Dynamic Issues in MPLS Service Restoration*

Radim Bartoš and Arun Gandhi
Department of Computer Science
University of New Hampshire, Durham, NH 03824, USA
E-mail: {rbartos,agandhi}@cs.unh.edu

ABSTRACT
The rapid growth of real-time and high-priority traffic over IP networks makes network survivability more critical. Several MPLS-based recovery mechanisms have been proposed to ensure continuity of service following network impairments. These approaches, however, suffer from the transient effects that negatively impact the traffic that is being rerouted onto the protection paths. *Packet loss* and *reordering* are the most significant negative effects resulting from protection switching. In conventional MPLS networks, the detection and retransmission of out-of-order or lost packets is left to the higher layers, which in effect, degrades the overall performance. This paper proposes a signaling mechanism to minimize the impact of protection switching on packet loss and reordering. The proposed signaling protocol is general and independent of the particular MPLS protection mechanism. The signaling protocol uses the existing queues on the nodes to buffer incoming traffic and hence reduce loss of data and packet reordering during recovery operations. Easy to implement calculations are used by the nodes to estimate the required queue sizes and to control the signaling procedure. The protocol has been implemented and studied in the three MPLS protection mechanisms under consideration. The results of our simulation study show that, with the implementation of the proposed signaling scheme in the basic protection mechanism, the number of packets reordered is significantly reduced while maintaining or improving packet loss without imposing much overhead on the nodes in the network.

KEY WORDS
IP Networks, Network Protocols, Service Restoration, MPLS.

1 Introduction

Although IP networks offer flexibility and scalability, they need to be enhanced in the areas of availability, dependability, and *Quality of Service* (QoS) in order to provide a mission-critical networking environment. *MultiProtocol Label Switching* (MPLS) [1] is a new forwarding paradigm that integrates IP and link layer technologies thus allowing sophisticated routing control capabilities to be introduced into IP networks. The MPLS architecture combines scalability and feasibility of routing with performance, QoS, and traffic management of link layer switching.

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

An important component of providing quality service, however, is the ability to transport data reliably and efficiently. Failures in the network and interruptions of traffic are unacceptable to many applications that require highly-reliable service, that is, the recovery times to the order of tens of milliseconds to seconds [2], such as voice, streaming video, etc. Presently, IP traffic is routed and forwarded over the Internet using standard IP routing protocols which reroute traffic across the failed paths or links. Although the current routing algorithms are very robust and survivable, the amount of time they take to recover can be significant, on the order of several seconds or minutes, causing serious disruption of service to critical communications. Also, there are inherent limitations to improving the recovery times of current routing algorithms. SONET's *Automatic Protection Switching* (APS) provides fast restoration times (to the order of 50 ms) at the expense of inefficient use of bandwidth and is typically limited to ring-based topologies. Moreover, with the emergence of high-speed networking over Ethernet (Gigabit and 10 Gigabit) or other link layer technologies, SONET's failure detection no longer suffices.

2 Background

MPLS networks are more vulnerable to failures because of their connection-oriented nature. Fault recovery is usually first attempted in the physical layer and if unsuccessful (or not possible) is escalated to the higher layer. Fault recovery in MPLS is necessary to eliminate dependency on the physical layer recovery mechanisms which may differ between the networks [2]. MPLS-based recovery can give the flexibility to select the recovery mechanisms, choose the granularity at which the traffic is protected, and also to choose the specific types of traffic that are protected. It is possible to provide different levels of protection for different classes of service based on their service requirements.

In addition to a general signaling protocol [3], three mechanisms, *Fast Reroute* [4], *RSVP-based Backup Tunnels* [5] and *Two Path Protection* [6] have been proposed for the restoration of the LSPs in the MPLS domain. The Fast Reroute[1] scheme reverses the traffic at the point of failure of the protected LSP such that the traffic flow can then be redirected via a parallel LSP between source and des-

[1]The term *Fast Reroute* has been also used for various other protection and restoration mechanisms (e.g., [3]). In this paper it refers to the method proposed by Haskin and Krishnan [4].

