QoS Aware Path Protection Schemes for MPLS Networks

Ashish Gupta, Ashish Gupta, B.N. Jain
Department of Computer Science and Engg.
Indian Institute of Technology
New Delhi, India
{ag, ashish, bnj}@cse.iitd.ac.in

Satish Tripathi
College of Engineering
University of California at Riverside
Riverside, CA, USA
tripathi@engr.ucr.edu

Abstract

Uninterrupted transmission in case of failure is a major concern in MPLS based networks, and path protection is an effective method for this. Existing approaches towards path protection include local and global path protection. However, they provide fixed solutions and cannot be adapted to QoS and resource constraints for various kinds of traffic. In this paper, we introduce a new approach, "segment" based path protection, which provides efficient recovery from failure and guarantees QoS (like end-to-end delay, jitter) even after failure. We develop a resource-efficient algorithm which provides Segment-based protection for a given primary path, which guarantees an upper bound on the switch-over time. The protection configuration can also adapt to various QoS constraints, hence availing flexibility to the network administrator. We also present simulation results which show that Segment-based algorithms perform well in comparison to existing schemes in terms of resource reservation and assurance of QoS.

Keywords: MPLS, Routing, Path protection, QoS aware routing, Switch-over time

1.0 Introduction

Routing in the Internet today is focused primarily on connectivity, and typically supports only one class of service, viz. the best effort class. Multi-protocol label switching (MPLS) [1], on the other hand, allows flexiblity in routing traffic along different routes based on QoS and Traffic Engineering. In MPLS, packets are encapsulated at ingress points with labels that are then used to forward packets along label switched paths (LSPs). Label Switching Routers (LSRs) along the LSP perform label stack operations and forward the packets along a pre-determined path. This enables more sophisticated features such as quality-of-service (QoS) and traffic engineering to be implemented [2] [3]. An important component of providing QoS, however, is the ability to do so reliably and efficiently. Although the current routing algorithms are robust, the time they take to recover from a failure can be significant

(of the order of several seconds), thus causing serious disruption in service. This is unacceptable in certain applications, that require recovery times to be on the order of tens of milliseconds [4]. Any breakdown of a network component must not affect the traffic stream to a point that the service is impaired. Hence, a robust scheme for fault - tolerance must be present to deal efficiently with failures.

Method of path protection ([5], [6], [7], [8], [9], [10], [11] and [12]) is a way to provide fault tolerance in a network. It enables faster recovery from failures than is possible with Layer 3 mechanisms alone. In IP networks for example, in case of a failure, the source router on receiving failure notification identifies an alternate path to the destination using existing dynamic routing algorithms. It can then send packets over the new route. However the time consumed by this procedure is usually of the order of seconds, which may be unacceptable for network applications that involve transmission of multimedia or other real-time traffic. Thus, to minimize Switch over time the scheme of path protection was introduced. In path protection, an alternate backup path is pre-reserved at the time the primary LSP itself is established. The packets are initially sent on the primary LSP. In case of a failure on the primary LSP, the source LSR on recieving the failure notification re-routes the traffic over the backup path. This provides significant improvement over the Layer 3 mechanisms, in which major time is spent in finding an alternate path.

The Internet Draft [5] lays out a framework and requirements for providing protection services in MPLS networks. Two major schemes proposed for this, in the literature are a) global path protection and b) local path protection.

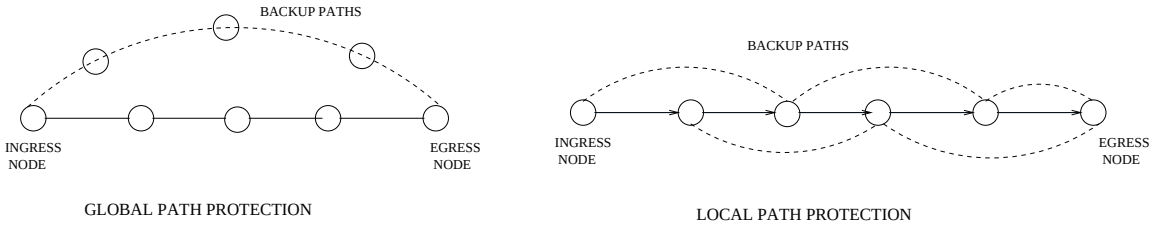

Figure 1. Existing Path Protection Schemes

In global path protection [6], there is only one backup path starting from the ingress LSR to the egress LSR. So in case of a failure, the ingress LSR is responsible for switching over the traffic to the alternate backup path.

In local path protection ([7], [10], [11] and [12]), the idea is to provide protection for each link or router separately. The difference in these two approaches is in the speed of recovery after failure and the amount of resources used by the backup paths.

These two schemes lie at opposite ends and provide fixed solutions to the problem of path protection. However, a whole range of solutions for path protection are possible between these two extremes. The goal is to provide efficient recovery and also meet QoS constraints even in case of failure, while also conserving bandwidth required for reservation of backup paths. There exists a tradeoff between these two requirements. In this paper, we look at a significantly general approach: Segment-based Protection. This method avails flexibility in providing path protection. It offers a two fold advantage: first of all it enables efficient recovery from failure while conserving backup resources (as compared to existing schemes). Secondly, using Segment-based Protection, we can also ensure satisfaction of various QoS constraints for the protected traffic streams, after a failure, like end-to-end delay, jitter and reliability. Though we use the context of MPLS networks, the scheme can easily be adapted to optical networks [4], or wherever path protection schemes can be implemented.

