Proc. of 5th International Workshop on Design of Reliable Communication Networks (DRCN2005), Naples, Italy, October 2005

Network Protection Design for MPLS Networks

Gaurav Agrawal*, Dijiang Huang†, Deep Medhi*
* Computer Science and Electrical Engineering Department
University of Missouri-Kansas City, MO 64110, USA
† Computer Science and Engineering Department
Arizona State University, Tempe, AZ 85287, USA

***Abstract*— In this paper, we consider the problem of designing partially and fully survivable explicit label switched paths for MPLS networks. We model the problems as Mixed Integer Linear Programs. Furthermore, we present a survivability assessment of the models. We also evaluate the performance of the models presented in the paper, in terms of total used capacity in the network.**

***Index Terms*— Routing, Traffic Engineering, Restoration, Survivability, Diversity.**

I. Introduction

In the past decade, there has been exponential growth in the traffic carried on the Internet. At the same time, most of the real-time applications, such as voice over IP, video conferencing and multimedia have migrated to the IP networks. These applications require high reliable service and quality of service (QoS) assurances. Therefore, network survivability has become an important consideration for network providers.

Multi-protocol Label Switching (MPLS) [1] has emerged as potential solution for addressing traffic engineering and providing survivability for IP networks. MPLS controls the packet flows through label switching. A label is assigned to a packet, when it enters an MPLS network at ingress Label Switched Router (LSR). At subsequent hops, this label is used as index into a table that specifies packet's next hop and a new label. The incoming label is swaped with this label and packet is forwarded to next LSR. The path traversed by the packet is called Label switched path (LSP) and Label distributed protocol (LDP) is used to set up the LSP. MPLS framework uses constraint-based routing to set-up explicit label-switched paths between a source and a destination. These explicit paths appear as point-to-point logical links at IP layer. An important feature of MPLS is its capability to set up multiple label switched paths between a source and destination. In this paper, we exploit this feature of MPLS to design explicit label switched paths such that traffic can be partially or fully survived in the event of a link failure.

Network survivability in MPLS Networks can be provided in many ways. One option could be to enforce path diversity [2] at the time of flow allocation. This way, we can force the demand volume to split in more than one path. Further, a restriction can be put on amount of flow on each path by introducing diversity constraints [3] to introduce some notion of survivability. In case of single link failure, only demand carried on one of the paths is lost, while rest of the demand is still carried in the network, thereby providing partial survivability against single link failures.

Another approach is to provide full network survivability by maintaining pair of link-disjoint paths (primary and alternate) for each service class between each source and destination. In the event of link failure, flow on the affected path can be quickly switched to the alternate path. This is known as Protection Switching or Fast Reroute [4]. Protection Switching can be divided into local and global repair [5]. In local repair, an alternate path is used as a bypass from point of protection to the next LSR, while global repair is used on an end-to-end basis. Protection Switching has the advantage of fast recovery in event of link failure, however additional resources are required to set up the alternate path. For example, in [6], the alternate path may remain unused until a link failure occurs (1:1 Path Protection) or carry the same traffic as carried by primary path (1+1 Path Protection). To avoid the bottleneck of maintaining the alternate path (even when all the links in the network are operational), the alternate path can be established after failure has occurred. However, in this case the network needs significant time to setup the alternate path and reroute the traffic. For a survey of restoration techniques for MPLS networks, see [7].

In the past, Wang and Wang [8] have presented the explicit routing models for MPLS traffic engineering, however they have not considered link failure in their models. Authors in [9] have used the path generation technique to design situation-disjoint pair of paths. [10] presents an integrated optimization formulation, where authors unify different objectives into single objective function and consider varied survivability requirements for different service class. Kodialam and Lakshman [11] have presented optimization models and algorithms for guaranteed tunnels with restoration. Yetginer and Karasan [12] have given integer linear programming based formulation to design working and recovery paths, while optimizing the MPLS network.