tination switches of the protected LSP tunnel [4] and thus provides fast restoration times at the expense of comparatively less efficient use of resources. It also minimizes the path computation complexity but the obvious drawback is the total length of protection paths. Moreover, the straightforward implementation of this method would result in temporary packet reordering at PML as packets arriving from the reverse direction are mixed with incoming packets. RSVP-based Backup Tunnels mechanism extends the RSVP protocol to support the establishment of LSP tunnels. In this scheme, a node adjacent to a failed link signals the failure to the upstream nodes. An ingress node upon reception of the failure signal reroutes the traffic over either a pre-established or dynamically established path that is link disjoint with the working path. Two Path Protection scheme is based on protecting the entire domain and aims to reduce the number of protection paths required to make the MPLS domain fully protected. The protection paths are setup using traffic engineering and their topology calculated using the two path algorithm. Once the paths are established apart from the working path, every node will have two alternate paths to reach the egress node.

3 Dynamic Issues in MPLS Restoration

All three schemes presented above suffer from transient effects that negatively impact the traffic that is being switched onto the protection paths. Negligible impact on the switched traffic is very difficult and costly to achieve. For many applications, a certain degree of QoS degradation is acceptable, provided it is balanced by a significantly reduced cost of providing the protection. We have identified *packet loss* and *reordering* as the most significant negative effects resulting from protection switching. Recently published work [7] independently proposed to use the measures of loss and reordering as the major evaluation criteria of an MPLS protection scheme. Extended periods of packet loss could be unacceptable for some unreliable-transport-protocol-based streaming applications. Performance of reliable transport protocols, such as TCP, is also negatively affected by packet loss and reordering which are often used as the indicators of the network condition and trigger various congestion control mechanisms resulting in reduced application throughput.

3.1 Causes of Transient Effects

Consider a general protection switching scenario in the MPLS domain shown in Figure 1. The *working path* passes through nodes *IUDE*. When the link between nodes *U* and *D* fails, depending on the particular service restoration method, several pre-established alternate paths can be utilized: The ingress node *I*, acting as a PSL[2], is responsible

[2] *Protection Switching LSR* (PSL) is responsible for switching traffic between the working path and the protection path. *Protection Merging LSR* (PML), on the other hand, is an LSR that receives both the working path traffic and its corresponding protection path traffic. The PML either merges their traffic into a single outgoing path or sends it to higher layer protocols if it terminates the LSP [2].

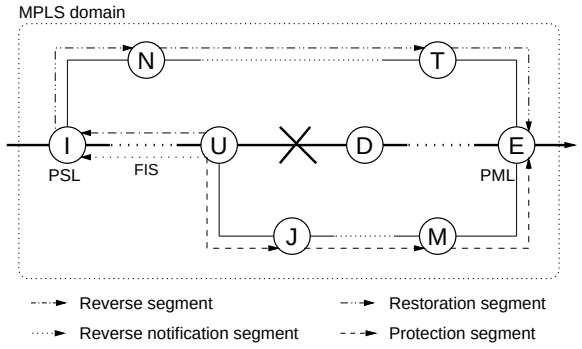

Figure 1. Service restoration in an MPLS domain.

for switching traffic over *restoration segment* (*INTE*). The node *U* upstream to the failed link has an option to send the traffic back to the ingress node over *reverse segment* (*UI*), send a *Failure Indication Signal* (FIS) [2] to the ingress node over *reverse notification segment* (*UI*), or to send the traffic over *protection segment* (*UJME*). Egress node *E*, acting as a PML, merges traffic from restoration and protection segments, and the working path.

Packet loss and reordering may occur at PML when the tail of a traffic stream arriving over one path temporarily overlaps with the head of rerouted traffic arriving over another path. Since the exact propagation delay of a path is typically difficult to predict, such conflicts cannot be resolved during protection path setup. Furthermore, the cases where the traffic is switched from a path with longer delay to a path with shorter delay are not unusual in MPLS protection schemes (e.g., traffic is switched over to a better but more costly path, use of which is justified only by the short duration of service restoration actions).