Our simulation results show significant improvements in protection design, where the satisfaction of certain QoS constraints can be assured while using lesser number of backup paths.

In the rest of the paper, we introduce Segment-based Protection in Section 2, mechanisms required for providing path protection in 2.1, design of segment based algorithms for QoS constraints like bounded Switch over time after failure (Section 3), issues related to formation of backup paths in 4, and experiment results related to the new approach in Section 5.

2.0 Segment-based Protection

The way flexibility is introduced, is to look at the primary LSP not just as a sequence of links but also as a sequence of segments, with each segments itself consisting of a sequence of links. The entire primary path can thus be broken down into variable sized segments (Figure 2). In Segment-based Protection, the idea is to provide protection for each segment as a whole, instead of providing protection for the whole path or each link separately.

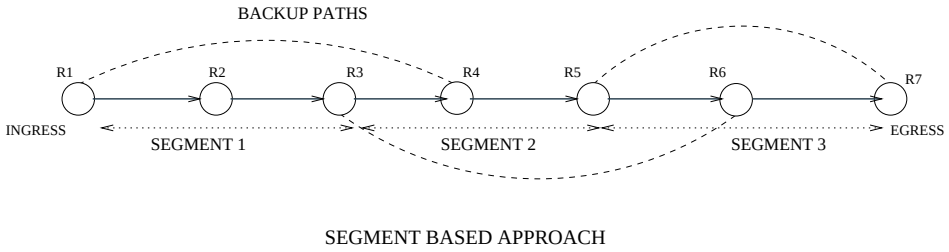

Figure 2. Segment-Based Protection

Each segment consists of a Segment Switching Router (SSR) and a set of protected routers (Figure 2). SSR is reponsible for switching over the traffic to the backup path in case of any failure among the protected routers or links connecting the routers in the segment. For example, in Segment 1, R_1 is the Segment Switching Router and it protects the routers R2 and R3 and the links connecting R_1, R_2 and R_2, R_3. We can denote this segment by $< R_1, R_2, R_3 >$. Similarly in Segment 2 ($< R_3, R_4, R_5 >$)

, LSR R_3 protects routers R_4 and R_5 and the interconnecting links. Note that the SSR cannot protect itself but is protected by the upstream SSR. In case R_5 fails (which is the SSR of Segment 3), its failure will be protected by the SSR R_3.

The number of segments and size of each segment can vary, thus providing flexibility in protection design. Segment-based Protection will result in fewer backup paths, and may still be able to meet the given recovery bounds and QoS constraints as required for a particular traffic stream, resulting in a better protection configuration. Note that, local path protection and global path protection are special cases of segment based path protection.

The main advantage of this approach is that it can be configured to not only provide efficient recovery after failure, but to also ensure satisfaction of QoS constraints (like end-to-end delay, jitter, reliability and bandwidth constraints) for the traffic streams, after a failure, and thus provide flexibility to the administrator to choose between the tradeoffs and select the most appropriate protection scheme.

Since lot of possibilites exist for providing protection using Segment-based Protection, the main question that arises is that how to convert the SLA parameters (the QoS and bandwidth constraints) into an efficient segmentation of the entire path. Seperate algorithms need to be developed based on Segement-based Protection, for the different QoS constraints. In the rest of the paper we investigate this problem and show how these algorithms can be developed. We present analysis and a new algorithm for providing segment-based protection configurations which satisfy the QoS constraints of bounded Switch over time and end-to-end delay with an aim of conservative use of backup path. We look at the mechanisms required for providing path protection. We also consider issues concerning creation of backup paths and the possibility of sharing the backup bandwidth among multiple LSPs.

Note that in the above protection scheme, protection cannot be provided in case the ingress or the egress fail. Besides these, any number of link or node failure within a single segment can be handled. Failures in more than one segment at once may not be protected, depending on the backup path configuration. For example, in Figure 2, if R_2 and R_4 fail, no backup path exists, while failure of R_2 and R_6 can be handled.

2.1 Protection Mechanisms in Segment Based Protection

To provide protection against faults, various mechanisms need to be in place like detection of fault, locating it, its notification to the router responsible for switching the traffic to the backup path, and then finally switching over the traffic to the backup path. [5] discusses these mechanisms and gives simple approaches to achieve the above-mentioned mechanisms. Here we discuss these mechanisms in context of the Segment Based Approach.

Figure 3 shows a scenario in which one of the routers on the LSP has failed. $< R_i, R_{i+1}, .., R_k, .., R_{j-1}, R_j >$ is a segment where R_i is the SSR and R_{i+1} to R_j are the protected routers. Router R_k has failed where $i + 1 \leq k \leq j$.

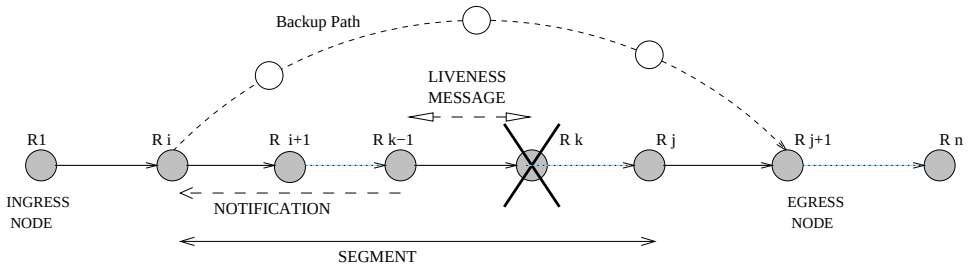

Figure 3. Mechanisms involved in Path Protection

2.1.1 Detection and Location of fault

For detection, we assume the simple mechanism of a liveness message [5]. It is a "Hello" message which is exchanged between two neighboring LSRs in the network. Every LSR sends the liveness message periodically to its neighboring LSRs and waits for an acknowledgement. In case of no response within certain duration of time, the LSR infers that the particular neighboring link or the LSR has failed.