This paper considers one of the main issues for traffic engineering of MPLS networks: How to set up explicit label-switched paths between edges nodes of a MPLS network, such that a pre-defined objective is optimized. We first present a model, which does not take link failure into consideration. Therefore, in the event of a link failure, all the demand carried on the affected paths are lost. Further, we extend this model to incorporate single link failure scenario, such that label-switched paths for each source-destination pair are able to carry the full demand volume in the event of a single link failure. For the purpose of this paper, we do not consider multiple link failure at same time. We assume that only one link fails at one time, while other links are still operational.

In this paper, we consider only two disjoint paths to carry the demand for each service class between each source-destination pair. Instead of making one path as primary path and other back-up path, we use both the paths to carry the demand in the normal state (with no failure). This, in turn leads to lower value of maximum link utilization in the normal state.

We present a Mixed Integer Linear programming (MILP) formulation to set up pair of link-disjoint paths for each service class between source and destination. We incorporate a diversity constraint to restrict the flow on one path. Further, we extend the formulation to incorporate the single link failure scenario. We use a bi-criteria objective for the formulation, where we use an artificial constant to weigh between normal and failure states. Through computational studies, we do a survivability assessment of the network and a performance evaluation for the models presented in the paper.

The rest of the paper is organized a follows. In section II, we present the problem formulation. In section III, we present the results for experimental networks. Finally, we conclude in section IV.

II. Problem Formulation

The traffic engineering problem for robust path design in MPLS networks consists of finding an optimal pair of paths for each service class, such that in event of a single link failure, the alternate path can carry the flow carried by affected path, while optimizing a measure for network performance. We consider an aggregated-flow based network, where traffic arriving to a source for a specific destination, needs to be sent over two LSPs. For simplicity, we assume that traffic for a particular demand belongs to only one particular service class. We present a Mixed Integer Linear Programming Formulation for the problem. Notations for the formulation is described in Table I. We are given the information about N, D, $\mathcal{L}$, h_d, c_ℓ and θ.

TABLE I
Notations used in formulation

$\mathcal{N}$	:	Set of nodes in the network
$\mathcal{L}$	:	Set of links in the network
$\mathcal{D}$	:	Set of origin-destination demand pairs in the network
$\mathcal{P}_d$	:	Set of candidate link-disjoint paths for demand pairs $d \in \mathcal{D}$
S	:	Set of failure states
c_ℓ	:	Capacity of link $\ell \in \mathcal{L}$
h_d	:	Traffic demand volume for demand pair $d \in \mathcal{D}$
θ	:	Maximum allowed fraction of demand volume for each demand
r	:	The maximum link utilization in normal state
$\bar{r}$	:	The maximum link utilization over all failure state
$\bar{\omega}$	:	Weigh factor used to weigh between normal and failure states
$\alpha_{\ell s}$	:	1 if link $\ell \in \mathcal{L}$ is up in state $s \in S$, 0 otherwise
u_{dj}	:	Binary variable, 1 if demand d uses path $j \in \mathcal{P}_d$, 0 otherwise
x_{dj}	:	Volume of flow $d \in \mathcal{D}$ that is allocated on path $j \in \mathcal{P}_d$ in normal state
M	:	Very large positive number

Note that it is important to have enough link-disjoint paths in the set $\mathcal{P}_d$ such that pair of paths selected for each demand leads to optimal solution. Hence, fairly large number of paths need to be pre-computed. However, we can circumvent this computational overhead by using path-generation based techniques which can add relevant paths in an iterative fashion; for details, see [13]. We use k-shortest path algorithm to generate the set of link-disjoint paths in P_d. We use a link-path incidence matrix denoted by δ^{ℓ}_{dj}, which takes 1 if path j of demand d uses link ℓ, 0 otherwise.

