Proceedings of the 6th WSEAS International Conference on Applied Informatics and Communications, Elounda, Greece, August 18-20, 2006 (pp87-92)

Effects of Rerouting After Service Disruption in MPLS based networks

BJØRN JÆGER, *Bjorn.Jager@hiMolde.no*
KETIL DANIELSEN, *Ketil.Danielsen@hiMolde.no*
Molde University College, Molde, NORWAY

July 19, 2006

Abstract: - This paper considers the effects of dynamic restoration routing after a failure in MPLS based networks. MPLS technology enables scalable VPNs and end-to-end QoS over the Internet which requires efficient fault recovery in presence of link or node failures. A major factor on network performance after a failure is the transient congestion that results from restored connections trying to retransmit packets lost since the failure. We focus on the importance of appropriate selection of a rerouting algorithm to control the transient congestion, and we present results from a simulation-based performance study of several routing algorithms. The results show that proper selection of fault recovery routing algorithm can improve MPLS-network performance after a failure.

1 Introduction

Minimizing the impact of failures on services in connection oriented packet switched networks, such as the emerging quality-of-service (QoS) enabled Internet, is a problem of growing importance as the Internet is becoming the common transport network for a wide range of communication services including services with strict QoS-requirements like VoIP, VoD and Video Conference. Internet provides traffic engineering [6] possibilities by the Multi Protocol Label Switching (MPLS) scheme. In MPLS [5], packets are encapsulated at ingress routers with labels that are used to forward the packets along Label Switched Paths (LSPs). These LSPs are virtual traffic trunks that carry flow aggregates generated by classifying the packets arriving at the ingress routers of an MPLS network into Forwarding Equivalent Classes (FEC). The aggregation into FECs combined with explicit routing of bandwidth-guaranteed LSPs enables service providers to perform traffic engineering in their networks [6]. The new QoS requirements put forward a need for including new fault recovery mechanisms, and a general framework for MPLS-based recovery has been defined [7, 12].

A primary goal of fault recovery is to restore service leaving the users unaware of any internal failures in the network. Restoration of service leads to a new QoS routing problem in the Internet where it is necessary to dynamically reroute traffic affected by a failure [7, 11, 12]. The MPLS-based recovery framework defines a Path Switch Router (PSR) as a point-of-repair that actually performs the reroute operation. The PSR can be the ingress router of an LSP if end-to-end rerouting is used, or the PSR can be the upstream router next to the failure if patch-rerouting is used. In the case of patch-rerouting, when there is a failure, *all* the LSPs using the failed component are rerouted as one unit across the failed component. This is different from the end-to-end rerouting where the PSR is equal to the ingress router of the LSPs which makes it possible to reroute each LSP individually. The major advantages by end-to-end rerouting over patch rerouting are; better utilization of available resources since the demand is being restored in small units

of bandwidth, better adoption to changes in policies and equipment among the various service providers, and it handles both node and link failures. This comes at the cost of increased restoration time [7].

In end-to-end rerouting, the failure is detected by the node upstream to the failure, who sends a notification message to the source router (i.e. the PSR) of the LSPs traversing the failed component. Upon receiving the failure notification the PSR first identifies affected LSPs, then it search for possible backup LSPs (these may be precalculated for each source-destination pair). Then the PSR selects the path with minimum cost according to a cost function as discussed below, before finally establishing the disrupted connection along a new LSP. Each PSR maintains a link cost database which keeps an updated record of the residual (available) bandwidth on each link. The residual bandwidth is defined as the difference between the link capacity and the amount of bandwidth already in use by the active and the backup paths traversing the link. This information is obtainable from routing protocol extensions described in [6].