A PML has several straightforward options when dealing with overlapped streams of traffic. Packets from one of the streams can be dropped causing a significant packet loss, the two streams can be merged together causing packet reordering, or packets from the head of the stream can be buffered at the PML until all packets from the tail of the other stream are sent on. The last approach does not suffer from reordering; however, the buffers in the PML may not be able to store all the packets, forcing it to drop the excess ones. This paper proposes a signaling mechanism to reduce these transient effects caused due to the protection switching by distributed temporal realignment of the traffic streams. The proposed scheme makes heuristic decisions when controlling the distributed buffering of the traffic.

3.2 Protection Scenarios

Assuming the network in Figure 1 uses *Fast Reroute* [4] (Scenario 1) and the link between nodes U and D fails, then node U redirects the incoming stream toward node I over the reverse segment. The redirected stream and the input stream may then be transmitted simultaneously on the

restoration segment. The Fast Reroute mechanism does not cause packet loss beyond unavoidable loss of packets that were in transmission over the link at the time of its failure. However, substantial packet reordering occurs when the incoming traffic merges with the traffic on the reverse segment. Further packet reordering may occur at node *E* (the PML) when the delay on the segment of the working path between nodes *D* and *E* is longer than the delay of the reverse segment together with the delay of the restoration segment, $d_{DE} > d_{UI} + d_{INTE}$.

In the case when the network utilizes *RSVP-based Backup Tunnels* [5] (Scenario 2), node U sends a FIS signal to the PSL (node *I*). When node I receives the FIS, it stops forwarding traffic on the working path and switches incoming traffic onto the restoration segment (*INTE*). The traffic in transit on the working path segment *IU* is dropped. In addition to packet loss, this protection scenario may cause packet reordering at the PML if the delay of the segment of the working path between nodes *D* and *E* is longer than the delay of the reverse notification segment[3] together with the delay of the restoration segment, $d_{DE} > d'_{UI} + d_{INTE}$.

Under the *Two Path protection* [6] (Scenario 3), node U makes an immediate decision to reroute the current traffic stream onto the protection segment passing through the nodes *UJME* (in this scenario every node on the working LSP has a pre-established restoration path). Typically, only limited resources are allocated to the protection segments that is utilized only for a limited time before the ingress node *I* switches traffic onto the restoration segment which is assumed to have all the required resources pre-allocated. Two path protection does not cause packet loss beyond unavoidable loss of packets in transit over the failed link. Packet reordering may occur at the PML, which merges three traffic streams, under two conditions: First, when the delay of the segment of the working path between nodes *D* and *E* is longer than the delay of the protection segment, $d_{DE} > d_{UJME}$. Second, when the delay of the reverse notification segment together with the delay of the restoration segment is shorter than either the delay of the segment of the working path between nodes *D* and *E* or the delay of the protection segment, $d'_{UI} + d_{INTE} < d_{DE}$ or $d'_{UI} + d_{INTE} < d_{UJME}$.

4 Proposed Signaling Mechanism

To control the flow of the incoming traffic streams toward the PML and to address the issue of finite buffer space in the routers, a *Backpressure Signaling* mechanism is proposed. The goal of the signaling is to control the volume of traffic that reaches PML by deferring the transmission of packets between routers for as long as possible without overflowing the buffers in the routers. Three new signaling messages are introduced:

NR(b) - Not Ready, can receive up to b bytes,

SE - End of traffic stream,

RD - Ready to receive traffic stream.

Message *NR(b)* is generated to inform the upstream router that the traffic on a particular LSP is being buffered in an attempt to avoid reordering. The message includes an estimate of remaining buffer space in the router that transmits the signal. Upon reception of the *NR(b)* message, a router is permitted to send no more traffic than the signaled amount b. When its buffer reaches a predetermined threshold level, it propagates the message further upstream with a new estimate of its available buffer space. As a result, the traffic is being buffered in the routers in a distributed way. When the PML detects the tail of one of the streams via signaling message *SE* (generated either by a node downstream from a fault or by a PSL after switching traffic to the protection path to facilitate the detection), it sends signaling message *RD*, to inform the upstream routers that it is now capable to receive more traffic. The *RD* message is propagated as far upstream as the *NR(b)* messages have reached. Note that the reliability of signaling in MPLS is ensured by the use of reliable transport protocol. The proposed signaling protocol is described below.