Locating the fault is trivial in this case as the immediate upstream LSR detects the fault in this case. In the given example, when R_k fails , LSR R_{k-1} detects the failure.

2.1.2 Notification

If R_l is the LSR which detects or is notified of an error in the LSP and R_l is the segment switch LSR for the LSP , then it switches the traffic to the backup path. If R_l is not the Segment Switching Router then it sends a notification message to R_{l-1}.

Note that multiple LSPs may get affected by a single failure. In this case, the above mentioned mechanisms will be executed for each LSP.

2.2 Creating the Backup Paths

Creating the backup paths for each segment is an important step in providing protection. While creating the backup paths, we need to take care of certain conditions to provide reliable path protection, They are:

1. Each backup path should satisy the set of constraints that were specified for the primary LSP.
2. The backup path must be node-disjoint with all the nodes on the segment which it is protecting, otherwise packets after being transmitted over the backup path may again be dropped because of failure in the segment.
3. The backup paths must not create loops in the path for the data packets. For this the backup paths created for a segment must be node-disjoint from all the LSRs upstream of the Segment Switching Router.

3.0 QoS Aware Protection Algorithms

In this section of the paper, we discuss design of Segment-based algorithms, which provide an efficient protection configuration, given the QoS constraints while aiming to conserve the backup bandwidth used. QoS constraints which we consider in this paper are:

1. Bounded Switch Over Time
2. End-to-end delay

Switch over time refers to the time for which the packets will be dropped over the primary LSP after a failure.

End-to-End delay refers to the transmission delay from the ingress to the egress. It is an important QoS constraint especially in multimedia applications like voice communications. Path protection schemes must ensure that after a failure, the packets do not violate end-to-end constraints after being switched to a backup path.

3.1 An algorithm for bounded Switch over time

In this section, we focus on the first QoS constraint i.e. bounded Switch over time and present the design strategy for segment based algorithms for QoS constraints.

For this, we first present an analysis for Switch over time and derive a closed form expression for it in terms of network parameters. This is needed for deciding the size of segments on the primary path and for designing the algorithm. Then we discuss the design principles behind the new segment-based algorithm and present the problem statement.

3.2 Analysis for bounded Switch over time

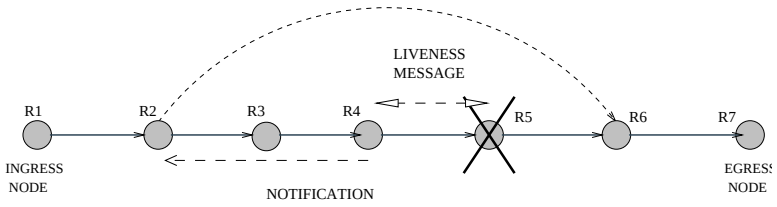

Figure 4. Notification Mechanism

Consider Figure 4 showing a situation in which LSR R_5 has failed (LSR R_5). R_2 is the Segment Switching Router of the segment to which R_5 belongs.

We assume the following parameters:

- T_{test} : Periodicity of the Liveness message which is exchanged between the neighboring LSRs (for example R_4 and R_5 here)

- $OWD(R_i, R_j)$: One Way Delay is the transmission delay for a data packet from LSR R_i to reach LSR R_j.
- $RTT(R_i, R_j)$: Round Trip Time for the LSR pair (R_i, R_j). Note that RTT$(R_i, R_j) = OWD(R_i, R_j) + OWD(R_j, R_i)$.

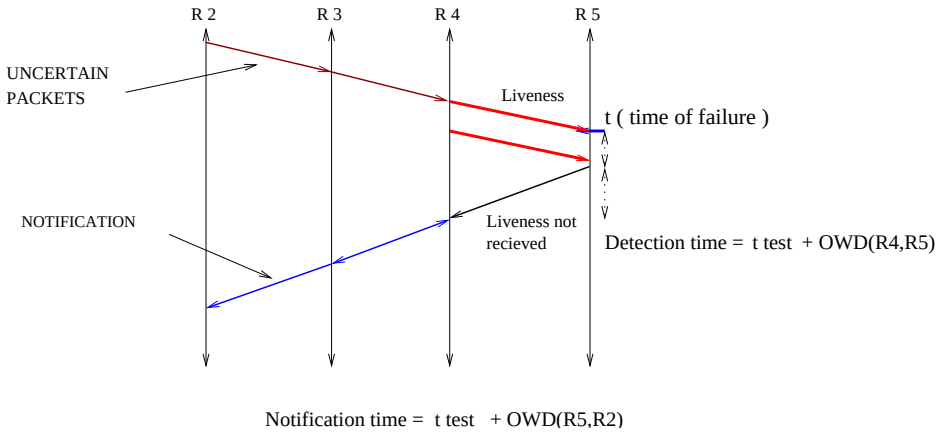

Figure 5. Analysis for Switch Over Time

Figure 5 shows the analysis in a timing diagram in which the worst case scenario is shown, where the fault occurs just after acknowledging the 'Hello' messsage. Now, also note that since the LSR fails at time t , the fate of the packets which enter this segment after $t - OWD(R_2, R_5)$ is uncertain. By the time, they will reach the LSR R_5, it would already have gone down.

