A Heuristic Approach to Service Restoration in MPLS Networks*

Radim Bartoš and *Mythilikanth Raman*
Department of Computer Science, University of New Hampshire
Durham, NH 03824, USA. Phone: (603) 862-3792, Fax: (603) 862-3493
E-mail: {rbartos,mraman}@cs.unh.edu

***Abstract*– This paper proposes a new approach to providing fault tolerance in MPLS networks based on the concept of "domain protection" where protection paths for all working paths that terminate in an egress router are calculated simultaneously. The proposed scheme guarantees that every protected node is connected to two protection paths placed in a way that no single link failure would cause simultaneous loss of connectivity between a node and the egress router on both protection paths. The use of dual protection paths permits decoupling the protection path placement from the working path placement thus allowing much greater flexibility than other recently proposed schemes. Several heuristics to improve the quality and reduce the cost of the protection path placement are proposed and evaluated. The simulation results show that the algorithm together with the heuristic extensions achieves protection which is less costly or comparable to two recently proposed MPLS protection schemes – RSVP Backup Tunnels and Fast Reroute – while exhibiting comparatively lower algorithmic complexity.**

I. INTRODUCTION

Rapidly increasing volume of traffic carried by the Internet together with imposing requirements for reliability, quality of service, and manageability, force the network technology designers to come up with new approaches and solutions. As the Internet moves towards a IP over WDM model, existing means of network engineering to provide assured bandwidth, quality of service and fault tolerance should be substituted. MultiProtocol Label Switching (MPLS) [1] has emerged as a technology that can provide many of the functionalities now associated with ATM and/or SONET/SDH without incurring much of the overhead.

Current backbone networks rely primarily on the protection in the link layer provided by SONET/SDH and the capability of routing protocols in the network layer to reroute the traffic around the failed link. SONET/SDH is capable of service restoration within few tens of milliseconds, however, the scope of the protection is limited. Standard routing protocols provide much greater degree of flexibility at the cost of restoration time in the order of seconds to minutes. It is generally accepted that desirable failure recovery time should be of the order of tens of milliseconds [2]. MPLS appears to be a suitable place to provide fault tolerance. It is the lowest layer with the knowledge of the entire network topology as well as a point with the necessary traffic engineering capabilities.

The standard goal of MPLS protection schemes, including the one proposed in this paper, is to protect the domain against a single link failure. Provided that the links are not placed in shared conduits, a multiple link failure is a relatively unlikely scenario. Node faults are not currently considered in MPLS protection schemes since routers in backbone networks are typically highly reliable devices with multiple layers of built-in fault protection. The authors are currently investigating extensions to the presented scheme to cover a wider variety of failures as well as to address the issues of resource reservation.

II. BACKGROUND

Two mechanisms have recently been proposed for the restoration of Label Switched Paths (LSP) set up in the MPLS networks, namely the RSVP Backup Tunnels [3] and the Fast Rerouting scheme [4].

Extensions to RSVP [3] have been made to incorporate the concept of LSP tunnels into the RSVP flows. RSVP makes use of the make-before-break concept in rerouting tunnels, i.e., a new alternate path is created before the current path is torn down. This principle applies not only in the case of a failure but also in the case when better routes are available than the existing ones. Fast restoration of LSPs can be achieved by setting up preconfigured backup paths using traffic engineering.

The motivation behind the Fast Reroute approach [4] is to reverse traffic at the point of the failure back to the ingress node of the protected LSP and redirect it via a parallel preconfigured LSP. This mechanism involves setting up two path segments. The *reverse segment* runs in the reverse direction of the working path, from the egress node to the ingress node, while the *alternate segment* runs from the ingress node to the egress node through nodes that are path and link disjoint with the working path. These two segments put together form the backup path for an LSP.

The major problem associated with these methods is the number of alternate paths that are to be established for every ingress and egress pair and the issue of finding link/node disjoint backup paths. As shown in Figure 1, it may not always be possible to find such a link disjoint path unless the working path is rerouted. Given that working path placement is a result of a potentially complex decision, adding further constraints is highly undesirable.

* This research was supported in part by NSF-ARI grant No. 9601602.