A. Formulation without link failure

First, we present a model to design two link-disjoint paths for each source-destination pair without considering link failure in the formulation. We define the flow allocated to path $j \in \mathcal{P}_d$ of demand pair $d \in \mathcal{D}$ as x_{dj}, which can take values between 0 and $\theta \cdot h_d$, where h_d is the traffic demand volume for demand pair $d \in D$ and θ restricts the fraction the demand that can be allocated to a path and it is referred as diversity constraint. We define a binary variable u_{dj} such that $u_{dj} = 1$, if demand d uses path j and $u_{dj} = 0$ otherwise. To ensure the total demand volume is routed and no path carries more than allowed fraction of demand, we have the following constraints:

$$\sum_{j\in\mathcal{P}_d} x_{dj} \;=\; h_d, \quad d\in\mathcal{D} \tag{1}$$

$$\sum_{j\in\mathcal{P}_d} u_{dj} \;=\; 2, \quad d\in\mathcal{D} \tag{2}$$

$$x_{dj} \;\le\; \theta h_d u_{dj}, \quad d\in\mathcal{D} \;\; j\in\mathcal{P}_d \tag{3}$$

In this paper, we allocate each demand onto exactly two paths (restricted by (2)). However, this condition is not restricted and we can change the value of the summation of u_{dj} to fulfil different path-allocation requirements. Thus, the total bandwidth on any link to carry all the demands can be captured by $\sum_{d\in\mathcal{D}}\sum_{j\in\mathcal{P}_d}\delta^{\ell}_{dj}x_{dj}$. We define another variable r to denote maximum link utilization in normal state (no link failure). Since, we are given the capacity of each link, we have the following constraint for each link $\ell\in\mathcal{L}$:

$$\sum_{d\in\mathcal{D}}\sum_{j\in\mathcal{P}_d}\delta^{\ell}_{dj}x_{dj} \;\le\; c_\ell r, \quad \ell\in\mathcal{L} \tag{4}$$

Our goal is to minimize the maximum link utilization in the network. At this point, we have given conditions to design two link-disjoint paths for each demand under normal condition (with no link failure). We refer to this formulation as $\mathbf{RDP}$. Thus, $\mathbf{RDP}$ problem can be formulated as

$$\bar{F} = \min \;\; r \tag{5}$$

subject to constraints (1- 4)

B. Formulation incorporating single link failure

Next, we extend the model presented in the previous subsection to incorporate the single link failure. For the purpose of this paper, we consider a single link failure at one time. We illustrate the idea of failure states by adding another subscript, s, to account for failure states. Each failure state consists of the failure of one link at a time, while other links are fully functional. Therefore, we can define failure states $s = 0, 1...S$, where S is equal to number of links ($\mathcal{L}$).

We use the notation $\alpha_{\ell s}$ to denote 1 if link ℓ is up and 0 if it is down in state s. For each state s, we use x^s_{dj} to denote the additional flow on path j for demand d in state s. To ensure that in the event of a link failure, the flow on affected path is rerouted to the second path, we have following constraints:

$$\sum_{i\in\mathcal{P}_d, i\neq j} x^s_{di} \;=\; (1-\alpha_{\ell s})\delta^{\ell}_{dj}x_{dj} \tag{6}$$
$$s\in S \;\; \ell\in\mathcal{L} \;\; j\in\mathcal{P}_d \;\; d\in\mathcal{D}$$

$$x^s_{dj} \;\le\; u_{dj}M \;\; s\in S \;\; j\in\mathcal{P}_d \;\; d\in\mathcal{D} \tag{7}$$

We use a large number for M. In effect, constraint 7 ensures that in event of a link failure, flow on the affected path for a demand is switched to the alternate path for that particular demand. In the case of a link failure, only one path will carry the whole demand, if the failed link is on one of the two paths for that demand.

Thus, the total flow on a particular link in a particular state will be the sum of flow carried in normal state and additional flow in that failure state. However, we still need to ensure that total flow on a link does not exceed the link capacity in any failure state.