The router R_4 will detect the error at $t + t_{test} + OWD(R_5, R_4)$.

It will take $OWD(R_4, R_2)$ time for the error notification to reach the SSR R_2.

Thus, it can be seen that the from time $t - OWD(R_2, R_5)$ to time $t + t_{test} + OWD(R_5, R_2)$, the packets will be dropped. Therefore, the total time for which packets will be dropped is $t_{test} + OWD(R_5, R_2) + OWD(R_2, R_5)$ which is equivalent to $t_{test} + RTT(R_2, R_5)$

For a general expression, for the segment $< R_i, R_{i+1}, ..., R_j >$, in the worst case (when R_j fails), the packets will be lost for time $t_{test} + RTT(R_i, R_j)$.

If the QoS constraints specify the bound on Switch over time to be δ, then for each segment $< R_i, R_{i+1}, ..., R_j >$, the following must be satisfied:

$$RTT(R_i, R_j) + t_{test} < \delta \tag{1}$$

With this expression, we now have a method to limit the size of the segment, while dividing the primary LSP into segments.

3.3 Problem Statement

We consider a given primary LSP $< R_1, ..., R_i, ..., R_n >$. We want to provide path protection such that:

1. It satisfies the bounded Switch over time constraint
2. Minimizes the number of backup paths used.

There are two aspects to this problem:

1. We need to divide the LSP into segments such that the bounded Switch over time constraint is satisfied. This is also equivalent to finding the origin of the backup paths.
2. Finding the backup path. This can be governed by various heuristics like conserving the use of bandwidth. The backup path itself may also affect other QoS constraints. All this needs to be taken into account while creating the backup paths. Note that in addition, the backup path must satisfy the constraints given earlier in Section 2.2.

Here we focus on the first aspect along with the constraints on the backup paths given in Section 2.2.

The Segment Switching Routers are denoted as S_i which refers to the ith such SSR. The ith Segment can then be denoted as $< S_i, S_{i+1} >$. We denote the backup path originating at S_i by $< P_{i_1}, ..., P_{i_m} >$ having m routers.

The problem statement can be defined formally as:

Given an LSP $< R_1, ..., R_i, ..., R_n >$ *, and an upper bound* δ*, identify* k *segments* $\{< S_i, S_{i+1} >, i = 1, ..., k\}$ *and backup paths* $\{< P_{i_1}, ..., P_{i_m} >, i = 1, ..., k\}$*, such that*

- $S_1 = R_1$
- $S_k = R_n$
- $RTT(< S_i, S_{i+1} >) + t_{test} <= \delta$
- *For each* $S_i, i = 1, ..., k$ *there exists a path* $< P_{i_1}, ..., P_{i_m} >$ *such that*
  - $\{R_1, .., R_n\} \cap \{P_{i_2}, .., P_{i_{m-1}}\} = \phi$.
  - $P_{i_1} = S_i$
  - $P_{i_m}\ \varepsilon\ \{R_{j+1}, ..., R_n\}$ *where* $R_j = S_{i+1}$
- k *is minimum.*

where RTT *is the round-trip time and* t_{test} *is the periodicity of liveness messages which is assumed to be same for all links.*

3.4 Algorithm Design

By earlier analysis, we showed that the primary LSP has to be divided into segments such that for each segment the condition: $RTT(R_i, R_j) + t_{test} < \delta$ is satisfied where δ is the maximum permissible Switch over time. We also want to minimize the number of segments. Note that larger the segment size, fewer the number of segments required.

A simple way to do this is to segment the LSP using a greedy approach. Starting from the ingress, identify the largest such segment which satisfies Eqn. 1. After finding each segment, repeat this process for rest of the LSP. This way it segments the entire path into minimum number of segments such that they satisfy Eqn. 1. However, this algorithm for segmentation does not ensure the presence of backup path for each segment. This is beacuse, for a given segmentation of the path, it may not be possible to find backup paths originating from the SSRs due to topology or bandwidth restrictions.

Therefore, we suggest another algorithm, which takes into account the existence of a backup path while forming each segment. This algorithm is a modification of the greedy approach in which the process of finding the backup path is combined with the process of segmenting the primary path. We follow an adaptive process of segmenting the primary LSP in which the segment size may not be the largest possible according to the constraints but may actually be shorter , to accommodate the formation of alternate paths for protecting that segment.

The algorithm is based on the following ideas:
To segment the entire path, we start from the egress LSR and find the largest possible segment towards the ingress LSR, which will satisfy Equation 1. Let the SSR of this segment correspond to S_i. Then we try to find a backup path from S_i, which will protect this segment. If we are unable to establish a backup path, we shorten the segment size by one link and try to find the backup path again. If it is not possible to shorten the segment further, then it implies that it is not possible to segment the LSP with the given QoS constraints. If we are sucessful in finding a backup path, we continue this process for segmenting the rest of the path using the same approach.

