DiffServ and MPLS – Concepts and Simulation

Raymond Law and Srihari Raghavan
Bradley Department of Electrical and Computer Engineering
Virginia Polytechnic Institute and State University
{rlaw,srihari}@vt.edu

Abstract

Differentiated Services (DiffServ) is scalable for deployment in today's Internet, and Multiprotocol Label Switching (MPLS) provides fast packet switching and the opportunity for traffic engineering. Thus, the combination of DiffServ and MPLS presents a very attractive strategy to backbone network providers. This paper attempts to explain the concepts of DiffServ + MPLS and illustrate its effectiveness by performing a simulation using Network Simulator (ns-2). The results show the fast rerouting feature of MPLS and how it alleviates the problem of link failures in DiffServ networks.

1 Introduction

The motivations for DiffServ + MPLS include user demands for consistent QoS guarantees, efficient network resource requirements by network providers, and reliability and adaptation of node and link failures. DiffServ provides scalable edge-to-edge QoS, while MPLS performs traffic engineering to evenly distribute traffic load on available links and fast rerouting to route around node and link failures. Moreover, MPLS can be deployed over a wide variety of link layer technologies such as IP, ATM, and Frame Relay. MPLS support for DiffServ over IP is discussed in this paper. Information regarding MPLS support for DiffServ over ATM and Frame Relay can be found in [6]. This paper first explains the concepts behind DiffServ, MPLS, and DiffServ + MPLS. It then presents results from an event-driven simulation using Network Simulator (ns-2) [10] to show the fast rerouting feature of MPLS and how it solves the problem of link failures in a DiffServ network.

2 Backgrounds [Raymond Law]

2.1 Differentiated Services

DiffServ [3] provides scalable and "better than best-effort" QoS. DiffServ routers is stateless and do not keep track of individual microflows, making it scalable to be deployed in the Internet. The DiffServ Code Point (DSCP) in the Differentiated Services (DS) field of the IP header identifies the Per Hop Behavior (PHB) associated with the packet, which is used to specify queuing, scheduling, and drop precedence [3]. There are three defined PHBs: (i) Best effort, (ii) Assured Forwarding (AF), and (iii) Expedited Forwarding (EF) [1]. A PHB group is a set of PHBs that must maintain the order of packets in microflows [5]. A Behavior Aggregate (BA) is an aggregate of microflows with the same DSCP [3].

At the ingress node in a DiffServ domain, the DSCP value is determined based on multifield classification [2] of the incoming packet. At the interior nodes, the PHB is determined from the DSCP and appropriate QoS treatment is applied to the packet. At the egress node, the packet is routed to the next hop in the next domain [2]. Traffic conditioning is performed at the boundary nodes to ensure the traffic streams conform to the traffic conditioning agreement (TCA) between two domains [2].

2.2 Multiprotocol Label Switching

MPLS Label Switching Routers (LSR) [4] provide fast packet forwarding compared to routers, with lower price and higher performance. They also offer traffic engineering, resulting in better utilization of network resources such as link capacity as well as the ability to adapt to node and link failures [1]. To deploy MPLS in an IP network, the Shim label header [4] is inserted between the link layer and network layer headers. The Label field identifies the Forwarding Equivalence Class (FEC), which is a group of packets that are forwarded to the same next hop, specified by the Next Hop Label Forwarding Entry (NHLFE), in the same manner (i.e. over the same path with the same forwarding treatment). The EXP field can be used to select the appropriate PHB for the packet.

MPLS consists of two components [1]. The conventional routing protocol uses the FEC-to-NHLFE mapping (FTN) to forward unlabeled packets. The label binding and distribution uses the Label-to-NHLFE mapping (ILM) to forward labeled packets. A Label Switching Path (LSP) [4] is set up as follows. The ingress LSR sends a Label Request message toward the egress LSR, which sends back a Label Mapping message back to the ingress LSR [1]. During the propagation of these label messages, all LSRs on this path use these label information to set up their ILMs so that packets can be forwarded using the label headers [6]. MPLS uses the label swapping mechanism to forward labeled packets through a MPLS domain. The ingress LSR pushes a label to a packet and the egress LSR pops a label from a packet. Intermediate LSRs perform both a label pop and label push [4].