We incorporate another variable $\bar{r}$ in the formulation to denote the link overload variable over all failure states. Thus, we can define the capacity constraint as :

$$\sum_{d\in\mathcal{D}}\sum_{j\in\mathcal{P}_d}\delta^{\ell}_{dj}(x_{dj}+x^s_{dj}) \;\le\; c_\ell\bar{r}, \;\; s\in S \;\; \ell\in\mathcal{L} \tag{8}$$

For determining how much weight to give to the worst link utilization factor over all failure states, we incorporate a constant, $\bar{\omega}$, to weigh between normal and failure states. We refer to this problem as $\mathbf{FRDP}$. The problem $\mathbf{FRDP}$ can be defined as

$$\bar{F} = \min \;(1-\bar{\omega})r + \bar{\omega}\bar{r} \tag{9}$$

subject to constraints (1- 4) and constraints (6- 8)

C. Modified Formulation

In problem $\mathbf{FRDP}$, we need a constraint for rerouting flow on each link for every path and each demand in all failure states. This leads to large number of constraints and makes the problem very complex. Next, we present a simplified version of the $\mathbf{FRDP}$ formulation. Here, x^s_{dj} denotes the actual flow on path j of demand d in failure state s. To ensure that all demand is carried in all failure states, we have the following constraints:

$$\sum_{j\in\mathcal{P}_d} x^s_{dj} \;=\; h_d, \quad d\in\mathcal{D} \;\; s\in S \tag{10}$$

$$\sum_{s\in S} x^s_{dj} \;\le\; u_{dj}M, \quad d\in\mathcal{D} \;\; j\in\mathcal{P}_d \tag{11}$$

Constraint (10) ensures that each demand is carried in all failure states. Constraint (11) forces the flow x^s_{dj} to be zero if path j of demand d is not used to carry the demand d ($u_{dj} = 0$).

We have the following constraint to ensure that flow on each link in any failure state does not exceed the link capacity:

$$\sum_{d \in \mathcal{D}} \sum_{j \in \mathcal{P}_d} \delta_{dj}^{\ell} x_{dj}^{s} \;\; \leq \;\; \alpha_{\ell s} c_{\ell} \bar{r}, \;\; s \in S \;\; \ell \in \mathcal{L} \qquad (12)$$

We refer to this problem as $\mathbf{MFRDP}$. The problem $\mathbf{MFRDP}$ can be defined as

$$\bar{F} = \min \; (1 - \bar{\omega}) r + \bar{\omega} \bar{r} \qquad (13)$$

subject to constraints (1- 4) and constraints (10- 12)

III. Numerical Results

In this section, we present a survivability analysis for the models presented in section II. We also analyze the performance of the models in terms of the fraction of used capacity and link utilization. We have implemented the models by using CPLEX [14] callable libraries.

A. Experimental Networks

We have used four topologies as our experimental networks (EN), as shown in Fig. 1–4, which can be found in the literature [3] and [15]. EN I has 6 nodes and 12 links. EN II has 6 nodes and 10 links. EN III has 9 nodes and 14 links. EN IV has 10 nodes and 14 links. We have generated a reasonably large set of possible paths $\mathcal{P}_d$ for each demand d. We assume that there are 15 candidate paths for each demand pair for all experimental networks.

For the given experimental networks, we assume that there are traffic demand volume of 100 Mbps between all pairs of nodes. For each demand, we find the shortest paths between each pair of nodes based on hop-count and then assign all the demand volume on one of the shortest paths. Upon such allocation, we find the flow on each link and determine the capacity of link with maximum flow so that this link is at 40% utilization. We assign this capacity to all the links in the network.

B. Performance Measures

In our study, we consider the following performance measures to evaluate the formulations presented in section II:

- Minimum Carried Demand (MCD) captures the least fraction of demand volume carried for any demand in the network in any failure state. Ideally, the network provider would be interested in providing some level of survivability for each demand in the case of a link failure. MCD value of 1.0 indicates that all demands are fully carried in the network in any failure state.
- Total Carried Demand (TCD) captures the fraction of total demand carried in the network, averaged over all failure states.
- Fraction of Used Capacity (FU) captures the total used capacity in the final solution as a fraction of total capacity in the network.