4.1 The Protocol

There are four main categories of nodes in the proposed scheme: a protection switching LSR (PSL), a working path node, a protection merging LSR (PML), and an alternate path[4] node. A complete description of the proposed protocol can be found in [8].

There are three possible sequences of events in a *Protection Switching LSR* (PSL) . First, the PSL may detect an an error on its upstream working path link or, equivalently, receive FIS from a node that is adjacent to a failed link. In both cases, the node switches the incoming traffic onto the working path. Second, the actions taken in Scenario 1 lead to traffic being received on the reverse path. The PSL interprets this as an indication of a downstream fault. In an attempt to reduce packet reordering, incoming traffic on the working path is buffered while reverse path traffic is being sent on. Buffering is stopped either when there is no more traffic on the reverse path (indicated by the *SE* message) or when the buffer capacity is reached. Termination of buffering due to overflow may still lead to packet reordering which is preferable to packet loss that would result otherwise. Finally, buffering of incoming working path traffic can be also triggered by reception of *NR(b)* message from the alternate path. Buffering is stopped when either an *RD* message is received or the buffer capacity is reached. Neither of the two messages is propagated upstream.

Working path nodes can detect link errors on either upstream or downstream link. Actions taken when an error is detected on the downstream link depends on the protection scenario. Fast Reroute (Scenario 1) requires the node to switch the incoming traffic over to the reverse path. Under RSVP-based Backup Tunnels (Scenario 2) the node

[3] Since failure indication signal is typically given the highest priority, it is assumed that $d_{UI} > d'_{UI}$.

[4] The term *alternate path* is used throughout this section for both protection and restoration paths.

send an FIS message to the PSL and starts dropping all working path traffic. Two Path Protection (Scenario 3) assumes that there is a protection path established at every node of the working path. When an error is detected, the traffic is temporarily switched onto the protection path and an FIS message is sent to the PSL to trigger switching over to the preferred restoration path. An upstream link error triggers transmission of an *SE* message at the tail of the working path traffic. The message will be used by the PML as an indication that no more traffic will be arriving over the working path.

The restoration steps can lead to two distinct sequences of events at a *Protection Merging LSR* (PML). First, the more common scenario, the remaining traffic arriving over the working path, terminated by an *SE* message passes through the PML before the alternate path traffic arrives. The PML has to note that it has seen all of the working path traffic so that it does not attempt to block the alternate path traffic. In the other case, the working path and the alternate path traffic streams overlap and the PML attempts to realign them using the backpressure signaling. First, it attempts to buffer incoming traffic on the alternate path while passing on the working path traffic. When a threshold buffer occupancy is reached, the value of b is calculated and *NR(b)* message is sent to the upstream node on the alternate path. The incoming alternate path traffic is buffered until either the end of the working path stream is detected (an *SE* message received) or the buffer reaches its capacity. In both cases, an *RD* message is sent to the upstream node on the alternate path to cancel the request for buffering. Note that PSL buffer overflow is an unlikely event since the backpressure signaling prevents sending more traffic than the buffer can store. The buffer overflow can be caused by an incorrect estimate of value for b, which is possible if the conditions of the network change rapidly, or when the backpressure mechanism exhausts the buffering capacity of all nodes along the alternate path.

An *alternate path* node, on reception of an *NR(b)* message, sends b bytes of traffic downstream and then starts buffering. When a buffer occupancy threshold is reached, it calculates its own value for b and sends an *NR(b)* message upstream. The buffering stops when an *RD* message is received or when the buffer capacity is reached. In both cases, an *RD* message is sent to the upstream node.

4.2 Estimation of Buffering Requirements

The proposed signaling scheme requires the routers to estimate the amount of traffic that is transmitted onto the link after the message is sent and before it is received by the upstream router. The estimate is then subtracted from the available buffer space. Underestimation of the volume of traffic in flight will cause packet loss due to buffer overflow. Overestimation leads to poor utilization of buffer space.