A network failure of a router or a link typically cause many LSPs to be disrupted *simultaneously*, some of which may carry traffic that requires retransmission of lost packets. This introduce a burst that is not seen under normal operation. Packets to be retransmitted creates a backlog at the PSR who tries to get rid of the backlog as fast as possible. This and the collective attempt to reestablish connections cause traffic congestion and undesirable transients in the network. In [2] Tipper et al. showed that the additional transient load imposed on the network by the retransmissions typically cause buffer overflow at both the source node and at the buffers at other nodes along the path to the destination. In addition, packets dropped after reconnection due to this overflow also need to be retransmitted, adding a positive feedback to the source, further worsening the congestion. As noted in [2], congestion control schemes are not effective in preventing congestion after a major failure since the overload at network nodes is mainly due to the rerouted connections needing to work off their backlogs simultaneously, and since the backbone links used are high-speed links carrying a large set of data already on-route when feedback based control mechanisms are used. In [3] it is shown for ATM-networks that rerouting after a failure can control the transient congestion to some extent.

In this paper we show the effect of various restoration routing algorithms on the transient congestion in MPLS-networks. The metrics used are the duration of the congestion in seconds and the spatial distribution of the congested area.Note that traditional performance metrics like number of connections rejected or amount of bandwith restored can not be used since all connections are restored. Here we focus on the dynamic performace of the network queues during recovery of all failed LSPs.

2 Rerouting Strategies

We study the effect on network performance of various strategies to select restoration paths. Each path is assigned a cost according to the chosen strategy by letting the path cost W for each candidate path j be defined as the sum of link costs V for each link ℓ along the path:

$$W_j = \sum_{\ell \in L} V_\ell$$

where L is the ordered set of directed links in the network, and j is an element in the set of candidate paths between the ingress and egress MPLS-router.

The first strategy is to spread the load over a large geographic area by using a large set of links in order not to seriously degrade the performance of any particular link. A scheme that use this approach is *Load Distribution Among Paths (LDAP)* [10] were the link cost is set equal to the bandwidth being used on the link by active- and eventual previously established backup LSPs. This is equivalent to using the negative of the residual capacity R as the cost for link l: V_ℓ = -R_ℓ. The path cost W for each possible backup path j is defined as the sum of link costs along the path:

$$W_{j,LDAP} = \sum_{\ell \in L} -R_\ell \qquad (1)$$

Another approach is to select the path that minimize end-to-end delay. Here we use a scheme that minimize the additional *increase* in delay that a LSP will experience when routed along a path, thus the scheme is called *Minimum Incremental Delay (MID)*. The formula for MID is found by taking the derrivative of the link queueing delay in a M/M/1 type queueing model as discussed in [1]. This gives a link cost formula: $V_\ell = \frac{C_\ell}{(R_\ell)^2}$, where C_ℓ is the link

capacity of link l. Thus the path cost is:

$$W_{j,MID} = \sum_{\ell \in L} \frac{C_\ell}{(R_\ell)^2} \quad (2)$$

A third strategy is to try to isolate and restrict the area of the congestion. This can be done by using *Minimum Hop (MH)* as path cost letting each link having a cost of 1. MH results in the number of nodes and links directly handling rerouted connections being minimized since the restoration paths are as short as possible. The formula for MH path cost is:

$$W_{j,MH} = \sum_{\ell \in L} 1 \quad (3)$$

The fourth and last aproach is similar to LDAP but in addition it looks at the end-to-end distribution of residual capacity along each candidate path. Paths with increasing amount of link residual capacity along a path will be preferred. To acheive this an additional cost δ_ℓ is added to each link if $R_{\ell+1} < R_\ell$, otherwise the link cost is -R_ℓ as for LDAP. Also, the additional cost is weighted according to how far from the ingress router it is. The weight is set equal to the hop index, H i.e. between link 2 and 1 have weight 1, between link 3 and 2 have weight 2, and between 4 and 3 have weight 3. This scheme termed *Incremental Residual Capacity (IRC)* is formulated as:

$$W_{j,IRC} = \sum_{\ell \in L} (-R_\ell - \delta_\ell)$$

$$where\ \delta_\ell = H(max(0, R_\ell - R_{\ell+1})) \quad (4)$$

3 Performance Evaluation

We consider a distributed rerouting approach that is consistent with routing protocol extensions for Traffic Engineering [8, 9]. Each router maintains a topology database containing the cost of using each link in the network. Also, to speed up the routing computation each node maintains a precomputed set of routes between each source destination pair that is determined from the topology of the network and is restricted by a hop count limit. The database of link cost is updated periodically with each router notifying the other routers of its current link cost at the update times using a flooding approach. Also, asynchronous updating of the link cost occurs whenever a link utilization changes by an amount exceeding a predefined threshold. Routing is accomplished at connection setup time by the ingress LSR of the LSP using it's local cost database and the set of predefined paths to select the minimum cost path from the ingress to egress LSR. The rerouting schemes proposed above will fit easily into this format using the appropriate link cost in calculation of the route.