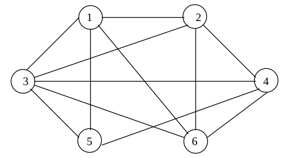

Fig. 1. EN I

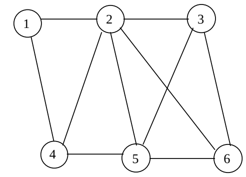

Fig. 2. EN II

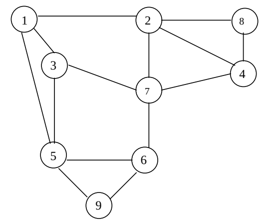

Fig. 3. EN III

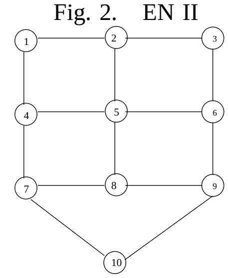

Fig. 4. EN IV

C. Survivability Assessment

We first present a survivability assessment for the solution of the problem $\mathbf{RDP}$, where we assume that all the links are operational. We evaluate the performance of $\mathbf{RDP}$ in terms of the performance measures MCD and TCD. We solve the problem $\mathbf{RDP}$ for $\theta = 0.55, 0.6....0.95$ and assess the solution in terms of demand carried in event of link failures. In Figure 5 and Figure 6, we present MCD value and TCD values respectively for increasing value of θ for EN I. We present results for only one of the experimental networks, as the values of MCD and TCD metric were found in close vicinity for all the experimental networks for a particular θ, and the values follow similar trend with increasing θ for all experimental networks.

We observe that the MCD value decreases with the increasing value of θ. The limit on the maximum fraction of each demand, that can be carried on each path increases with increasing θ, therefore, it is possible that one path carries most of demand volume for a particular demand while other path carries only small volume of

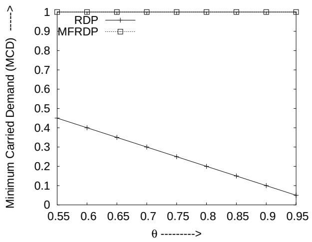

Fig. 5. MCD Values for EN I

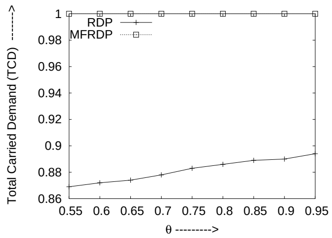

Fig. 6. TCD Value for EN I

that demand. For example, with $\theta = 0.95$, one of the path can carry up to 95% of demand and other path might carry only 5% of demand. If a link fails on the path carrying 95% of demand, only 5% of that demand will be carried in the network. Therefore, with increasing value of θ, the minimum carried demand for a demand in the network decreases, and hence the MCD value also decreases with the increasing value of θ.

MCD metric gives the survivability assessment for each demand, however TCD metric gives an account of the percentage of the total demand volume carried in the network. TCD value increases with the increasing value of θ. As explained above, it is possible that for a high value of θ, one of the path carry most part of demand. If the failed link is on one of the paths used to carry a demand, the actual demand volume carried for each network may vary drastically depending on the amount of flow on affected path. For example, for $\theta = 0.95$, it is possible that for one demand only 5% of the demand volume is carried while for most of the other demands, 95% of the demand volume is carried. On the other hand, for smaller values of θ, the demand volume carried for different demands will be in same vicinity. For example, for $\theta = 0.55$, for all the demands with affected paths due to link failure, the actual carried demand volume will be between 45% and 55%. This can lead to overall lower value of TCD for smaller values of θ while high value of TCD for larger values of θ.

We also show the values of MCD and TCD metrics for $\mathbf{MFRDP}$ formulation. Here, we assume that network has enough capacity to carry all the demand in case of a single link failure, therefore both MCD and TCD values are equal to 1.0 for all values of θ.

D. Performance Evaluation of $\mathbf{MFRDP}$

Next, we evaluate the performance of formulation $\mathbf{MFRDP}$. We present values of the FU metric for increasing values of θ in Figure 7 for experimental networks. We calculate the FU metric for $\mathbf{MFRDP}$ problem in the normal state (with no link failure) and in the failure states (averaged over all failure states). We also show FU values for $\mathbf{RDP}$ problem to compare the solution of $\mathbf{MFRDP}$ problem. For the purpose of this paper, we assign equal weight to maximum link utilization for normal and failure states ($\bar{\omega} = 0.5$) in the objective function defined in (13). We observed that flow allocation for $\mathbf{RDP}$ and $\mathbf{MFRDP}$ (in normal state) is almost same for EN I, however for EN II, EN III and EN IV, the flow allocation is different in both cases, leading to the difference in FU values.

