMES

Theoretically, both preplanned and dynamic methods can be used to establish an alternative route (backup VP) between any two restoration paired nodes to bypass a failure. The dynamic method is usually used to restore a failure between the adjacent paired nodes of a faulty link. The downsteam restoration paired node is called the sender and the upstream one called the choser. Because multiple sender-choser pairs are involved and conflicts in path searching and bandwith capture will occurs between different sender-choser pairs, the restoration process is much more complicated in the dynamic method for protection against node failure or restoration of a path between the virtual path terminators. Azuma et al.[3] described a dynamic method for protection against node failure. On the other hand, in the preplanned method, a backup path can be preassigned between any pair of nodes on a working VP route. Therefore, protection a node or end-to-end VP path is as easy as protecting a link from the operation point of view.

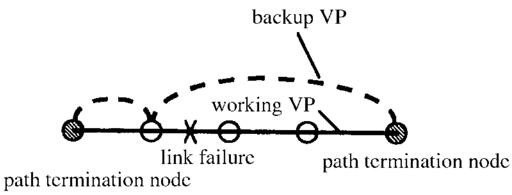
Two different survivability criteria are considered in this paper: one for link failure protection and one for node failure protection. The first one, called link failure survivablity, requires that the VP layout be designed to in sure survival of any single link failure. That is, if a link failure occurs, all the affected working VPs reroute it traffic to bypass the failure using either the dyamical or preplanned method. The second one, called node failure survivablity, requires that the nework be able to survive any single node failure.

For each survivability criteria, we use two policies to select the restoration paired nodes for backup paths. The first policy is to select a backup path that can protect the longest path segment of a working VP and the second one is to select a shortest backup path between a node pairs. The first policy is expected to provide better spare resource utilization, but the second one is expected to provide faster restoration.

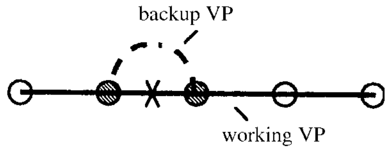
For link failure protection, the first policy selects the two virtual path termination nodes as the restoration paired nodes, and the second policy selects the adjacent nodes of a link as the restoration node pair. The former is called the path restoration for protection scheme or the path-link scheme. The latter, which is the link restoration for link protection scheme is called the link-link scheme. For node failure protection, the restoration node pair of the first policy is the same as the path-link scheme whereas the second policy selects nodes two-hops away from each other in the corresponding working VP as the restoration node pair. These two schemes are referred to below as the path restoration for node protection scheme (path-node scheme) and the two-hop node protection scheme (2-hop-node scheme), respectively. Fig. 3 depicts the differences between these four schemes. One major difference between the path-link and path-node schemes is that in the former, the backup path disjoins the working path only in the links whereas in the latter, the backup path and the working path are completely different both in links and nodes except for the two path termination nodes.

3. PATH ASSIGNMENT MODELS

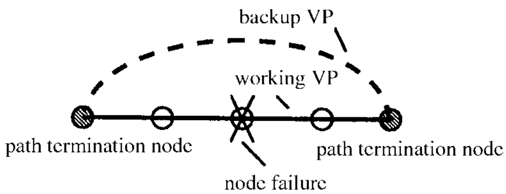
In this section, the problems of determining working VPs and backup VPs in the path-link, the link-link, the path-node and 2-hop-node schemes are formulated as nonlinear combinatorial optimization problems. The objective is minimization of both the working and backup VPs' bandwidth usage, and the constraints are required to satisfy the



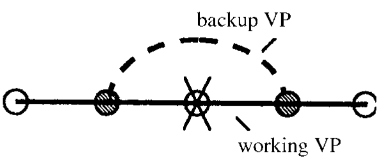
(a) Path-link protection scheme



(b) Link-link protection scheme



(c) Path-node protection scheme



(d) 2-hop-node protection scheme

Fig. 3. Link and node protection schemes.

survivability, end-to-end hop number and physical capacity limitations. The inputs of theproblems are the network topology, link capacity and traffic requirement, and the outputs are the working path route, the backup path route and the bandwidth usage in each link.

3.1 Path Restoration for the Link Protection (Path-Link) Scheme

The ATM network is modeled as a graph $G(V,L,C)$, where node set V is the set of switch nodes, link set L represents the set of physical links and $C_i \in C$ is the physical capacity of link i, $i \in L$. Furthermore, we let K be the collection of the Original Destination (OD) pairs in the network and b_k be the bandwidth requirement for traffic between OD pair $k \in K$. $\omega 1_i$ is used to denote the per unit capacity cost used by the working paths on link i, and $\omega 2_i$ denotes, the per unit capacity cost used by the backup paths on link i. Our formulatio minimizes the total working and backup VPs' bandwidth cost. The following notations are also used in the formulation.

P_k : Set of possible candidate virtual paths for OD pair $k \in K$.
$\alpha_{k,p}$: Virtual path routing variables, 1 if OD pair $k \in K$ uses path $p \in P_k$; 0 otherwise.
$\beta_{k,q}$: Backup virtual path routing variable, 1 if OD pair $k \in K$ uses path $q \in P_k$; 0 otherwise.
$\delta^i_{k,p}$: Link-path incidence matrix, 1 if path $p \in P_k$ for OD pairs $k \in K$ uses link $i \in L$; 0 otherwise.
H_k : Hop limit for the working path and the backup path of OD pair $k \in K$.

The path-link scheme tries to select a working path and a link-disjointed backup path between any OD pair.

The following model, to be referred to as Problem (*PL*), can be used to determine the working VPs' and backup VPs' routes in a survivable ATM network using the path-link protection scheme. Note that constraint (PL.4) explicitly assigns to working VPs the required bandwidth while constraints (PL.5) assigns the bandwidth to backup VPs only when a failure happens to achieve spare capacity sharing.