3 DiffServ + MPLS [Raymond Law]

There are two basic problems for MPLS support of DiffServ. First, the DSCP is carried in the IP header, but the LSRs only examine the label header. Second, the DSCP has 6 bits but the EXP field has only 3 bits. There are two solutions defined in [6] to remedy these two problems: (i) EXP-Inferred-PSC LSP (E-LSP), and (ii) Label-Only-Inferred-PSC LSP (L-LSP). The LSR DiffServ label switching behavior, as discussed below, has four stages which are defined in [6].

3.1 Incoming PHB Determination

For E-LSP, the EXP-to-PHB mapping can be either preconfigured or explicitly signaled during the E-LSP establishment. The LSR determines the PHB to be applied to the incoming packet by looking up the EXP field in the EXP-to-PHB mapping [6]. For L-LSP, the EXP-to-PHB mapping is a function of the PSC carried on the L-LSP, and is set up during the L-LSP establishment. Therefore, the PSC (i.e. queuing and scheduling) is already known to the LSR based on the Label field. The LSR then determines the drop precedence, hence the PHB, to be applied to the incoming packet by looking up the EXP field in the EXP-to-PHB mapping [6].

3.2 Outgoing PHB Determination with Optional Traffic Conditioning

A DiffServ LSR may perform marking, policing, and shaping on the incoming traffic streams, potentially changing the outgoing PHBs associated with non-conforming packets in the incoming traffic streams [6]. Thus, the incoming and outgoing PHB may be different. If no traffic conditioning is performed, the incoming and outgoing PHB are identical.

3.3 Label Forwarding

Each DiffServ LSR must know the DiffServ context for a label, which is stored in the NHLFE for each outgoing label [6]. A DiffServ context consists: (i) LSP type, (ii) Supported PHBs, (iii) EXP-to-PHB mapping for an incoming label, and (iv) PHB-to-EXP mapping for an outgoing label. This information is populated into the ILM and FTN during label setup and is used to forward packets to the next hop. A label can be mapped to multiple NHLFEs in the ILM or FTN to allow multiple next hops in the case of load balancing and fast rerouting.

3.4 Encoding of DiffServ Information into Encapsulation Layer

For E-LSP, the PHB-to-EXP mapping can be either preconfigured or explicitly signaled during E-LSP-establishment. The LSR determines the EXP value to be written to the outgoing packet by looking up the PHB in the PHB-to-EXP mapping [6]. For L-LSP, the PHB-to-EXP mapping is a function of the PSC carried on the L-LSP, and is set up during the L-LSP establishment. The LSR determines the EXP value to be written to the outgoing packet by looking up the PHB in the PHB-to-EXP mapping [6].

3.5 EXP-Inferred-PSC LSP (E-LSP)

E-LSP determines the PHB of a packet solely from the EXP field, and thus can support up to only 8 PHBs per E-LSP. The EXP field conveys the queuing, scheduling, and drop precedence to the LSR. PHB signaling can be used to explicitly signal the supported PHBs during LSP setup, but is not required (i.e. preconfigured PHBs) [6].

3.6 Label-Only-Inferred-PSC LSP (L-LSP)

Packets in a microflow must maintain the same order from the ingress LSR to the egress LSR, so they belong to the same PHB Scheduling Class (PSC) [5], which is a PHB group such that the order of packets in the group must be preserved, and are placed in a common queue. The set of BAs whose order must be maintained during transmission constitutes an Ordered Aggregate (OA) [5].