A simulation based performance study was conducted to evaluate the effect of the rerouting strategies on the transient congestion. We adopted the nonstationary simulation methodology of [4] to observe the behavior of the average number in the queueing system versus time for all links in the network. The basic approach is to run the simulation a number of times and average the quantities of interest across the ensemble of independent runs at a particular time instant. Many such points may be obtained at different time instants and the behavior of the system can be studied as a function of time.

The 10 node network with 42 bidirectional links shown in Figure 1 was simulated. The topology was selected since it has a average node degree of 4.2 which is a typical value in the range of many existing networks, and due to the large number of alternate paths between any pair of nodes. A simulation model of the network was developed in the ns-2 package [13] combined with MPLS [14]. The model focus on a comparative analysis of the rerouting algorithms during fault recovery under a common set of simplified assumptions described here. The rerouting algorithms were implemented in the simulation in a distributed fashion with each router maintaining a local database of the link cost with periodic cost updates. A hop count limit of four links was used in the model to restrict the number of feasible paths and speed up the route selection.

All nodes in the network are MPLS routers, all links are 1 mb/s duplex links, and the link output queues are FIFO-queues with a size of 20 packets giving a system size (including the server) of 21. All traffic sources k generate 512 Byte packets at a Poisson rate equal to the demand rate d_k. The time to detect a failure and begin notification of the source nodes was set to two seconds. It was assumed that all dropped packets need retransmission and the resulting backlogs at the PSR was determined using the normalization approach given in [2].

A set of 50 LSPs between various source destination pairs were set up and were running in stady state when a link failure was introduced. The average network link utilization was 0.334, thus the network has enough residual capacity to restore all failed connections regardless of the routing algorithm used.

At $t = 1$ the link 2-4 breaks and node 2 sends an LDP notification upstream to the ingress LSR. Upon receiving the notification, the ingress LSR performs the restoration by first identifying failed LSPs. Then it search for new backup LSP for each failed LSP, calculate the cost for each of these and select the least cost path, which is established. The time to detect a failure, send notification and do the restoration steps were simplified as a 2 second delay, after which backup paths were loaded. Retransmission of lost traffic (backlogs) were simulated by having sources send at maximum rate until all lost packets are retransmitted. The queue sizes were sampled each .01 second, and a 10 run average is shown in the figures.

It was assumed that all dropped packets need retransmission and the resulting backlogs at the PSR was determined using the normalization approach given in [2]. The metrics used to characterize the transient congestion were duration and spatial distribution following the approach in [4]. Here the duration is measured by two numbers; the average length of the congestion over all congested links, and the time of the link that were congested for the longest period. The spatial distribution of the congested area is measured by the total number of links used by the bachup LSPs selected, and the number of these links that actually became congested.

The failure affected nine LSPs from PSR 9 with an aggregate load of 0.84. The traffic rates d_k, number of possible restoration paths P_k, and the paths selected by the rerouting schemes for the 9 affected LSPs are given in Table 1. Table 1 illustrates that the routes selected by the four algorithms are as expected from their definition. Specifically, the MH scheme picks 1 and 2 hop paths resulting in the fewest number of links used. Where as, the LDAP algorithm uses four hop paths for all LSPs and affects the greatest number of nodes. As for LDAP the IRC scheme also selects four hop paths, but we see that these are different from LDAP due to the requirement of having increasing residual capacity along the path. The MID approach prefers shorter paths while avoiding heavily loaded links. Since spare capacity is plentiful all LSPs are restored regardless of the routing scheme used. Thus by the standard survivability metrics such as percentage of connections blocked or percentage of bandwidth restored, all the strategies are equivalent. However the transient congestion illustrates marked differences between the strategies as can be seen in Figure 2. The figure shows for each routing scheme the average number in the system versus time after traffic restoration for three selected network links. The results are the ensemble average of 10 simulation runs. There is little variance in the simulation output in the various runs which results in narrow confidence intervals on the simulation results and the intervals have been left off the plots for clarity.