Problem (*PL*)

$$Z_{PL} = \min\left[\sum_{k\in K}\sum_{p\in P_k}\sum_{i\in L}\omega 1_i \times b_k \times \alpha_{k,p} \times \delta^i_{k,p} + \sum_{k\in K}\sum_{q\in P_k}\sum_{j\in L}\omega 2_j \times b_k \times \beta_{k,q} \times \delta^j_{k,q}\right]$$

subject to:

$$\sum_{p\in P_k} \alpha_{k,p} = 1, \forall k \in K \quad \text{(PL.1)}$$

$$\sum_{q\in P_k} \beta_{k,q} = 1, \forall k \in K \quad \text{(PL.2)}$$

$$\alpha_{k,p} \times \delta^i_{k,p} \times \beta_{k,q} \times \delta^i_{k,q} = 0, \forall_i \in L, k \in K, p,q \in P_k \quad \text{(PL.3)}$$

$$\sum_{k\in K} b_k \times \sum_{p\in P_k} \alpha_{k,p} \times \delta^j_{k,p} \le C_j, \forall_j \in L \quad \text{(PL.4)}$$

$$\sum_{k\in K} b_k \times \sum_{p\in P_k} \alpha_{k,p} \times \delta^j_{k,p} + \sum_{k\in K} b_k \times \sum_{p\in P_k}\sum_{q\in P_k} \alpha_{k,p} \times \delta^i_{k,p} \times \beta_{k,q} \times \delta^j_{k,q} \quad \text{(PL.5)}$$

$$- \sum_{k\in K} b_k \times \sum_{p\in P_k} \alpha_{k,p} \times \delta^i_{k,p} \times \delta^j_{k,p} \le C_j, \forall_i, j \in L, i \ne j$$

$$\sum_{i\in L} \alpha_{k,p} \times \delta^i_{k,p} \le H_k, \forall k \in K, p \in P_k \quad \text{(PL.6)}$$

$$\sum_{i\in L} \beta_{k,q} \times \delta^i_{k,q} \le H_k, \forall k \in K, q \in P_k \quad \text{(PL.7)}$$

$$\alpha_{k,p} = 0 \quad or \quad 1, \forall k \in K, p \in P_k \quad \text{(PL.8)}$$

$$\beta_{k,q} = 0 \quad or \quad 1, \forall k \in K, q \in P_k\ . \quad \text{(PL.9)}$$

In Program (*PL*), the variables to be determined are the routing variables $\alpha_{k,p}$ and, $\beta_{k,q}$, and the others are the inputs of the problem. The objective function represents the total capacity cost for working and backup VP routing. Constraints (PL.1) and (PL.8) are used to choose a working VP for each OD pair. Constraints (PL.2) and (PL.9) are used to choose a backup VP for each working VP. Constraint (PL.3) says that the working VP and its backup VP must be link disjointed. The total capacity for working VPs on a link should be less than the physical capacity is expressed in constraints (PL.4). These constraints guarantee that the working VP of each OD pair acquires enough bandwidth. Constraint (PL.5) states that the flow on link *j* must not exceed the link capacity after a failure on link *i* happens. This constraint make sure that the backup VP can get enough bandwidth after a link failure on link *i*. The first term in constraint (PL.5) is the original working traffic on link *j* , the second term is the increase in traffic because of traffic rerouting from the faulty link *i* to link *j* and the last term is the decrease in working traffic on link *j* because of traffic rerouting. Constraints (PL.6) and (PL.7) limit the working VP hop count and backup VP hop count.

3.2 Link Restoration for the Link Protection (Link-Link) Scheme

The following problem, Problem (*LL*), routes the working VPs and backup VPs using link restoration for the link protection scheme. In addition to the previous notations, the new definitions for Problem (*LL*) are:

L_p : Set of links used by working path *p*.
L_q : Set of links used by backup path *q*.
L_k : Set of links used by the working candidate path for OD pair $k \in K$.
Q_l : Set of candidate backup paths for link $l \in L$.
$\beta_{k,l,q}$: Backup virtual path routing variable, 1 if for OD pair $k \in K$ uses path $q \in Q_l$ of link $l \in L$; 0 otherwise.
$\gamma^i_{l,q}$: 1 if backup path $q \in Q_l$ uses link *i*; 0 otherwise.

Problem (*LL*)

$$Z_{LL} = \min[\sum_{k\in K}\sum_{p\in P_k}\sum_{i\in L}\omega1_i \times b_k \times \alpha_{k,p} \times \delta^{\mathrm{i}}_{\mathrm{k,p}} + \sum_{k\in K}\sum_{j\in L}\sum_{q\in Q_j}\sum_{i\in L_q}\omega2_i \times b_k \times \beta_{k,j,q} \times \gamma^i_{j,q}]$$

subject to:

$$\sum_{p\in P_k}\alpha_{k,p} = 1, \forall k \in K \quad \text{(LL.1)}$$

$$\sum_{q\in Q_l}\beta_{k,l,q} = \alpha_{k,p}, \forall k \in K, p \in P_k, l \in L_p \quad \text{(LL.2)}$$

$$\alpha_{k,p} \times \delta^i_{k,p} \times \beta_{k,l,q} \times \gamma^i_{l,q} = 0, \forall i \in L, k \in K, p \in P_k, l \in L_p, q \in Q_l \quad \text{(LL.3)}$$

$$\sum_{k\in K} b_k \times \sum_{p\in P_k}\alpha_{k,p} \times \delta^j_{k,p} + \sum_{k\in K} b_k \times \sum_{q\in Q_i}\beta_{k,i,q} \times \gamma^j_{i,q} \le C_j, \forall\, i, j \in L \quad \text{(LL.4)}$$

$$\sum_{i\in L}\alpha_{k,p} \times \delta^i_{k,p} \le H_k, \forall k \in K, p \in P_k \quad \text{(LL.5)}$$

$$\sum_{i\in L_p}\alpha_{k,p}\times\delta_{k,p}^{i}+\sum_{j\in L_q}\beta_{k,l,q}\times\gamma_{l,q}^{j}-1\le H_k,\forall k\in K,p\in P_k,l\in L_p,q\in Q_l \quad \text{(LL.6)}$$

$$\alpha_{k,p}=0 \;\; or \;\; 1,\forall k\in K,p\in P_k \quad \text{(LL.7)}$$

$$\beta_{k,l,q}=0 \;\; or \;\; 1,\forall k\in K,l\in L_k,q\in Q_l\,. \quad \text{(LL.8)}$$