An Example

In Figure 6 we show a simple scenario with the above algorithm at work. In Figure 6.I. we are given an MPLS network with the RTT for each link as 10 milliseconds. An LSP has been reserved as shown. Now we want to protect this LSP such that the maximum permissible Switch over time is 40 milliseconds. We assume the value of t_{test} to be 5 milliseconds. We start with the egress router and try to indentify the largest possible segment satisfying Equation 1. For the possible segment $< R_3, R_4, R_5, R_6 >$, RTT + t_{test} comes out to be 35 milliseconds and for the segment $< R_2, R_3, R_4, R_5, R_6 >$, it is equal to 45 milliseconds which exceeds the bound. Therefore R_3 is proposed as the SSR as shown in Figure 6.II. However, in the given network, no backup path exists originating from R_3. Therefore, we shrink the segment and propose the possible segment as $< R_4, R_5, R_6 >$. Since a backup path is available from R_4, this segment is accepted. Figure 6.III. shows R_4 as the SSR with the backup path shown. Figure 6.IV. shows the final protection configuration.

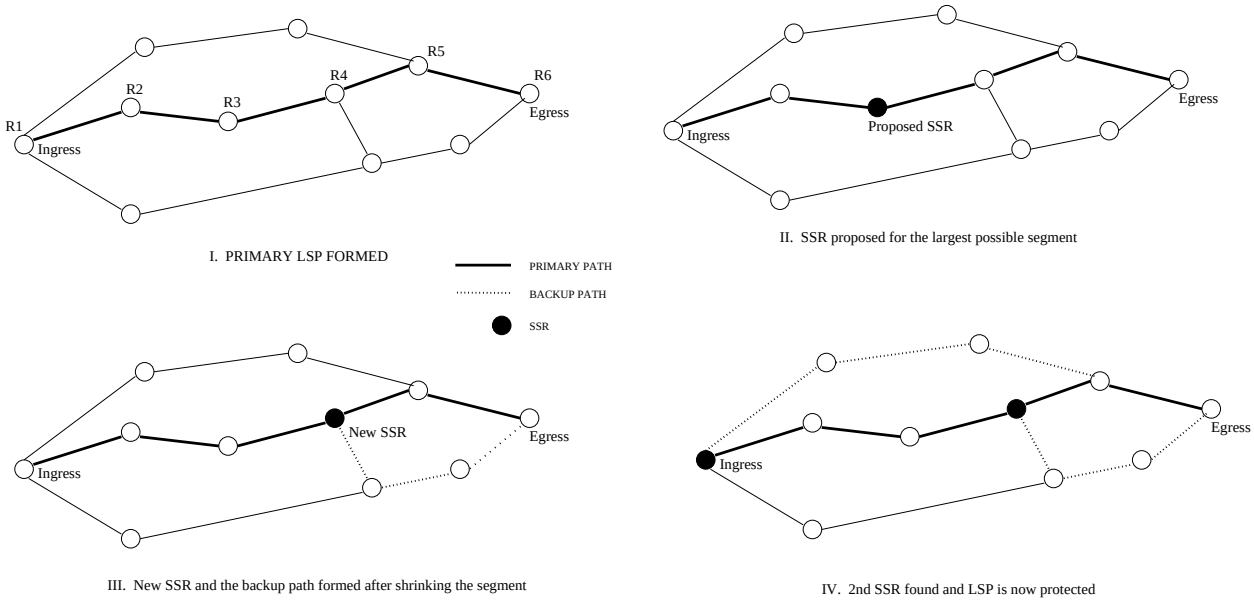

Figure 6. Example of the proposed algorithm

3.5 Algorithm for Segmenting the LSP

The following algorithm formally describes the approach described above:

$SegmentAlgo(< R_1..R_n >, \delta)$
{
$segment \leftarrow 1;$
$last \leftarrow n;$
$while(last > 1)do$
{
$S^*_{segment} \leftarrow R_{last};$
$segment \leftarrow segment + 1;$
$first \leftarrow identify\_next\_segment(n, last, \delta);$
$while(!find\_backup\_path(first, last, < P_{segment_1}, ..., P_{segment_m} >)and(first! = last))$
{
$first \leftarrow first + 1;$
}
$if(first = last)$
{
$print($"Unable to make backup paths"$);$
$exit(0);$
}
$last \leftarrow first;$
}

$S^*_{segment} \leftarrow R_1;$
}

$identify\_next\_segment(last, \delta)$
{
$first \leftarrow last;$
$while(first >= 1) and (t_{test} + RTT(R_{first-1}, R_{last}) <= \delta) do$
{
$first \leftarrow first - 1;$
}
$return \ \ first;$
}

Here $identify\_next\_segment$ identifies the largest possible segment satisfying Equation 1 with R_{last} as the last LSR of this segment. R_{first} correponds to the SSR of this newly identified segment. Note that in the above algorithm $S_{segment} = S^*_{k-segment+1}$ where k is the number of segments formed. This is necessary because we are forming the segments from the egress hence SSRs are assigned in the reverse order in the algorithm.

4.0 Finding the Backup Paths

Certain issues are involved in selecting backup paths for each segment. First is the conservation of backup bandwidth, which is possible by selecting backup paths with min hop count and sharing bandwidth among the backup paths as much as possible. Another issue concerns satisfaction of various QoS constraints after failure, as backup path formation may need to take into account certain QoS parameters. We consider the example of the QoS metric end-to-end delay and present an analysis which shows how backup paths may affect QoS.