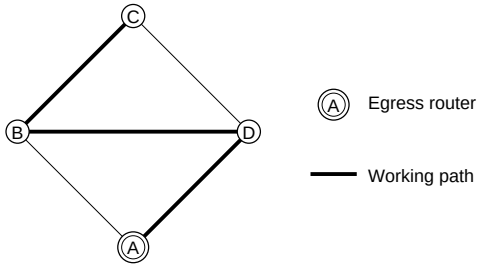

Figure 1: An example of a working path placement preventing a link disjoint protection path placement.

III. Two Path Protection Scheme

The proposed scheme addresses the issues of flexibility and cost as outlined above by establishing two protection paths between a node and an egress router. Since the two protection paths may share links with the working path, the the scheme provides greater flexibility than conventional single protection path methods at a potentially lower cost.

All paths protecting LSPs leading towards a common egress router are calculated simultaneously using the proposed heuristic algorithm. The paths placement generated by the algorithm utilizes LSP merging thus providing for reduction in required label table sizes in the routers. A full domain protection is achieved by a concurrent execution of the protection path placement algorithm in each egress router. MPLS traffic engineering methods are then used to establish the protection paths.

A. The protection path placement algorithm

Assume an MPLS domain whose topology represented by graph $G(N, L)$, where N is the set of n nodes and L is the set of l links between the nodes. Furthermore, assume that graph G is two-edge redundant and therefore can be protected against any single link failure. The algorithm attempts to locate two trees in graph G such that no single link failure would disconnect a node from the root of the tree (the egress node). A preliminary version of this algorithm have been presented by the authors in [6]. A more formal definition of the problem (termed multi-tree approach [7]) and alternative solutions can be found in [7, 8].

Input: The MPLS domain D and egress router e.

Output: Two collections of protection paths connecting ingress routers to egress router e.

Initialization: Find a spanning tree of graph G rooted in the egress router e. Let P be the set of nodes for which the protection paths have been established. Initially it contains the egress router: $P = \{e\}$.

Repeat until all nodes are protected ($P = N$):

1. Select one of the branches of the spanning tree attached to the egress node and mark all its nodes except for the egress node.
2. Scan all marked nodes to find node i that has a link to an unmarked node j.

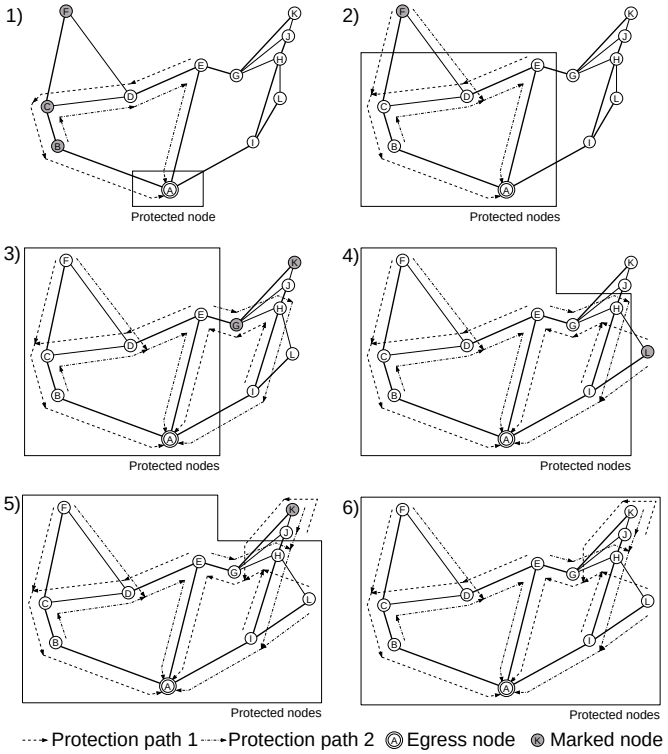

Figure 2: Protection path construction: vBNS topology.

3. Consider a ring path consisting of the links of the spanning tree leading from e to i, the link between i and j, and the links of the spanning tree between j and e (note that this segment of the ring is empty in the case $j = e$).
4. Place two protection paths along the ring: one in clockwise, the other in counterclockwise direction. The paths originate in the two nodes of the ring that are adjacent to the egress router and follow the ring all the way to the egress node. Merge the created protection paths with the protection paths established in the previous iterations of the algorithm for the protected nodes that are now a part of the egress node. All nodes on the ring are now connected to both protection paths and added to P.
5. In the subsequent iteration of the algorithm consider a new graph constructed by treating all nodes in P as a single node that will act as the egress node and by removing all links that connect two protected nodes.