In Problem (*LL*), the objective function represents the total capacity cost for working VP assignment and backup VP routing. Constraints (LL.1) and (LL.7) are the same as those in (*PL*). Constraints (LL.2) and (LL.8) are used to choose a backup virtual path for each link if the chosen working VP uses the link. Constraint (LL.3) says the backup VP and the link protected by it must be link-disjointed. Constraint (LL.4) is the physical capacity constraint for every link. Constraint (LL.5) is the working VPs' hop count constraint, and constraint (LL.6) limits the hop count after restoration.

3.3 Path Restoration for the Node Protection (Path-Node) Scheme

This scheme is similar to the path-link protection schemes. The major difference is that the working path and the corresponding backup path must both be link-disjoined and node-disjoined. Problem *(PN)* in the following is the model for minimizing the bandwidth usage in a survivable network using this schemes. Some more notations are needed in the model.

$\psi_{k,p}^{v}$: 1 if working path p of OD pair k uses node v.
s_k: source node of OD pair $k\in K$.
d_k: destination node of OD pair $k\in K$.

Problem (*PN*)

$$Z_{PN}=\min[\sum_{k\in K}\sum_{p\in P_k}\sum_{i\in L}\omega 1_i\times b_k\times\alpha_{k,p}\times\delta_{k,p}^{i}+\sum_{k\in K}\sum_{q\in P_k}\sum_{j\in L}\omega 2_j\times b_k\times\beta_{k,q}\times\delta_{k,q}^{j}]$$

subject to:

$$\sum_{p\in P_k}\alpha_{k,p}=1,\forall k\in K \quad \text{(PN.1)}$$

$$\sum_{q\in P_k}\beta_{k,q}=1,\forall k\in K \quad \text{(PN.2)}$$

$$\alpha_{k,p}\times\delta_{k,p}^{i}\times\beta_{k,q}\times\delta_{k,q}^{i}=0,\forall\, i\in L,k\in K,p,q\in P_k \quad \text{(PN.3)}$$

$$\sum_{k\in K}b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta_{k,p}^{j}\le C_j,\forall j\in L \quad \text{(PN.4)}$$

$$\sum_{k\in K}b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta_{k,p}^{j}+\sum_{k\in K}b_k\times\sum_{p\in P_k}\sum_{q\in P_k}\alpha_{k,p}\times\delta_{k,p}^{i}\times\beta_{k,q}\times\delta_{k,q}^{j} \quad \text{(PN.5)}$$

$$-\sum_{k\in K}b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta_{k,p}^{i}\times\delta_{k,p}^{j}\le C_j,\forall\, i,j\in L,i\ne j$$

$$\sum_{k\in K} b_k \times \sum_{p\in P_k} \alpha_{k,p} \times \delta_{k,p}^{j} + \sum_{k\in K} b_k \times \sum_{p\in P_k} \sum_{q\in P_k} \alpha_{k,p} \times \psi_{k,p}^{v} \times \beta_{k,q} \times \delta_{k,q}^{j} \tag{PN.6}$$

$$- \sum_{k\in K} b_k \times \sum_{p\in P_k} \alpha_{k,p} \times \delta_{k,p}^{i} \times \delta_{k,p}^{j} \le C_j, \forall v\in V, i\in L, j\in L, i\ne j$$

$$\sum_{i\in L} \alpha_{k,p} \times \delta_{k,p}^{i} \le H_k, \forall k\in K, p\in P_k \tag{PN.7}$$

$$\sum_{i\in L} \beta_{k,q} \times \delta_{k,q}^{i} \le H_k, \forall k\in K, q\in P_k \tag{PN.8}$$

$$\alpha_{k,p} \times \psi_{k,p}^{v} \times \beta_{k,q} \times \psi_{k,q}^{v} = 0, \forall k\in K, p,q\in P_k, v\in V-\{s_k, d_k\} \tag{PN.9}$$

$$\alpha_{k,p} = 0 \quad or \quad 1, \forall k\in K, p\in P_k \tag{PN.10}$$

$$\beta_{k,q} = 0 \quad or \quad 1, \forall k\in K, q\in P_k\ . \tag{PN.11}$$

In this model, the working and backup paths are link-disjointed as well as node-disjointed except at the source or destination nodes. This scheme is also a path restoration scheme, and the formulation is similar to Problem (*PL*). However, we need to add the node-disjointed constraint (PN.9) and constraint (PN.6) to make sure that the flow on link *j* does not exceed the physical capacity when node *v* fails.

3.4 Two-Hop Node Protection (2-Hop-Node) Scheme

This scheme tries to set up a backup path between two hop away node pairs along the working VP (see Fig. 3). In addition to the common notations used in the previous schemes, the following notations are also needed to model this scheme.

N : Set of Protected working Path Segments (PPSs).
Q_n : Set of candidate backup paths for PPS $n \in N$.
$\gamma_{n,q}^{i}$: 1 if backup path $q \in Q_n$ use link i; 0 otherwise.
$\varphi_{l,q}^{i}$: 1 if backup path $q \in Q_l$ use link i; 0 otherwise.
$\beta_{k,n,q}$: backup path routing variable, 1 if OD pair k use backup path $q \in Q_n$; 0 otherwise.
$\chi_{k,l,q}$: backup path routing variable, 1 if OD pair k use backup path $q \in Q_l$; 0 otherwise.
N_υ : set of PPSs that contain node υ as the intermediate nodes.
N_p : set of PPSs on working path p.
L_p : set of links used by path p.
V_q : set of nodes on back up path q.
$\xi_{n,q}^{v}$: 1 if $\upsilon \in V_q$, $q \in Q_n$; 0 otherwise.
f_q : first node of backup path q.
l_q : last node of backup path q.