Consider the plot for link 9-1 shown in Figure 2. One can see that after the restoration of LSPs the buffer at link 9-1 quickly saturates under all four schemes, but the amount of time the link is overloaded varies with the routing scheme. Specifically, the MH approach congests link 9-1 much longer than the other algorithms.

By examining plots similar to Figure 2 for all the links in the network, we can identify the links that become congested and the duration of the congestion. A link was considered congested if the average number in the queueing system at a link attained the buffer size, and congestion was deemed over when the number in system fell below 5. The two first rows in Table 2 shows the duration of the congestion for each algorithm. Row 1 shows the average congestion time which is the mean of the time each link is congested for those that become congested. Similarly, the maximum congestion time listed in row 2 is the length of time for all links that become congested to clear. Rows 3 and 4 shows the spatial distribution by the number of links used for each algorithm and the number of links that congest.

4 Conclusions

In this paper we presented a performance study of the MPLS fault recovery routing in the QoS enabled Internet. Variour rerouting strategies suitable for traffic restoration after a failure were studied. Specifically we studied, the Minimum Hop (MH) scheme – which ensures the number of nodes directly effected by the rerouting is minimized, the Minimum Incremental Delay (MID) approch focusing on minimizing the end-to-end delay, the Load Distribution

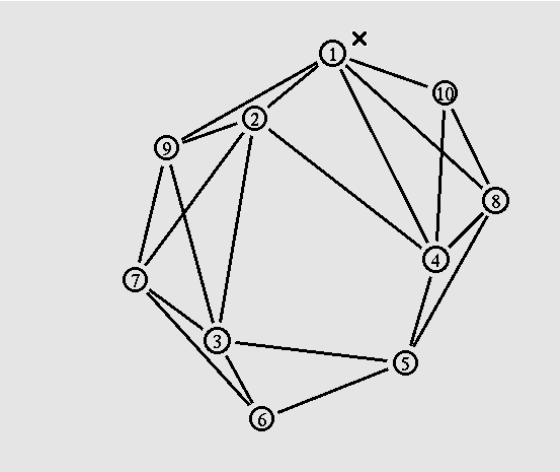

Figure 1: Network topology

Among Paths (LDAP) scheme – which picks the source destination route that has the maximum residual capacity and the Incremental Residual Capacity (IRC) scheme – which balances the load on the outgoing links of a node that must reroute several connections. The results shows that MID and MH congests a smaller area for a longer time when compared with LDAP and IRC that congests a larger area for a smaller time. Thus, there is a tradeoff tradeoff between what parameters ISPs want to prioritze.

We have shown that the choice of routing algorithm has effect on MPLS network performance during fault recovery. Futhermore, it was illustrated that in addition to standard network survivability metrics, it is important to include metrics that characterise the real-time behaviour of the network immediately after a failure.

References

[1] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 1992.

[2] D. Tipper, J. Hammond, S. Sharma, A. Khetan, K. Balakrishnan, and S. Menon, "An Analysis of the Congestion Effects of Link Failures in Wide Area Networks", *IEEE Jrnl. Sel. Areas Comm.*, vol.12:179-192, 1994.

[3] W.P. Wang, D. Tipper, B. Jaeger and D. Medhi, "Fault Recovery Routing in Wide Area Packet Networks", *Proceedings of 15th International Teletraffic Congress*, Washington, DC, June 1997.

[4] W. Lovegrove, J. Hammond, and D. Tipper, "Simulation Methods for Studying Nonstationary Behavior of Computer Networks", *IEEE Jrnl. Sel. Areas Comm.*, vol.8:1696-1708, 1990.

[5] E. Rosen, A. Viswanathan and R. Callon, "RFC 3031 - Multiprotocol Label Switching Architecture", Network Working Group, Jan. 2001.

[6] VD. Awduche, J. Malcolm, J. Agogbua, M. O'Dell and J. McManus, "RFC 2702 - Requirements for Traffic Engineering Over MPLS", Network Working Group, Sep. 1999.