4.1 Taking end-to-end delay into consideration

For the primary LSP, the end-to-end delay is the sum of the delays of each of the individual links. However, when we provide protection using alternate routes, we need to make sure that the new paths thus formed also meets the end-to-end delay constraints. While setting up a backup path for a segment, we need to make sure that in case that particular segment fails, the end-to-end delay characteristics of the new route from the ingress to the egress which includes this backup path, meets the specified constraints.

Specifically, let T be the end-to-end delay from the ingress to the egress over the primary path (Figure 7). If the end-to-end delay from the source of the backup path (corresponding to the i^{th} segment) to its landing place on the primary LSP is T_{i_2} and the end-to-end delay of the portion of the primary LSP which is protected by the backup path is T_{i_1}, then the difference $T_{i_2} - T_{i_1}$ is the additional end-to-end delay incurred by the packets in case of failure in the ith segment. If ω is the upper bound on

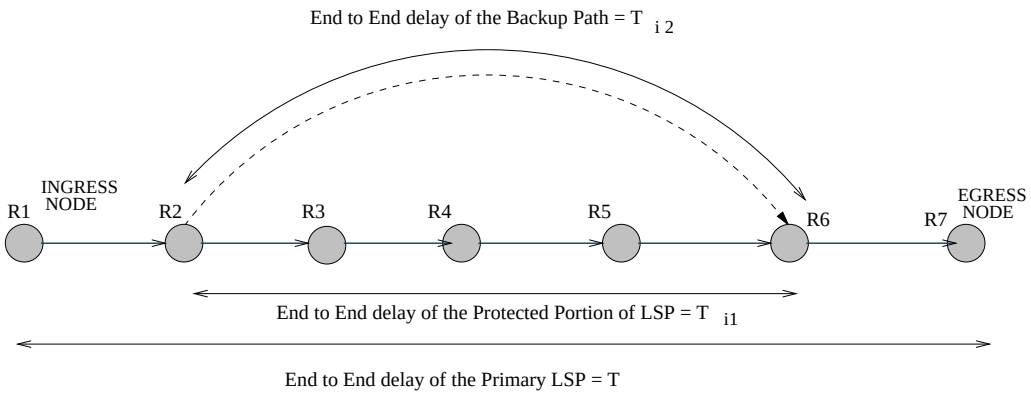

Figure 7. End-to-End delay

end-to-end delay, then

$$\forall_i \quad T + (T_{i_2} - T_{i_1}) < \omega \tag{2}$$

4.2 Function $find\_backup\_path$

Here we discuss the function $find\_backup\_path$ in the segment based algorithm described previously. First of all, constraints specified in section 2.2 must be met to avoid improper formation of backup paths. Without any additional constraints, the backup paths can be simply found by graph search methods such as DFS or BFS. However, as we have seen, many issues may govern their formation like conservation of protection resources by sharing of bandwidth and satisfaction of QoS constraints. Other schemes like computing maximally-disjoint multiple paths for improved fault tolerance has also been proposed in [13], which can be used for computing backup paths efficiently. Another paper [14] discusses the approach of minimum interference routing , where the aim is to find backup paths that do not interfere "too much" with possible future demands for primary LSPs. Another approach which is based on a new concept of Backup Load Distribution Matrix has been suggested in [15].

For sharing of backup paths [7] [10] [11], we need to maintain extra information for each link about the bandwidth used by the primary LSPs and the backup paths. A simple approach is that while creating the backup paths, we can find a path such that additional amount of bandwidth to be reserved is minimized. One approach for this is to use the shortest path algorithm with bandwidth of each link as the weight of the edge. Sharing of backup bandwidth assumes that the portions of primary LSPs which share bandwidth for their backup paths do not fail at the same time.

For taking end-to-end delay also into consideration, we can find the backup path which minimizes the transmission delay of packets from the SSR to the egress LSR. Here the weights of the edges will be the link delays. We need to make sure that it satisfies the end-to-end delay constraint specified earlier (Equation 2). This needs to be done for every backup path. Note that this constraint can be specified in addition to the bounded switch over time constraint.

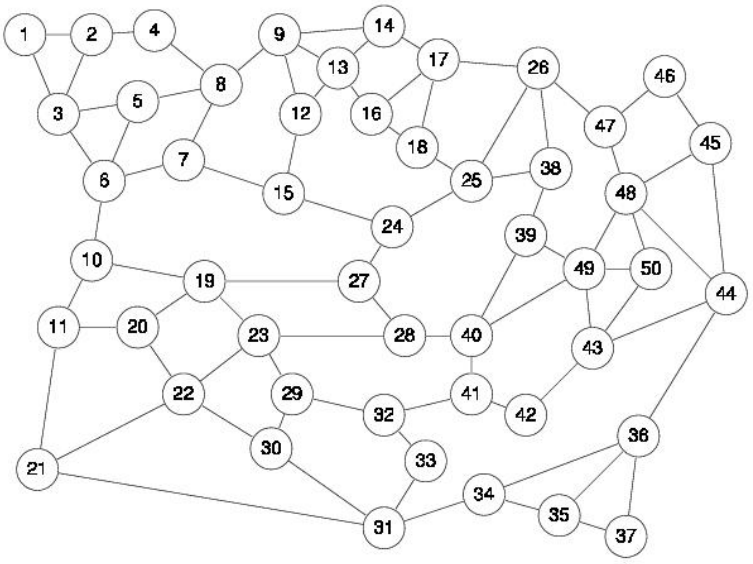

Figure 8. Topology used for simulations

5.0 Simulation Results

Before presenting the results, we describe the simulation model. We considered a topology with 50 routers and 82 edges (Figure 8).