Note that the protected working path segment (PPS) is the protected node together with its two adjacent links and two adjacent nodes along the working path (Fig. 4).

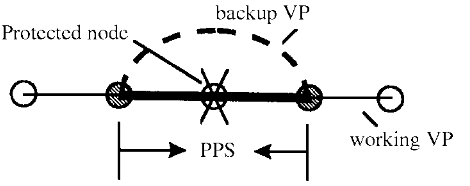


Fig. 4. Protected working path segment (PPS).

Problem (*HN*)

$$Z_{HN} = \min[\sum_{k\in K}\sum_{p\in P_k}\sum_{i\in L}\omega 1_i \times b_k \times \alpha_{k,p} \times \delta^i_{k,p}$$

$$+\sum_{k\in K}\sum_{j\in L}\omega 2_j \times b_k \times (\sum_{n\in N}\sum_{q\in Q_n}\beta_{k,n,q}\times\gamma^j_{n,q} + \sum_{l\in L}\sum_{q\in Q_l}\chi_{k,l,q}\times\varphi^j_{l,q})]$$

subject to:

$$\sum_{p\in P_k}\alpha_{k,p} = 1, \forall k\in K \quad \text{(HN.1)}$$

$$\sum_{q\in Q_n}\beta_{k,n,q} = \alpha_{k,p}, \forall k\in K, p\in P_k, n\in N_p, |p|>1 \quad \text{(HN.2)}$$

$$\sum_{q\in Q_l}\chi_{k,l,q} = \alpha_{k,p}, \forall k\in K, p\in P_k, l\in L_p, |p|=1 \quad \text{(HN.3)}$$

$$\alpha_{k,p}\times\delta^i_{k,p}\times\beta_{k,n,q}\times\gamma^i_{n,q} = 0, \forall i\in L, k\in K, p\in P_k, n\in N_p, q\in Q_n, |p|>1 \quad \text{(HN.4)}$$

$$\alpha_{k,p}\times\delta^i_{k,p}\times\chi_{k,l,q}\times\varphi^i_{l,q} = 0, \forall i\in L, k\in K, p\in P_k, l\in L_p, q\in L_n, |p|=1 \quad \text{(HN.5)}$$

$$\alpha_{k,p}\times\psi^v_{k,p}\times\beta_{k,n,q}\times\xi^v_{n,q} = 0, \quad \text{(HN.6)}$$

$$\forall v\in V_q - \{f_q, l_q\}, k\in K, p\in P_k, n\in N_p, q\in Q_n, |p|>1$$

$$\sum_{k\in K} b_k \times \sum_{p\in P_k}\alpha_{k,p}\times\delta^j_{k,p} + \sum_{k\in K} b_k \times \sum_{n\in N_v}\sum_{q\in Q_n}\beta_{k,n,q}\times\gamma^j_{n,q} \le C_j, \forall v\in V, j\in L \quad \text{(HN.7)}$$

$$\sum_{k\in K} b_k \times \sum_{p\in P_k}\alpha_{k,p}\times\delta^j_{k,p} + \sum_{k\in K} b_k \times \sum_{q\in Q_l}\chi_{k,l,q}\times\varphi^j_{l,q} \le C_j, \forall i,j\in L \quad \text{(HN.8)}$$

$$\sum_{i\in L}\alpha_{k,p}\times\delta^i_{k,p} \le H_k, \forall k\in K, p\in P_k \quad \text{(HN.9)}$$

$$\sum_{i\in L}\alpha_{k,q}\times\delta^i_{k,q} + \sum_{j\in L_q}\chi_{k,l,q}\times\varphi^j_{l,q} - 1 \le H_k, \forall k\in K, p\in P_k, l\in L_p, |p|=1 \quad \text{(HN.10)}$$

$$\sum_{i\in L}\alpha_{k,q}\times\delta^i_{k,q} + \sum_{j\in L_q}\beta_{k,n,q}\times\gamma^j_{n,q} - 2 \le H_k, \forall k\in K, q\in P_k, n\in N_p, |p|>1 \quad \text{(HN.11)}$$

$$\alpha_{k,p} = 0 \text{ or } 1, \forall k\in K, p\in P_k \quad \text{(HN.12)}$$

$$\beta_{k,n,q} = 0 \text{ or } 1, \forall k \in K, n \in N_k, q \in Q_n \quad \text{(HN.13)}$$

$$\chi_{k,l,q} = 0 \; or \; 1, \forall k \in K, l \in L_k, q \in Q_n \quad \text{(HN.14)}$$

In model (*HN*), the objective function represents the total capacity cost of working VP assignment. Constraints (HN.1) and (HN.12) are used to choose a working virtual path for each OD pair. Constraints (HN.2), (HN.3), (HN.13) and (HN.14) are used to choose a backup virtual path to protect a node if the chosen working VP path passes through the node. Constraints (HN.4), (HN.5) and (HN.6) say that working PPS and backup VP must be link-disjointed and node-disjointed. Constraints (HN.7) and (HN.8) are the physical capacity constraints for every link. Constraints (HN.3), (HN.14), (HN.5), (HN.8) and (HN.10) are the backup link protected in degenerate cases in which the working paths are single-link paths. Constraint (HN.9) is the working VP hop count limitation and constraints (HN.10) and (HN.11) limit the working path hop count and backup path hop count.

4. ALGORITHM AND SOLUTION PROCEDURE

Problems (*PL*), (*LL*), (*PN*), and (*HN*) are nonlinear 0,1 integer programming problems. Due to the combinatorial nature of these problems, it is not practical to solve them directly using methods such as the branch-and-bound method. The integer relaxation technique is not an appropriate solution approach because only 0,1 in the integer solution. Instead, we use Lagrangian relaxation to obtain tight lower bounds of objective functions and provide heuristics to obtain a sub-optimal feasible solution which is the upper bound of the problems. The solution procedures for this four problems are similar. Thus, only the solution procedure of Problem (*PL*) will be reported in this paper.