The Figure 7 shows that the FU value decreases with increasing values of θ for all experimental networks. This is due to the fact that for larger values of θ, path with smaller number of hops carries higher volume of demands, while path with the larger number of hops carry small demand volume, leading to reduced FU value for high value of θ. When we take failure states into consideration, we get lower FU values compared to normal state. This can be attributed to the observation that in case of a link failure, demand on the affected paths is routed on the shorter(in terms of hops) alternate path for more number of demands, while only for few demands, demand on the affected path is routed on the alternate longer path (in terms of hops). This in turn leads to overall reduced value for FU metric in failure states.

In Figure 8, we show the link utilization for different links for experimental networks. We present the results for $\mathbf{RDP}$ and $\mathbf{MFRDP}$ in normal state and failure state (averaged over all failure states). For EN I, we get same values of link utilization for all links for $\mathbf{RDP}$ and $\mathbf{MFRDP}$ in normal state, while for EN II, EN III and EN IV, values differ for both problems. This can be attributed to the observation cited above regarding difference in flow allocation.

We also see an increase in the maximum link utilization for $\mathbf{MFRDP}$ in the failure state. When we take failure states into consideration, flow on the affected path due to a link failure is routed on the alternate path, thus increasing the utilization of links on the alternate path. This can be observed in Figure 8, as the maximum link utilization is more for $\mathbf{MFRDP}$ in the failure state compared to $\mathbf{MFRDP}$ in normal state and $\mathbf{RDP}$. Also, since only one path is used to carry the whole demand in the event of link failure, some links have very high utilization while some links are very lightly loaded, depending on whether links fall on the paths for demands with high demand volume or demands with low demand volume. On the other hand, in the normal state,

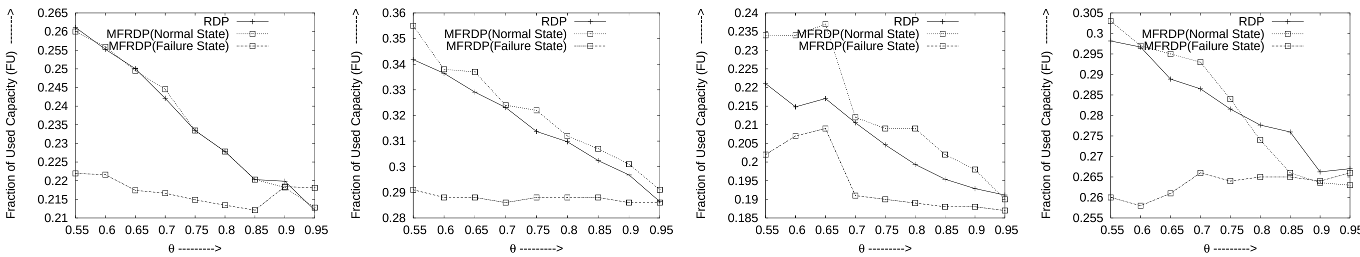

Fig. 7. FU Values for Experimental Networks

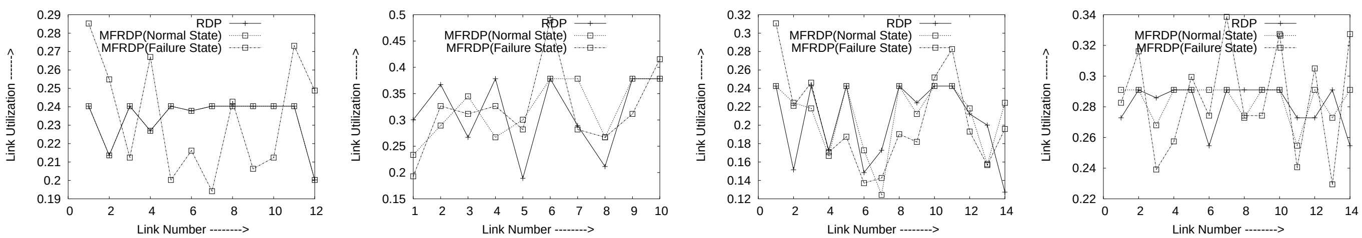

Fig. 8. Link Utilization for Experimental Networks

the demand gets distributed on the two paths leading utilization of the links in closer vicinity. Therefore, the degree of variation in link utilization among different links is quite high for failure state compared to normal state.