Figure 2 shows the steps of the algorithm calculating protection path placement for a network with topology of vBNS backbone (with unprotectable leaf nodes removed) and node A as the egress router. All other nodes of the network act as ingress routers.

It should be noted that the algorithm makes arbitrary choices at several points. The rest of this section explores several heuristic approaches that can be employed to improve the quality of a solution without significantly increasing the algorithmic complexity.

B. Heuristic decisions

Finding optimal protection path placement is a difficult problem especially since it is expected that the scheme will be employed for domains with a large number of nodes. Furthermore, in many cases in real networks, it is even difficult to come up with a clear measure of quality since many security and business related issues have to be taken into account when considering protection. The scheme presented in this paper does not attempt to find an optimal solution, rather it provides for maximum degree of flexibility, allowing unforeseen criteria to be considered while designing a protection path placement

In this paper we consider two main quantitative measures of quality of a protection scheme: the length of the protection paths and the number of protection paths per link. The length of protection paths was chosen as an indication of the delay the traffic will experience after a link failure. The average of all protection path lengths for the entire domain approximates the overall impact of a fault on the traffic streams. The maximum protection path length gives the worst case scenario, an important measure for real-time traffic provisioning. In addition to the introduced delay, the length of protection paths reflects the amount of resources required to protect the domain.

The number of protection paths that pass through a link is used as an indication of the amount of resources, such as label table sizes and signalling overhead, that are required to setup and maintain the protection. The average and the maximum of the number of protection paths per link are also an indication of how well the protection paths are distributed throughout the network. The heuristics outlined bellow aim at improving the performance of the proposed protection scheme using these measures of quality.

An example in Figure 3 shows two possible placement of protection paths in a network. The first protection path placement is the result of identifying a ring along the nodes $ABCDE$. The longest protection paths ($EDCBA$ and $BCDEA$) are 4 hops long and the average protection path length is 2.5. The second protection path placement is the result of selecting rings along nodes ABE, then BCE, and finally BDE. In this case, the longest protection paths are only two hops long and the average protection path length is 1.75 hops.

The path placement algorithm described above makes arbitrary choices at three points: when a spanning tree is found during the initialization step, when a branch of the spanning is selected in step 1, and when a link connecting a marked node to an unmarked node is chosen in step 3. In all three cases, the algorithm can chose any of the available options without affecting the protection status of the nodes. Clearly, the choices made have an impact on the quality of the protection path placement.

The goal of the heuristic used for the spanning tree selection in the initialization step of the algorithm is to come up with a spanning tree with many short branches. Since the protection paths are routed mostly along the branches of the spanning tree,

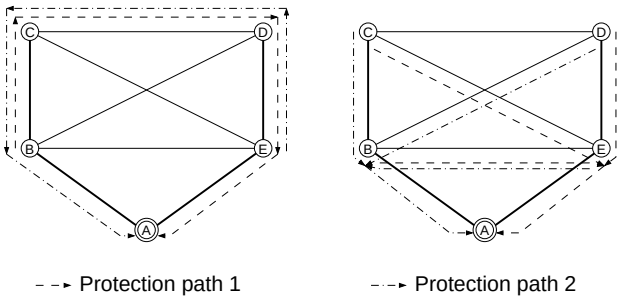

Figure 3: Two possible protection path placements.

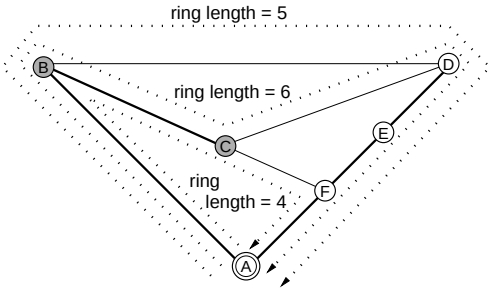

Figure 4: Heuristic edge selection.

the resulting protection path placement should exhibit lower maximum and average protection path lengths. Experiments described in the following section, utilized Dijkstra's algorithm to calculate the shortest path spanning tree.

Another point in the algorithm that can potentially affect the quality of the protection path placement is the choice of a spanning tree branch for marking in step 1. Branches with low depth (both average and maximum) appear to be suitable candidates as well as those with a smaller number of nodes.