4.1 Lower Bounds for Problem (PL)

We first dualize constraints (PL.4) and (PL.5) of (PL) and obtain the following Lagrangian relaxation of Program (*PL*):

Problem (PL_{Lag})

$$L_{PLL}(u,\upsilon) = min\{ \sum_{k\in K}\sum_{p\in P_k}\sum_{i\in L}\omega 1_i \times b_k \times \alpha_{k,p} \times \delta^i_{k,p} + \sum_{k\in K}\sum_{q\in P_k}\sum_{j\in L}\omega 2_j \times b_k \times \beta_{k,q} \times \delta^j_{k,q}$$

$$+ \sum_{i\in L}\sum_{j\in L} u_{i,j} [\sum_{k\in K} b_k \times \sum_{p\in P_k}\alpha_{k,p} \times \delta^i_{k,p} + \sum_{k\in K} b_k \times \sum_{p\in P_k}\sum_{q\in P_k}\alpha_{k,p} \times \delta^i_{k,p} \times \beta_{k,q} \times \delta^j_{k,q}$$

$$- \sum_{k\in K} b_k \times \sum_{p\in P_k}\alpha_{k,p} \times \delta^i_{k,p} \times \delta^j_{k,p} - C_j] + \sum_{j\in L}\upsilon_j [\sum_{k\in K} b_k \times \sum_{p\in P_k}\alpha_{k,p} \times \delta^j_{k,p} - C_j]\}.$$

subject to constraints (PL.1), (PL.2), (PL.3), (PL.6), (PL.7), (PL.8) and (PL.9), where $u = (u_{i,j})$ and $\upsilon = (\upsilon_j)$ are the Lagrangean multipliers. To solve Problem (PL_{Lag}), we can decomposite (PL_{Lag}) into subproblem ($PL_{Lag,k}$) for each OD pair *k* because the variables are separable for *k*.

Problem ($PL_{Lag,k}$)

$$L_k(u,\upsilon)=min\{\sum_{p\in P_k}\sum_{i\in L}\omega 1_i\times b_k\times\alpha_{k,p}\times\delta^i_{k,p}+\sum_{q\in P_k}\sum_{j\in L}\omega 2_j\times b_k\times\beta_{k,q}\times\delta^j_{k,q}$$

$$+\sum_{i\in L}\sum_{j\in L}u_{i,j}[b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta^j_{k,p}+b_k\times\sum_{p\in P_k}\sum_{q\in P_k}\alpha_{k,p}\times\delta^i_{k,p}\times\beta_{k,q}\times\delta^j_{k,q}$$

$$-b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta^i_{k,p}\times\delta^j_{k,p}-C_j]+\sum_{j\in L}\upsilon_j[b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta^j_{k,p}-C_j]\}.$$

If we prepare working and backup candidate paths whose hop counts satisfy the hop count limit for each OD pair requirement, then constraints (PL.6) and (PL.7) can be canceled by implicitly involving them in the candidate path set. The solution to Problem ($PL_{Lag,k}$) is easily obtainable by choosing the shortest link-disjointed working candidate path and backup candidate path between OD pair k. In order to speed up the solution time, the link-disjointed backup paths for any candidate working path can be computed in advance and stored in a data base.

According to the weak Lagrangean duality theorem, the optimal objective function value of (PL_{Lag}), $L_{PLL}(u,\upsilon)$, is a lower bound on the object function value of the primal Program (PL) for any $u_{i,j}\geq 0$ and $\upsilon_j\geq 0$. Naturally, we would like to detemine the greatest lower bound by solving the dual problem of (PL).

Problem (PL_{dual})

$$L_{PLD}=\max_{u\geq 0,\upsilon\geq 0}L_{PLL}(u,\upsilon).$$

The Lagrangian dual Program PL_{dual} is nonsmooth. We use the subgradient method to solve it. In iteration t of the subgradient optimization procedure, the multiplier vector $u^t=(u^t_{i,j})$ and $\upsilon^t=(\upsilon^t_j)$ is updated by $u^t=max\{0,u^{t-1}+\theta^{t-1}\pi\}$ and, $\upsilon^t=max\{0,\upsilon^{t-1}+\theta^{t-1}\rho\}$ where $\pi=(\pi_{i,j}),\rho=(\rho_j)$:

$$\pi_{i,j}=\sum_{k\in K}b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta^j_{k,p}+\sum_{k\in K}b_k\times\sum_{p\in P_k}\sum_{q\in P_k}\alpha_{k,p}\times\delta^i_{k,p}\times\beta_{k,q}\times\delta^j_{k,q}$$

$$-\sum_{k\in K}b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta^i_{k,p}\times\delta^j_{k,p}-C_j$$

$$\rho_j=\sum_{k\in K}b_k\times\sum_{p\in P_k}\alpha_{k,p}\times\delta^i_{k,p}-C_j.$$

The step size θ^t is determined by $\theta^t=\lambda^t\dfrac{(UB-L_{PLL}(u,\upsilon))}{\|\pi\|^2+\|\rho\|^2}$. UB is an upper bound on the optimal objective function value Z^*_{PL} of the primal problem (PL), and also an upper bound on (PL_{dual}). λ^t is a constant between 0 and 2. In our implementation, coded in the C language, we set the maximum dual iteration to 2000 with an initial λ^t equal to 2. Whenever the dual objective function does not improve over 50 iterations, λ^t is assigned by $\lambda^t/2$.