Table 1 outlines the parameters used to calculate the value of b sent with an *NR(b)* message to the upstream node. The node first calculates T_s, the time it will take for a signaling message to reach the upstream neighbor:

b	Maximum number of bytes the node is permitted to transmit
D	Link delay in seconds
L	Line rate in bits/sec
M	Size of a signaling message in bits
Q_c	Current queue occupancy on a node
Q_m	Maximum size of the queue (queue capacity)
R	Data rate in bits/sec
S	Size of a data packet in bits
T_s	Time it will take for a signaling message to reach the upstream neighbor

Table 1. Summary of parameters.

$$T_s = \frac{M}{L} + D. \tag{1}$$

The node then estimates the number of packets that will arrive over the link during this time:

$$N = \left\lceil T_s \frac{R}{S} \right\rceil = \left\lceil \frac{MR}{LS} + \frac{DR}{S} \right\rceil. \tag{2}$$

When a node receives an *NR(b)* message from its downstream peer or a packet from its upstream neighbor after having received an *NR(b)* message, it calculates the value of N. If the difference of the queue limit and queue size is less than N, then the node sends an *NR(b)* signal to its upstream neighbor. The value b gives the estimate of the number of bytes the downstream peer can receive and is given by:

$$b = \frac{S}{8}(Q_m - Q_c - N). \tag{3}$$

It is assumed that nodes are capable of determining all of the parameters required by equations (2) and (3). Two of the parameters are dependent on the character of the offered traffic are S and R and cannot be determined during the network configuration. Instead, the traffic passing through the node is monitored and estimated values for the parameters are calculated using standard exponential averaging. The estimated traffic rate R, after reception of k^{th} packet in the stream is given by:

$$R_k = \alpha R_{k-1} + (1-\alpha)\frac{s_k}{\Delta t}, \tag{4}$$

where s_k is the length of k^{th} packet in bits and Δt is the measured inter-arrival time between packets $k-1$ and k. Estimated average length of a packet S, after reception of k^{th} packet is calculated similarly:

$$S_k = \beta S_{k-1} + (1-\beta)s_k. \tag{5}$$

Weights α and β, $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$, control the relative impact of recent observations versus the longer term average. Given the transient nature of the restoration related events, we choose value $\alpha = \beta = 0.5$ in our experiments. Optimal values of α and β are subjects of further research.

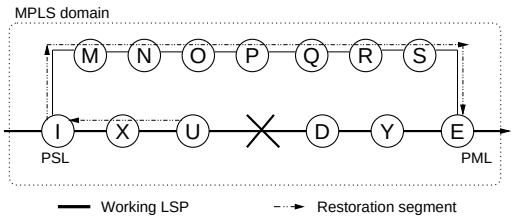

Figure 2. Scenario 1 - Fast Reroute.

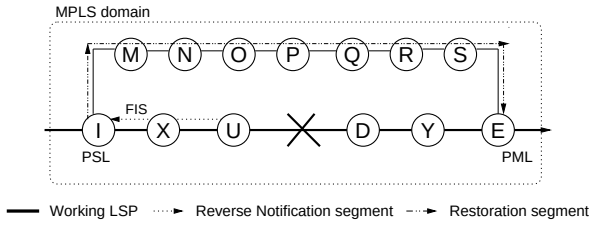

Figure 3. Scenario 2 - RSVP-based Backup Tunnels.

5 Performance Evaluation

Proposed signaling mechanism have been implemented in the LBNL *Network Simulator* (NS)[5]. The primary objective was to study the loss and reordering in the basic protection protection scenarios with and without the proposed signaling mechanism. The experiments used simple network topologies consisting of the working and the alternate paths for each protection scenario under consideration. Simulations have been conducted using three types of traffic: *Constant Bit Rate* (CBR), *Exponentially Distributed UDP* (each generated with the rate of 448 kbps) and *FTP over a TCP connection* (TCP). The delay of each link along the paths was set to 10 ms (unless the experiment called for variable delay, in which case the delay was varied from 10 ms to 250 ms) and its bandwidth was set to 1 Mbps. For majority of the experiments queue sizes were set to 200% and 50% of the buffering space required by equation (2). The first value corresponds to the margin of safety typically employed in routing equipment, while the second is used to evaluate the stability of the proposed protocol in the case of insufficient resources.