[7] V. Sharma and F. Hellstrand, "RFC 3469 - Framework for Multi-Protocol Label Switching (MPLS)-based Recovery", Network Working Group, Feb. 2003.

[8] D. Katz, K. Kompella and D. Yeung, "RFC 3630 - Traffic Engineering (TE) Extensions to OSPF Version 2", Network Working Group, Sep. 2003.

[9] H. Smit and T. Li, "RFC 3784 - Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE)", Network Working Group, June 2004.

[10] B. Jager and D. Tipper, "Prioritized Traffic Restoration in Connection Oriented Networks", *Computer Communication*, vol.26:2025-2036, 2003.

[11] Wayne D. Grover, "Mesh-based Survivable Transport Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking", Prentice Hall, Upper Saddle River, NJ, 2003.

[12] E. Harrison, "Protection and restoration in MPLS networks", White paper, Data Connection, Oct 2001.

[13] Kevin Fall and Kannan Varadan, The Network Simulator ns-2, version 2.26, available at http://www.isi.edu/ nsnam/ ns.

[14] A. Gaeil and C. Woojik, "Design and Implementation of MPLS Network Simulator (MNS) supporting LDP and CR-LDP", Proceedings of the IEEE International Conference on Networks (ICON'00), September 2000.

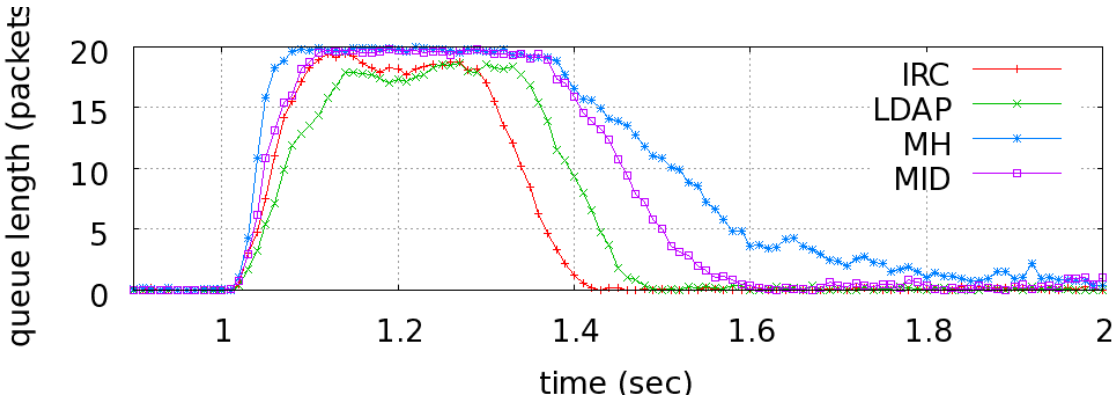

Figure 2: Link 9-1 queue behaviour

Table 1: Comparison of paths selected

d_k	P_k	MID	MH	LDAP	IRC
0.15	12	9–1–10	9–1–10	9–3–5–8–10	9–3–5–4–10
0.14	17	9–1–8	9–1–8	9–1–10–4–8	9–2–3–5–8
0.12	17	9–1–4	9–1–4	9–2–3–5–4	9–1–8–5–4
0.11	21	9–3–5	9–3–5	9–2–1–8–5	9–2–7–3–5
0.09	17	9–3–5–4	9–1–4	9–7–6–5–4	9–7–6–5–4
0.08	8	9–2–1	9–1	9–7–3–2–1	9–7–3–2–1
0.06	21	9–3–5	9–3–5	9–1–2–3–5	9–1–2–3–5
0.05	17	9–7–6–5–8	9–1–8	9–1–4–5–8	9–1–4–5–8
0.04	17	9–7–6	9–3–6	9–1–2–7–6	9–7–3–5–6

Table 2: Transient congestion metrics, avg. link load 0.334, no rejections

Routing algorithm	MID	MH	LDAP	IRC
Avg time congested	0.4	0.46	0.286	0.41
Max time congested	0.45	0.65	0.4	0.6
Links used	13	7	23	23
Links congested	2	2	5	5