4.2 Upper Bound (Heuristic Solution) of Problem (PL)

At each iteration during implementation of the subgradient solution procedure for Problem (PL_{dual}), we collect some good working path and backup path sets for a given Lagrangian multiplier (u, v). The collection is saved and checked for feasibility for the constraints of primal problem. As the algorithm proceeds, if it generates a more feasible (i.e., lower cost) solution, we substitute the solution into the upper bound UB. After solving Problem (PL_{dual}), a lower bound together with a heuristic solution (upper bound) can be obtained. We can further assess the quality of our solution by comparing the gap between the lower and the upper bound. If a feasible primal solution can not be found from our collection after solving (PL_{dual}), it strongly suggested that the existing network rescue can not accommodate the traffic requirement to achieve 100% survivability. The heuristic solution is then used to calculate the restoration ratio, which will be further defined in the next section. In addition, the heuristic solution can also locate the bottleneck in the network and can be used in capacity replanning.

5. NUMERICAL RESULTS

We applied our models to study the characteristics of the 2-hop-node, the link-link, the path-link, and the path-node schemes. Two network topologies were used in our study. The first network topology shown in Fig. 5 was a GTE network with 12 nodes and 50 links. The second one is the ARPA network with 21 nodes and 52 links (Fig. 6). The topology of the former was denser than that of the latter. We assumed that every link in the GTE network had 5 unit capacity, and that every link in the ARPA network has 10 unit capacity. The OD pairs are randomly generated and each OD pair built one VP with the 1 unit capacity requirement. The candidate paths which satisfied the working and backup hop count constraints were pre-generated.

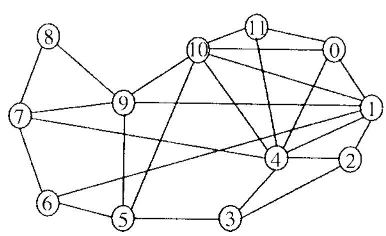


Fig. 5. The GTE network.

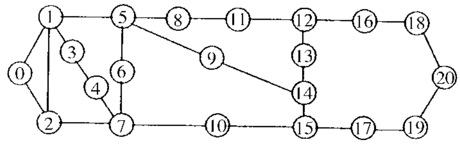


Fig. 6. The PRPA network.

The numerical results are listed in Table 1-Table 4. These tables show the following results : the total number of OD pairs in the network, the solution feasibility, the percentage difference between the upper bound and the lower bound, the total working bandwidth usage (W_BW), the average link restoration ratio (L_RR), the average node restoration ratio (N_RR), the normalized average node restoration ratio (NN_RR), the average link restoration time (L_ART), and the average node restoration time (N_ART). The percentage difference is defined as ([upper-bound – lower-bound] ×100 /lower-bound). The percentage difference was used to assess the quality of our heuristic solutions. The average link (node) restoration ratio is defined as the ratio of the restorable VPs to the VPs affect by a link (node) failure. However, if a failure node is the source or the destination node of a VP, there is no way to restore the VP; therefore, it is impossible to have a 100% node restoration ratio. To make a fair comparison, we defined the normalized average node restoration ratio as the node restoration ratio divided by the maximum node restoration ratio, which is the node restoration ratio for the corresponding network with infinite spare capacity.

Another performance metric, the average node or link restoration time (N_ART, L_ART), is defined as the average number of nodes the restoration message travels through when a failure is detected. For example, in Fig. 7, Node 1 detects a link failure and generates messages to inform the downstream restoration paired node, node 0. The downstream restoration paired node then switches to the backup path and informs the upstream restoration node, node 3, through the backup path. Thus, the restoration related messages are transmitted, received, and processed by nodes 1,0,4,5,6 and 3, and it is reasonable to assume that the restoration time is proportional to the number of nodes involved in the restoration process. Therefore, the restoration time in this case is defined as 6.

First, we note that the values in the columns labeled “Diff%” are zero or very small whenever the solutions are feasible. This indicates the quality of our solution. In fact, we indeed obtained the optimal solution in most of our cases when the value of “Diff%” was equal to zero. When a feasible solution could not be found using the procedure described in section 4.2., this strongly suggested that the networks resources could not satisfy the traffic requirement to achieve 100% NN_RR for node protection schemes or 100% L_RR for all schemes.

Assigning different values of w1 , the weight of a working path, and w2, the weight of a backup path, in our formulations indeed influenced the VP assignment results. For w1/w2=10, the objective function was dominated by the working VP bandwidth utilization. The four schemes had nearly the same working VP routing in both test networks. For w1/w2=1, different schemes had different working bandwidth usage. The working path bandwidth usage for w1/w2=1 was larger than that for w1/w2=10. Comparing these two different weight ratios, the larger w1/w2 ratio resulted in less total working bandwidth usage but longer average restoration time (ART).

Among these four schemes, because trying to select the shortest PPS, the link-link and 2-hop-node restoration schemes had smaller ART values than did the two other path restoration schemes. The ART value was strongly influenced by the network topologies. In the GTE network, the differences in ART among the four schemes were not very significant, but in the ARPA network the differences were more obvious.

Intuitively, the 2-hop-node restoration scheme and link-link restoration scheme should have worse total bandwidth usage than the path-link or path-node schemes. However, the results show that the backup capacity sharing performed so well that the total bandwidth usage for these two schemes was better than that for the two path schemes.

Table 1. ARPA (w1/w2=1, hop limit=15).

	OD	Feasible	Diff%	W_BW	L_RR%	N_RR%	NN_RR%	L_ART	N_ART
Path-link	25	Y	0	111	100	63.2	100	8.108	8.108
Link-link	25	Y	0	94	100	0	0	6.297	–
Path-onde	25	Y	0	111	100	63.2	100	8.108	8.108
2-hop-node	25	Y	0	94	100	57.9	100	6.574	6.420
Path-link	30	Y	0	125	100	61.2	100	7.936	7.936
Link-link	30	Y	0	106	100	0	0	6.358	–
Path-node	30	Y	0	125	100	61.2	100	7.936	7.936
2-hop-node	30	Y	0	106	100	55.8	100	6.613	6.447
Path-link	35	Y	0	143	100	60.6	100	7.741	7.741
Link-link	35	Y	0	116	100	0	0	6.396	–
Path-node	35	Y	0	143	100	60.6	100	7.741	7.741
2-hop-node	35	Y	0	116	100	53.6	100	6.646	6.486
Path-link	40	Y	0.02	180	100	63.6	100	7.616	7.616
Link-link	40	Y	–	131	99.2	0	0	6.603	–
Path-node	40	Y	0.01	180	100	63.6	100	7.616	7.616
2-hop-node	40	Y	0.24	132	100	53.2	100	6.755	6.626
Path-link	45	Y	0.15	186	100	59.7	97.8	7.919	7.919
Link-link	45	N	–	154	98.7	0	0	6.603	–
Path-node	45	N	–	196	79.0	56.4	90.0	7.871	7.871
2-hop-node	45	N	–	159	91.0	50.0	100	7.100	7.078

Table 2. ARPA (w1/w2=10, hop limit=15).