The aim of these experiments is to evaluate the resource related advantages of the Segment-based protection. We generate random LSP requests to the network with specified bandwidth requirements (between 20 to 70 units, uniformly distributed) and bound on switch over time which varies with the experiment. An LSP request is generated by specifying an ingress LSR and the egress LSR, amount of bandwidth to be reserved along with the QoS constraints like bounded switch over time in case of failure. The primary LSP is setup using Djiktra's shortest path algorithm between the ingress and the egress with link delay as the cost of the edge. All the edges which don't satisfy the bandwidth requirement for the primary path are not included while searching for the shortest path.

Bandwidth for each link is assigned between 3000 and 10000 units (with uniform distribution). The delay of each link was set to 8 to 12 milliseconds.

For providing protection to the LSP requests, we implement the segment based algorithm for bounded switch over time. For creating the backup paths, we also consider sharing of protection resources as described earlier.

5.1 Protection Resource reservation with Local, Global and Segment Based Protection Schemes

In Figure 9 we see the effect on protection resources used by the various protection schemes (local, global and segment based path protection). For segment based Path protection, we consider 3 different

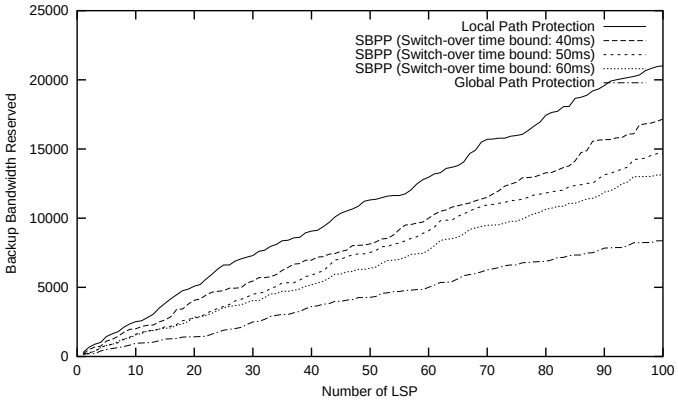

Figure 9. Comparison of schemes

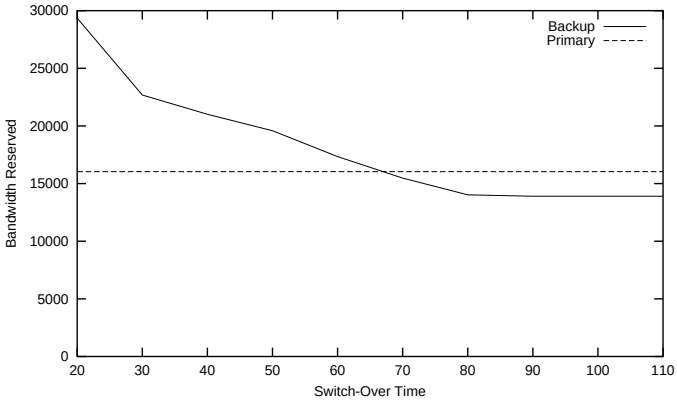

Figure 10. Effect of Switch over time bound on backup resources

variations with different bounds in switch over time. For local path protection, the average value of the bound on switch over time is 12 milliseconds and for global path protection the average value is 70 milliseconds. Segment-base Protection is effective in using the protection resources and varies with the switch over time bound. The results also indicate presence of a tradeoff between the protection resources and the QoS constraints. Using this approach, one can decide upon an effective strategy for path protection.

5.2 Resource reservation with variation of bound on Switch over time

In Figure 10 we compare the reservation of backup resources (for providing path protection) with the resources used by the primary LSPs. Segment-based Protection adjusts the protection configuration according the bound on switch over time. For small bounds on switch over time, protection resources used amount to almost twice the primary resources. As the bound on switch over time increases, we are able to accommodate larger segments and hence , amount of backup resources used fall below the primary path resources.

5.3 Rejection Rate vs Bound on Switch over time

Using Segment-based protection (Figure 11), we can admit more LSP requests than local path protection (here we generate 150 LSP requests). Some LSP requests have to be rejected due to insufficient bandwidth availability.

5.4 Comparison of protection resources reserved with the primary path resources

In Figure 12 we also see the effect of sharing on the amount of protection resources reserved. Here we generate increasing number of LSP requests and note the amount of backup resources and primary resources reserved. When the number of LSPs is less, the amount of backup resources used is more than the primary resources. As number of LSPs increase, the protection resources required for the later LSP requests are able to utilize the already reserved backup resources and falls below the amount of resources used up by the primary LSP.