The choice of a ring selected in step 2 of the algorithm also affects the quality of the solution. The method described bellow attempts to reduce the maximum and the average protection path lengths by finding the smallest possible ring when an edge connecting a marked node to an unmarked node is selected. Note that as a result of calculating the spanning tree during the initialization step, the distance from every node to the egress router is known. This information is used to calculate the length of the ring that is being formed (the length of the ring = the distance from a marked node to the egress + the length of the link between the marked and an unmarked node + the distance form the unmarked node to the egress node). In the process of scanning all neighbors of marked nodes the link with the smallest resulting ring length is selected and used in the subsequent steps of the algorithm.

Figure 4 shows an example of edge selection in step 2 of the algorithm. Assume that the branch of spanning tree with nodes B and C was selected in step 1 and the nodes were

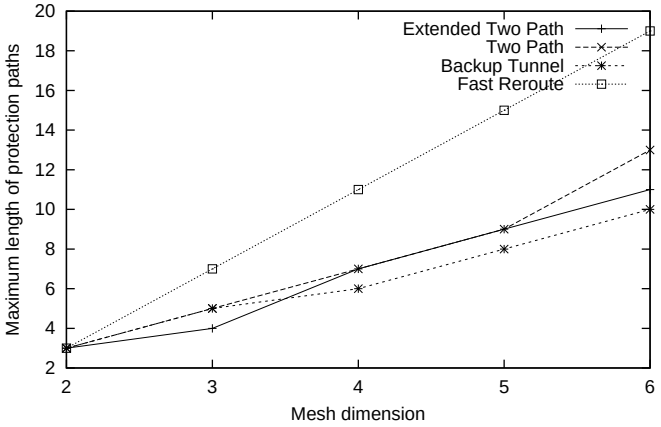

Figure 5: Maximum length of protection paths for meshes.

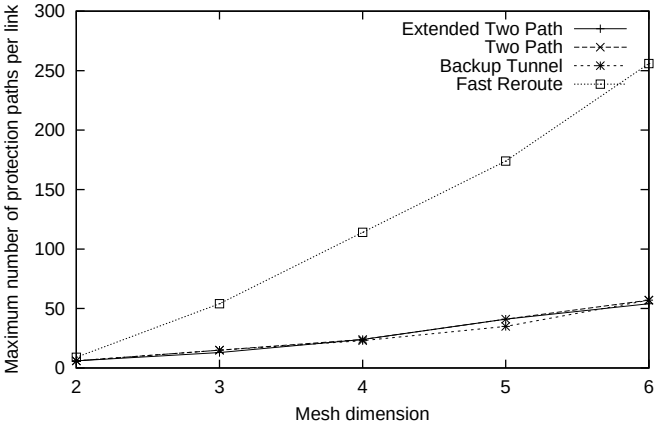

Figure 7: Max. number of protection paths per link for meshes.

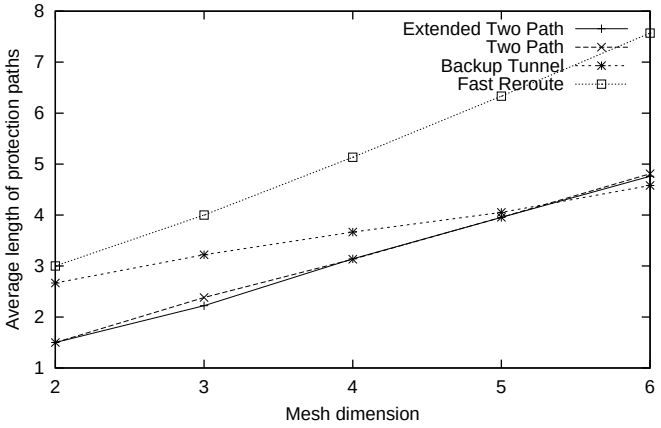

Figure 6: Average length of protection paths for meshes.

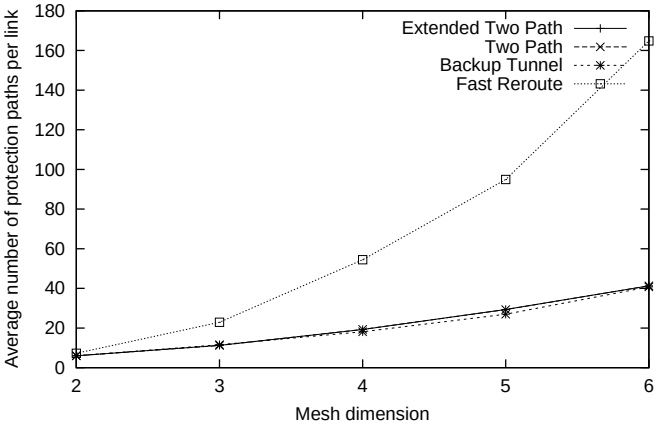

Figure 8: Avg. number of protection paths per link for meshes.

marked. There are three edges connecting marked nodes to the unmarked ones ((B, D), (C, D), and (C, F)). Corresponding ring lengths are 5, 6, and 4 respectively. Therefore, link (C, F) is selected and the ring is formed along nodes $ABCF$.