	OD	Feasible	Diff%	W_BW	L_RR%	N_RR%	NN_RR%	L_ART	N_ART
Path-link	25	Y	0	94	100	57.1	98.6	8.755	8.755
Link-link	25	Y	0	94	100	0	0	6.297	–
Path-onde	25	Y	0	94	100	57.9	1	8.914	8.917
2-hop-node	25	Y	0	94	100	57.9	1	6.574	6.420
Path-link	30	Y	0	106	100	55.1	98.7	8.555	8.555
Link-link	30	Y	0	106	100	0	0	6.358	–
Path-node	30	Y	0	106	100	55.8	100	8.698	8.698
2-hop-node	30	Y	0	106	100	55.8	100	6.613	6.447
Path-link	35	Y	0	116	100	52.9	98.6	8.560	8.560
Link-link	35	Y	0	116	100	0	0	6.396	–
Path-node	35	Y	0	116	100	53.6	100	8.698	8.698
2-hop-node	35	Y	0	116	100	53.6	100	6.646	6.486
Path-link	40	Y	0	131	100	52.6	98.8	8.564	8.564
Link-link	40	N	–	131	99.2	0	0	6.511	–
Path-node	40	Y	0	131	100	53.2	100	8.679	8.679
2-hop-node	40	Y	1.99	131	100	53.2	100	7.152	7.05
Path-link	45	Y	0.13	147	100	50.0	94.1	8.721	8.721
Link-link	45	N	–	151	96.0	0	0	6.549	–
Path-node	45	N	–	149	100	53.0	98.8	8.590	8.590
2-hop-node	45	N	–	154	98.0	53.7	100	7.519	7.467

Table 3. GTE (w1/w2=1, hop limit=5).

	OD	Feasible	Diff%	W_BW	L_RR%	N_RR%	NN_RR%	L_ART	N_ART
Path-link	30	Y	0.06	61	100	34.0	100	3.540	3.540
Link-link	30	Y	0	45	100	0	0	3.177	–
Path-onde	30	Y	0.07	61	100	34.0	100	3.540	3.540
2-hop-node	30	Y	0	53	100	2.77	100	3.622	3.173
Path-link	35	Y	0.04	69	100	32.6	100	3.478	3.478
Link-link	35	Y	2.22	52	100	0	0	3.230	–
Path-node	35	Y	0.10	69	100	32.6	100	3.492	3.492
2-hop-node	35	Y	0	61	100	27.0	100	3.557	3.115
Path-link	40	Y	0.49	82	100	34.4	100	3.524	3.524
Link-link	40	Y	0	62	100	0	0	3.209	–
Path-node	40	Y	0.49	82	100	34.4	100	3.524	3.524
2-hop-node	40	Y	0	70	100	27.2	100	3.585	3.166
Path-link	45	Y	0.06	92	100	33.57	97.6	3.608	3.608
Link-link	45	Y	2.21	73	100	0	0	3.479	–
Path-node	45	Y	0.07	92	100	34.3	100	3.619	3.619
2-hop-node	45	Y	0.14	76	100	25.6	100	3.736	3.419
Path-link	50	Y	0.09	101	100	33.7	100	3.600	3.600
Link-link	50	N	–	81	82.7	0	0	3.238	–
Path-node	50	N	–	107	86.9	33.4	91.1	3.579	3.579
2-hop-node	50	Y	0.01	86	100	28.0	100	3.685	3.282

Table 4. GTE (w1/w2=10, hop limit=5).

	OD	Feasible	Diff%	W_BW	L_RR%	N_RR%	NN_RR%	L_ART	N_ART
Path-link	30	Y	0	45	100	20.0	100	3.911	3.911
Link-link	30	Y	0	45	100	0	0	3.177	–
Path-onde	30	Y	0	45	100	20.0	100	3.911	3.911
2-hop-node	30	Y	0	45	100	20.0	100	3.911	3.800
Path-link	35	Y	0	52	100	19.5	100	3.826	3.826
Link-link	35	Y	0.60	52	100	0	0	3.230	–
Path-node	35	Y	0	52	100	19.5	100	3.826	3.826
2-hop-node	35	Y	0	52	100	19.5	100	3.820	3.705
Path-link	40	Y	0	61	100	20.7	100	3.901	3.901
Link-link	40	Y	0.59	61	100	0	0	3.278	–
Path-node	40	Y	0	61	100	20.7	100	3.901	3.901
2-hop-node	40	Y	0	61	100	20.7	100	3.819	3.666
Path-link	45	Y	0.02	72	100	23.0	100	3.930	3.930
Link-link	45	Y	0.70	72	100	0	0	3.527	–
Path-node	45	Y	0.02	72	100	23.0	100	3.930	3.6930
2-hop-node	45	Y	0	72	100	23.0	100	3.833	3.629
Path-link	50	Y	0.19	81	100	23.0	100	3.903	3.600
Link-link	50	N	–	81	91.3	0	0	3.543	–
Path-node	50	Y	0.21	81	100	23.6	100	3.938	3.938
2-hop-node	50	Y	0.17	81	100	23.6	100	3.851	3.612

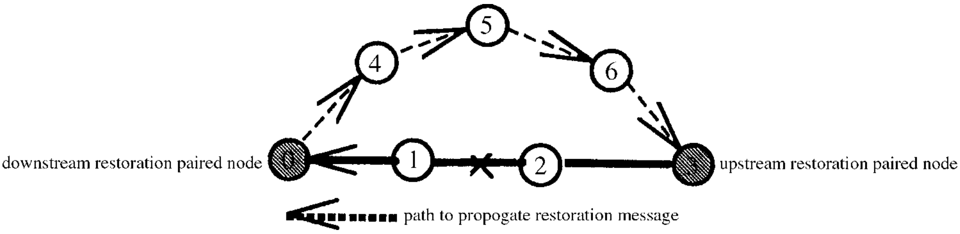


Fig. 7. Average restoration time.