The total of 48 experimental setups have been studied (one for each combination of scheme traffic type, protection scenario, queue allocation, and evaluation measure). Due to space limitations, only selected ones are presented here. A full set of graphs and a complete discussion of the results can be found in [9]. In order to distinguish between reordering caused by protection switching and reordering caused by early release of buffered traffic due to buffer overflow, in the presented experiments, the nodes keep holding the traffic in the buffers even if they are full and incoming traffic has to be dropped. This step, in effect, translates packet reordering into loss and allows for easier analysis of the results. In a real network, reordering is preferable to packet loss and, therefore, nodes with full queues would start passing the excess traffic downstream as required by the proposed protocol.

[5]http://www.isi.edu/nsnam/ns.

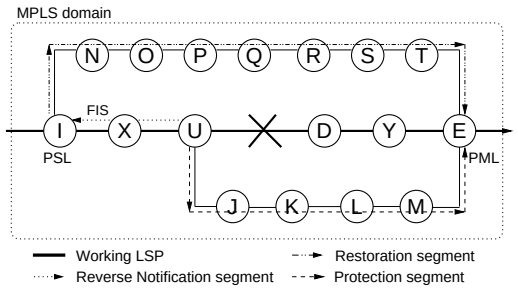

Figure 4. Scenario 3 - Two Path Protection.

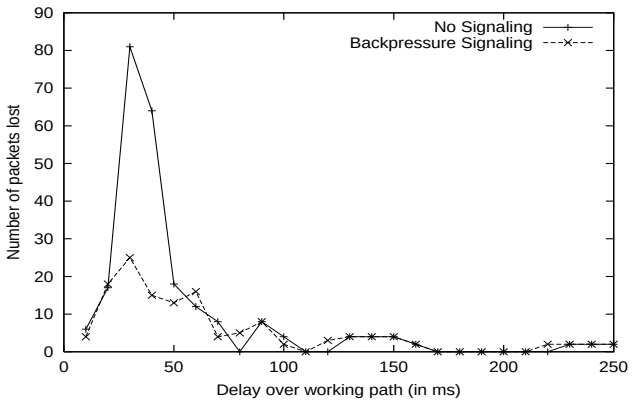

Figure 5. Scenario 1 - Packet loss under TCP traffic with 100% over-subscribed queues.

Experiments for *Scenario 1: Fast Reroute* (Figure 2) varied the delay of each of the links along the entire working path. In this scenario the traffic streams merge at the PSL and, therefore, there is only a limited opportunity for the backpressure signaling to carry out stream realignment. As expected, for CBR and Exponential UDP traffic, the simulation results show that the ability to reduce reordering is controlled by the amount buffer space available in the PSL. As shown in Figure 5, under TCP traffic the scheme helps in reducing packet loss in cases where the traffic stream overlap is relatively small. For longer stream overlaps, congestion control mechanisms of TCP are triggered and the amount of traffic injected into the network is significantly reduced. The lower traffic load results in lesser packet loss regardless whether the scheme is implemented or not.

The delay variation in *Scenario 2: RSVP-based Backup Tunnels* (Figure 3) takes place only along the segment of the working path between nodes *D* and *E*. The traffic streams are merged in the PML, giving the proposed mechanism an opportunity to use the buffers along the restoration path. This results in complete elimination of packet reordering in the cases where the buffering capacity of the nodes along the restoration path is sufficient to perform the traffic stream realignment. Note that the buffering capacity of the nodes may not be fully utilized because the buffering decisions are made using estimates that are based on observation of past traffic. Our experiments did not discover any major inefficiencies in the buffer utilization under the traffic types used for the experiments.

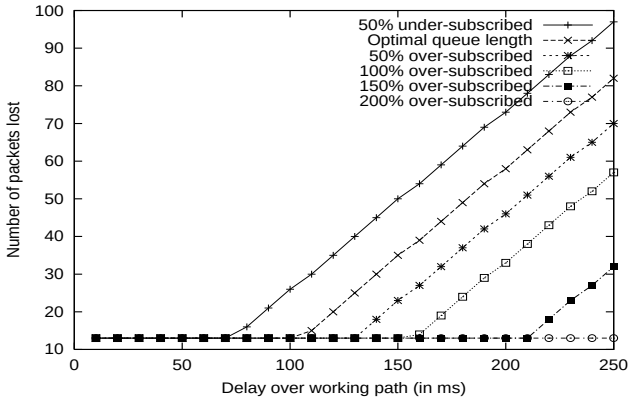

Figure 6. Scenario 2 - Packet loss under Exponential UDP traffic and varying queue lengths.