IV. Conclusion and Future Work

In this paper, we have presented a mixed integer problem formulation to set up two explicit label switched link-disjoint paths for each demand, such that all the demands in the network are fully carried in the network in event of a single link failure. We have presented a survivability assessment for the models presented in the paper for different value of diversity parameter. Through numerical studies, we show that the total flow in the network decreases with the increasing value of θ. We also show that in failure state, the total flow is less than the total flow in normal state.

As a part of ongoing research work, we are working on heuristic approaches to solve the problem for large networks. As the future work, we will explore how different objective functions can impact the survivable explicit label switched path design for MPLS networks.

References

[1] B. Davie, Y. Rekhter, *MPLS Technology and Applications*, Morgan Kaufmann Publishers, 2000.

[2] M. Pioro, D. Medhi, *Routing, Flow and Capacity Design in Communication and Computer Networks*, Morgan Kaufmann Publishers, 2004.

[3] S. Srivastava, B. Krithikaivasan, D. Medhi, *Traffic Engineering in the presence of Tunneling and Diversity Constraints: Formulation and Lagrangean Decomposition Approach*, International Teletraffic Congress, Berlin, Germany, 2003.

[4] D. Haskin, R. Krishnan, *A method for setting an alternative label switched path to handle fast reroute*, 2000, draft-haskin-mpls-fast-reroute-05.txt.

[5] L. Hundessa, J, D. Pascual, *Fast Rerouting mechanism for a protected Label Switched Path*, IEEE ICCCN, 2001.

[6] C. Huang, V. Sharma, K. Owens, S. Makam, *Building reliable mpls networks using a path protection mechanism*, IEEE Communication Magazine, vol. 40, no. 3, pp. 156-162, 2002.

[7] J. Foo, *A Survey of Service Restoration Techniques in MPLS Networks*, Australian Telecommunications, Networks and Applications Conference, 2003.

[8] Y. Wang, Z. Wang, *Explicit Routing Algorithms for Internet Traffic Engineering*, Proceedings of IEEE ICCCN, March 1999, New York, USA.

[9] B. Krithikaivasan, S. Srivastava, M. Pioro, D. Medhi, *Backup Path Restoration Design using Path Generation Technique*, DRCN, Alberta, Canada, 2003.

[10] S. Srivastava, S. R. Thirumalasetty, D. Medhi, *Network Traffic Engineering with varied levels of protection in the Next Generation Internet*, to appear in Performance Evaluations and Planning Methods for the Next Generation Internet, edited by A. Girard, B. Sanso, and F. Vazquez-Abad, Kluwer, 2005.

[11] M. Kodialam, T. V. Lakshman, *Dynamic Routing Bandwidth Guaranteed Tunnels with Restoration*, Proceedings of IEEE Infocom, 2000, Tel Aviv, Israel.

[12] E. Yetginer, E. Karasan, *Robust Path Design Algorithms for Tarffic Engineering with Restoration in MPLS Networks*, Proceedings of IEEE ISCC, 2002.

[13] G. Dahl, M. Stoer, *A Cutting Plane Algorithm for Multi-commodity Survivable Network Design Problems*, INFORMS Journal of Computing, vol. 10, pp. 1-11, 1998.

[14] *Manual of Cplex Callable Libraries*, ILOG Corporation, USA.

[15] R. Bartos, M. Raman, A. Gandhi, *New Approaches to Service Restorations in MPLS-based Networks*, IEEE Eurocon, 2001.