The behaviors of the path-node and the path-link were similar because the feasible solution set of the problem path-node scheme is a subset of the problem of the path-link scheme. However, path-node protection guarantees protection against a node failure because of the node survivability constraint added to its PPS. Since the PPSs of the link-link and 2-hop-node schemes are shorter than those of the path-line and path-node schemes, the link-link and 2-hop-node schemes have better ART than do the two path restoration schemes. The major drawback of the link-link restoration scheme and the 2-hop-node restoration scheme is that a larger database is needed to store the backup paths information. An obvious weakness of link-link scheme is its lack of any ability to protect against node failure.

6. CONCLUSIONS

Survivable VP layout planning has been considered for both link and node restoration schemes. Problems have been formulated as several combinatorial optimization problems in which the objective is to minimize bandwidth usage and constraints are required to satisfy the survivability, end-to-end hop number and physical capacity limitations. The gaps between our heuristic solutions and the lower bounds provided by Lagrangian relaxation show that our heuristic solutions are indeed the optimal solutions or solutions extremely close to the optimal solutions.

Four schemes, namely the path-link, the path-node, the link-link, and the 2-hop-node restoration scheme, have been investigated using our model. From the numerical results, we observe that the 2-hop-node and the link-link restoration schemes have better bandwidth usage and restoration time performance than do the corresponding path based restoration schemes. Because of the addition of a guarantee of protection against node failure, use of the node protection schemes in survivable networks is preferred.

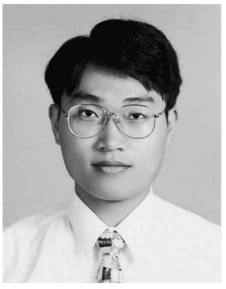
REFERENCES

1. Ravindra K. Ahuja, Thomas L. Magnanti and James B. Orlin, *Network Flow*, Prentice-Hall, 1993.
2. Jon Anderson, Bharat T. Doshi, Subrahmanyam Dravida and P. Harshavardhana, “Fast restoration of ATM networks,” *IEEE Journal on Selected Areas in Communications*, Vol. 12, No. 1, 1994, pp. 128-138.

3. M. Azuma, Y. Fujii, Y. Sato, T. Chujo and K. Murakami, "Network restoration algorithm for multimedia communication services and its performance characteristics," *IEICE Transactions on Communications,* Vol. E78-B, No. 7, 1995, pp. 987-993.
4. H. Fujii and N. Yoshikai, "Restoration message transfer mechanism and restoration characteristics of double-search self-healing ATM network," *IEEE Journal on Selected Areas in Communications*, Vol. 12, No. 1, 1994, pp. 149-158.
5. B. Gavish and I. Neuman, "Routing in a network with unreliable components," *IEEE Transactions on Communications*, Vol. 40, No. 7, 1992, pp. 1248-1258.
6. R. Kawamura, H. Hadama and I. Tokizawa, "Implementation of self-healing function in ATM networks based on virtual path concept," in *Proceeding of IEEE INFOCOM'95*, 1995, pp. 303-311.
7. F. Y. S. Lin, "Link set sizing for networks supporting SMDS," *IEEE Transactions on Networking*, Vol. 1, No. 6, 1993, pp. 729-739.
8. D. Medhi, "A unified approach to network survivability for teletraffic: models, algorithms and analysis," *IEEE Transactions on Communications*, Vol. 42, No. 2/3/4, 1994, pp. 534-547.
9. D. Medhi, "Multi-hour, multi-traffic class network design for VP-based wide-area dynamic reconfigurable ATM networks," in *Proceeding of IEEE INFOCOM'95*, 1995, pp. 900-907.
10. K. Murakami and H. S. Kim, "Virtual path routing for survivable ATM networks," *IEEE Transactions on Networking*, No. 4, No. 1, 1996, pp. 22-39.
11. R. Kawamura, K. Sato and I. Tokizawa, "Self-healing ATM networks based on virtual path concept," *IEEE Journal on Selected Areas in Communications*, Vol. 12, No. 1, 1994, pp. 120-127.
12. K. Sato, H. Ueda and N. Yoshikai, "The role of virtual path crossconnection," *IEEE LCS*, Vol. 2, 1991, pp. 44-54.
13. Tsong-Ho Wu, *Fiber Network Service Survivability,* Artech House, New York, 1992.

Cheng-Shong Wu（吳承崧）was born in Taoyuan, Taiwan on January 14, 1961. He received the B.S. and M.S. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan in 1983 and 1985, respectively, and received the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, U.S.A., in 1991.

He was a Research Assistant in the Department of Electrical Engineering at USC from 1987 to 1991. From 1991 to 1992, he was a visiting assistant professor at the Center for Advanced Computer Studies, the University of Southwestern Louisiana. In 1992, he joined the Institute of Electrical Engineering, National Chung Cheng University, Taiwan, where he is presently an Associate Professor. His research interests include hgih speed communication networks, queueing theory, optical netwoks, and mobile communications.



Shi-Wei Lee（李詩偉）was born in Taipei, Taiwan, on February 26, 1970. He received the B.S. degree in electrical engineering from the Tatung Institute of Technology, Taipei, Taiwan, in 1992. He is currently a Ph.D. student in electrical engineering, National Chung Cheng University, Chyiayi, Taiwan. His research interests include high speed networks, computer network optimization, and network flows.