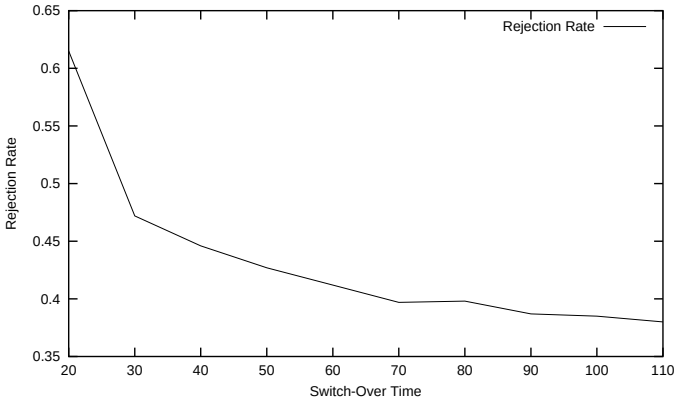

Figure 11. Rejection variation with bound on Switch over time

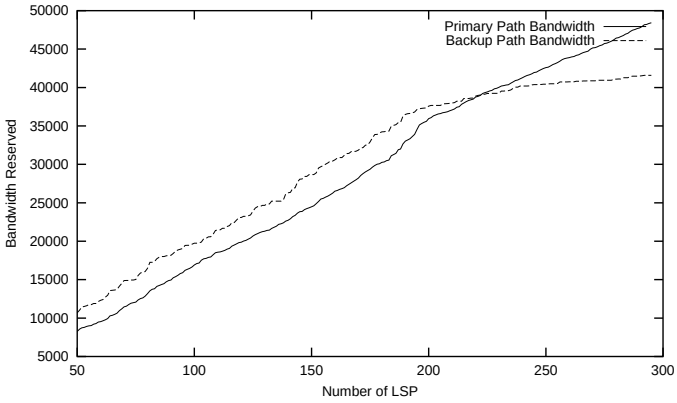

Figure 12. Backup resources vs Primary resources

6.0 Conclusions and Future Work

In this paper, we investigated a new approach for path protection, Segment-based protection, which views the primary LSP as a sequence of segments and provides protection for each segment seperately. The choice of choosing the number and size of these segments avails flexibility. The challenge is to come up with a path segmentation scheme which assures efficient recovery time as well as satisfaction of certain QoS constraints, even in case of a failure, while also using the protection resources efficiently. We developed a resource-efficient algorithm for segmenting the path, which provides a protection configuration for ensuring bounds on the switch over time in case of failure. We also investigated the issue of creating the backup paths and how different constraints like resource optimization and other QoS constraints (for example end-to-end delay) can affect their creation. The segment-based bounded switch over time algorithm proves to be efficient in comparison to existing schemes, as shown by the experimental results.

Using the design strategy presented in this paper, algorithms can also be developed for other important QoS constraints such as jitter and network reliability. Multi QoS constraint satisfaction algorithms can also be developed using this approach.

Acknowledgment

We are thankful to Anant Chaudhary and Abhishek Singh, Department of Computer Science and Engineering, IIT Delhi for their implementation of protection algorithms on a real network.

References

[1] E. Rosen, A. Viswanathan, and R. Callon, *"Multiprotocol label switching architecture,"* IETF RFC 3031, January 2001.

[2] S. Vutukury and J. Garcia-Luna-Aceves, *"A simple MPLS-based flow aggregation scheme for providing scalable quality of service,"* Proceedings of SPIE, Quality of Service over Next-Generation Data Network, vol. 4524, pp. 91–98, Aug. 2001.

[3] S. Dragos, R. Dragos, and M. Collie, *"Bandwidth management in MPLS networks,"* in IEI/IEE Symposium on Telecommunications Systems Research, November 2001.

[4] R. Doverspike and J. Yates, *"Challenges for MPLS in optical network restoration,"* IEEE Communications Magazine, vol. 39, no. 2, pp. 89-96, Feb. 2001.

[5] V. Sharma, B.M. Crane, S. Makam, K. Owens, C. Huang, F. Hellstrand, J. Weil, L. Andersson, B. Jamoussi, B. Cain, S. Civanlar, and A. Chiu, *"Framework for MPLS-based recovery,"* Internet Draft, July 2000.

[6] D. Haskin and R. Krishnan, *"A method for setting an alternative label switched paths to handle fast reroute,"* Internet Draft, Nov. 2000.

[7] Murali S. Kodialam and T. V. Lakshman, *"Dynamic routing of locally restorable bandwidth guaranteed tunnels using aggregated link usage information,"* in Proceedings of INFOCOM, 2001, pp. 376–385.

[8] S. Makam, V. Sharma, K. Owens, and C. Huang, *"Protection and restoration of MPLS networks,"* Internet Draft, Oct. 1999.

[9] J.P. Lang, J. Drake, Y. Rekhter, and A. Farrel, *"Generalized MPLS recovery mechanisms,"* Internet Draft, July 2001.

[10] S. Kini, M. Kodialam, T.V. Lakshman, S. Sengupta, and C. Villamizar, *"Shared backup label switched path restoration,"* Internet Draft, May 2001.

[11] A. Iwata, N. Fujita, and T. Nishida, *"MPLS signaling extensions for shared fast rerouting,"* Internet Draft, July 2001.

[12] Der-Hwa Gan, P. Pan, A. Ayyangar, and K. Kompella, *"A method for MPLS LSP fast-reroute using RSVP detours,"* Internet Draft, Apr. 2001.

[13] Scott Seongwook Lee and Mario Gerla, *"Fault tolerance and load balancing in QoS provisioning with multiple MPLS paths,"* Lecture Notes in Computer Science, vol. 2092, pp. 155–68, 2001.

[14] Murali S. Kodialam and T. V. Lakshman, *"Minimum interference routing with applications to MPLS traffic engineering,"* in Proceedings of INFOCOM, 2000, pp. 884–893.

[15] Samphel Norden, Milind M. Buddhikot, Marcel Waldvogel, and Subhash Suri, *"Routing bandwidth guaranteed paths with restoration in label switched networks,"* in Proceedings of the 9th International Conference on Network Protocols (ICNP 2001), Nov. 2001, pp. 71–79.