IV. Performance Evaluation

This section presents comparison of the proposed scheme with the two existing schemes for MPLS protection outlined in Section II. Two variants of the proposed scheme are considered in the experiments: one which uses makes arbitrary choice at all points of the algorithm except for the spanning tree selection where Dijkstra's algorithm is employed (Two Path). The second variant (Extended Two Path) utilizes the shortest ring selection heuristic described in the previous section. All four protection schemes have been implemented in a simulator and their performance was evaluated using the criteria described in the previous section: lengths of protection paths and the number of protection paths per node. For both measures average and maximum values are considered.

Mesh topologies were selected for one set of simulations since they resemble the backbone topologies and can be scaled. The results presented in Figures 5–8 for varying the mesh dimension show the trends as the number of nodes in a network increases.

Figures 9–12 show the performance for topology $R_{16}(k)$ [6] chosen to study the effects of varying connectivity of the network: $R_n(k)$, $k = 2, 4, ..., n-1$, is a network with n nodes, labeled $0, 1, ..., n-1$, where node i, $0 \leq i < n$, has links to nodes $i \ominus \lceil \frac{k}{2} \rceil$,..., $i \ominus 2$, $i \ominus 1$, $i \oplus 1$, $i \oplus 2$,..., $i \oplus \lceil \frac{k}{2} \rceil$ ($\oplus, \ominus$ represent addition/subtraction modulo n).

V. Conclusion

This paper has presented a novel approach to service restoration in MPLS network. The proposed scheme considers protection of all paths leading from ingress routers to a common egress router as opposed to traditional link or path protection. Protection using two paths allows for greater flexibility in protection path placement since the protection paths may share links with the working path.

Several heuristic methods that can be employed within the algorithm are proposed to improve the quality of the protection path placement without increasing the asymptotic complexity of the algorithm. Simulation results show that the proposed scheme provides protection that is better than recently

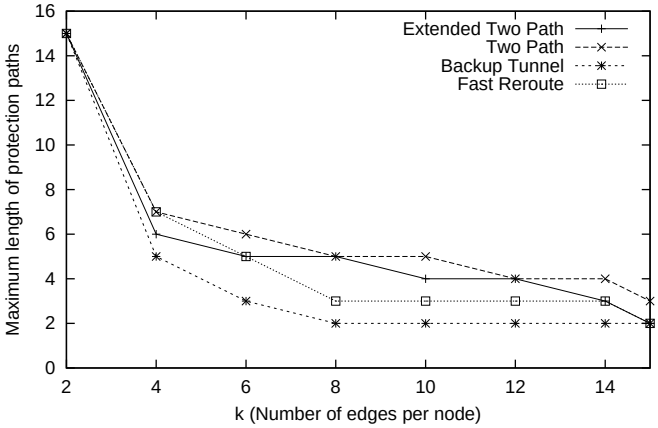

Figure 9: Max. length of protection paths for $R_{16}(k)$.

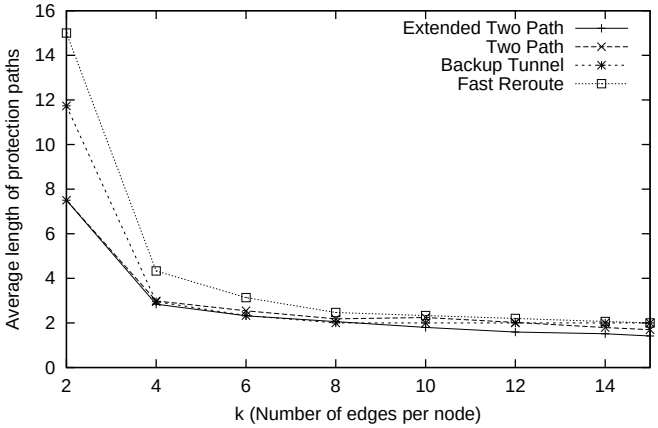

Figure 10: Avg. length of protection paths for $R_{16}(k)$.