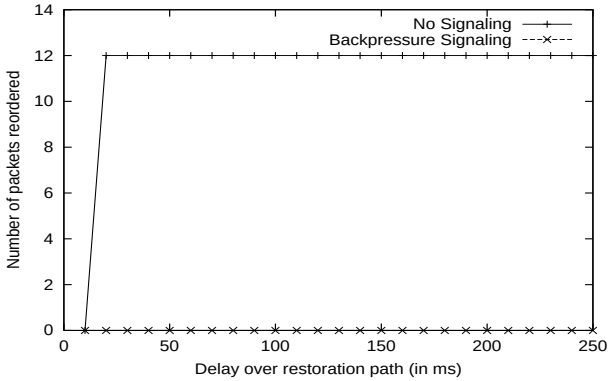

Figure 7. Scenario 3 - Packet reordering under CBR traffic and 50% under-subscribed queues.

Other types of traffic and their impact on the accuracy of the estimates is a subject of our current research. Figure 6 shows the packet loss resulting from insufficient buffering capacity on the alternate path for various buffer sizes under exponentially distributed UDP traffic.

Three traffic streams are being merged at the PML under *Scenario 3: Two Path Protection* (Figure 4). The experiments have been conducted considering two situations that will stress the egress node: varying delay over the working path segment while keeping delays over protection and restoration paths constant, and varying delay over the protection path while keeping the delays over working and restoration paths constant. The effects of overlap of working path and protection segment traffic as well as the effects of overlap of working path and restoration segment traffic are equivalent to those in the RSVP-based Backup Tunnels. The experimental results show that the proposed signaling mechanism reduces the negative effects as outlined in the previous paragraph. The overlap between protection and restoration segment traffic causes packet reordering whose volume is bounded by the limited time the traffic is sent over the protection path. Figure 7 shows that the proposed signaling mechanism completely eliminates this cause of packet reordering under CBR traffic load and under-subscribed queues in the nodes. Similar results have been obtained for other types of traffic and queue sizes.

6 Conclusions

MPLS provides means for fast and resource-efficient service restoration. Packet loss and reordering have been identified as the most significant forms of transient QoS degradation during protection switching. The signaling scheme proposed in this paper reduces the transient effects of protections switching in MPLS networks by distributed temporal realignment of the traffic streams. It is independent of the underlying protection mechanism and its integration with three current MPLS service restoration schemes outlined in the paper. The experiments confirm that all three existing MPLS protection schemes suffer from packet loss and/or packet reordering, during protection switching and show that the proposed signaling protocol can help to minimize packet reordering while maintaining or improving the packet loss performance.

References

[1] E. 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-04.txt]*, May 2002.

[3] P. Pan et al., "Fast reroute extensions to RSVP-TE for LSP tunnels." Work in progress *[draft-ietf-mpls-rsvp-lsp-fastreroute-00.txt]*, January 2002.

[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] K. Owens, V. Sharma, S. Makam, and C. Huang, "A path protection/restoration mechanism for MPLS networks." Work in progress *[draft-chang-mpls-path-protection-03.txt]*, July 2001.

[6] R. Bartoš and M. Raman, "A heuristic approach to service restoration in MPLS networks," in *Proc. of the 2001 IEEE International Conference on Communications (ICC), Helsinki, Finland*, June 2001.

[7] G. Ahn, J. Jang, and W. Chun, "An efficient rerouting scheme for MPLS-based recovery and its performance evaluation," *Telecommunication Systems*, vol. 9, pp. 481–496, March–April 2002.

[8] R. Bartoš and A. Gandhi, "A control protocol to minimize the transient effects of MPLS service restoration," Tech. Rep. TR 02-09, Dept. of Computer Science, Univ. of New Hampshire, September 2002.

[9] A. Gandhi and R. Bartoš, "Dynamic issues in MPLS service restoration," Tech. Rep. TR 01-10, Dept. of Computer Science, Univ. of New Hampshire, December 2001.