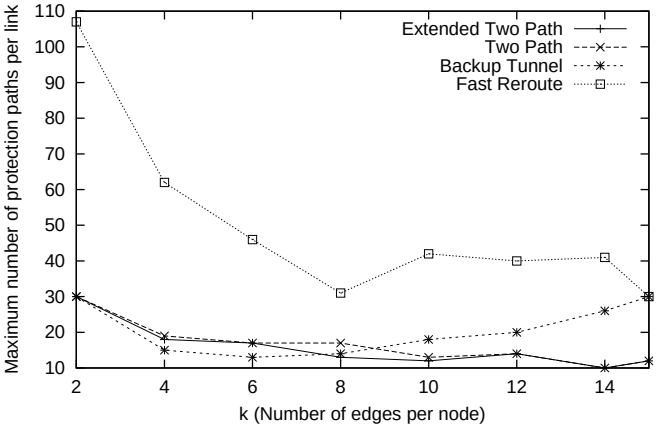

Figure 11: Max. number of prot. paths per link for $R_{16}(k)$.

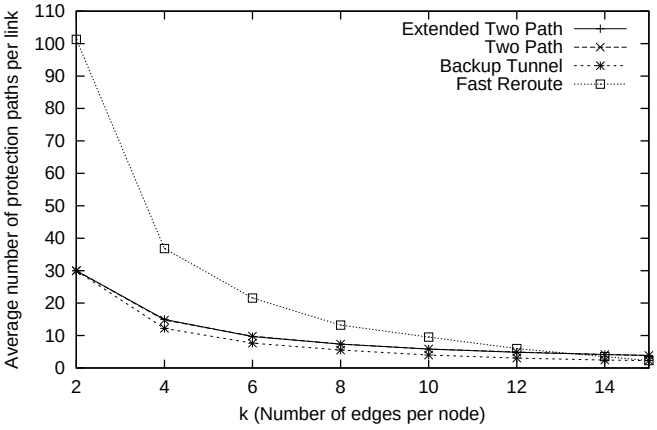

Figure 12: Avg. number of prot. paths per link for $R_{16}(k)$.

proposed Fast Reroute scheme. The algorithmic complexity of the proposed scheme is less than that of RSVP Backup Tunnels while providing comparably good protection. Unlike Fast Reroute and RSVP Backup Tunnels, the proposed scheme guarantees independence of the working and protection path placement.

ACKNOWLEDGMENT

The authors would like to express their thanks to Arun Gandhi for his help with implementation of the heuristic extensions in the simulator and conducting the simulation experiments.

REFERENCES

[1] E. C. Rosen, A. Viswanathan, and R. Callon, "MultiProtocol Label Switching Architecture." IETF RFC 3031, January 2001.

[2] V. Sharma et al., "Framework for MPLS-based recovery." Work in progress *[draft-ietf-mpls-recovery-frmwrk-01.txt]*, November 2000.

[3] D. O. Awduche, L. Berger, D. Gan, T. Li, G. Swallow, and V. Srinivasan, "RSVP-TE: Extensions to RSVP for LSP tunnels." Work in progress *[draft-ietf-mpls-rsvp-lsp-tunnel-07.txt]*, August 2000.

[4] D. Haskin and R. Krishnan, "A method for setting an alternative label switched paths to handle fast reroute." Work in progress *[draft-haskin-mpls-fast-reroute-05.txt]*, November 2000.

[5] S. Makam, V. Sharma, K. Owens, and C. Huang, "A path protection/restoration mechanism for MPLS networks." Work in progress *[draft-chang-mpls-path-protection-02.txt]*, November 2000.

[6] R. Bartoš and M. Raman, "A scheme for fast restoration in MPLS networks," in *Proc. of the Twelfth IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS), Las Vegas, NE* (M. Guizani and X. Shen, eds.), pp. 488–493, November 2000.

[7] A. Itai and M. Rodeh, "The multi-tree approach to reliability in distributed networks," *Information and Computation*, vol. 79, pp. 43–59, October 1988.

[8] M. Médard, S. G. Finn, R. A. Barry, and R. G. Gallager, "Redundant trees for preplanned recovery in arbitrary vertex-redundant or edge-redundant graphs," *IEEE/ACM Trans. on Networking*, vol. 7, pp. 641–652, October 1999.