L-LSP determines the PHB of a packet from both the Label and EXP fields. The Label field determines the PSC (queuing and scheduling) while the EXP field determines the PHB (drop precedence). An arbitrarily large number of PHBs can be supported. The DiffServ object defined in the Resource Reservation Protocol (RSVP) extension or the DiffServ TLV defined in the Label Distribution Protocol (LDP) extension can be used to support PHB scheduling group signaling, which is used to signal the PSC during L-LSP establishment [6].

4 Simulation [Srihari Raghavan]

4.1 Simulation Aims and Environment

The aim of this simulation is to underline the need of integration of MPLS with DiffServ. MPLS rerouting is shown in this simulation as the motivating reason behind the MPLS and DiffServ integration. MPLS traffic engineering is another important reason for MPLS and DiffServ integration, but will not be dealt with here. The environment consists of ns-2 network simulation software in Linux operating system. Two ns-2 patches, the DiffServ patch [11] and the MPLS patch [12] were applied to execute the simulations.

4.2 Simulation Setup and Details

Figure 4.1 below shows the topology that was used in the simulation.

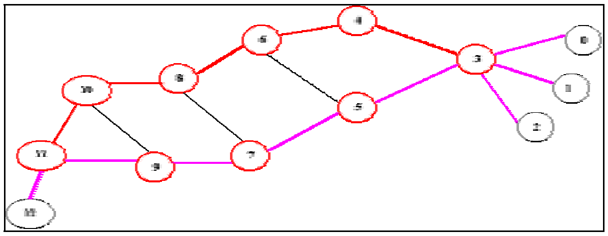

Figure 4.1. Simulation Topology [13]

The red-colored nodes represent MPLS LSRs, while the nodes, 0,1, and 2 represent source nodes. Node 12 acts as the destination node. Node 0 is the origin node for the User Datagram Protocol (UDP) traffic. This is implemented using the ns-2 Constant Bit Rate (CBR) traffic agent with customizable packet sizes and inter-packet intervals. Nodes 1 and 2 are the source nodes for Transmission Control Protocol (TCP) traffic. Ns-2 TCP agents are used for the same. The link capacities are one Megabit per second (Mbps). The idea is that both the TCP and UDP traffic are allowed to mix on a single link between nodes 3 and 11 via node 7. The second stage involves a significant increase in UDP traffic resulting in negative impact for the TCP traffic. The third stage uses DiffServ to alleviate the negative impact problems for the TCP traffic. The fourth stage introduces link failures in the network resulting in negative impact to both UDP and TCP traffic even with DiffServ enabled. The impact is characterized by measuring bandwidth received at the destination node 12 from both the TCP and UDP sources. The fifth and final stage introduces MPLS in these networks and uses an MPLS-based fast rerouting algorithm proposed by Haskin [13]. This fast rerouting scheme helps DiffServ networks by allowing continuous data flow even in the presence of link failures in the networks.

4.3 Simulation Results

The simulation results are in the form of bandwidth graphs with bandwidth in Mbps in Y-axis and time in seconds in the X-axis. The topology being followed is the one in figure 4.1. Figure 4.2 shows the results for the simulation experiment with both DiffServ and MPLS not enabled. Here, figure 4.2a shows a case where only TCP traffic is present in the network from node 1 to node 12. Figure 4.2b shows a case where only UDP traffic is present in the network from node 0 to node 12. The TCP bandwidth shown in figure 4.2a is as measured from the source. The UDP bandwidth shown in figure 4.2b is as measured from the destination.

Figure 4.3 shows the results for the simulation experiment with high rate of UDP traffic and TCP traffic being mixed in the same link between nodes 3 and 11 via node 7. This graph shows that TCP is getting negatively affected on a significant increase in UDP traffic. This is because of the nature of the TCP source. A TCP source undergoes congestion control phase when it senses congestion in the network. Figure 4.4 shows the results for the simulation experiment with DiffServ enabled in the network. Here, TCP traffic is marked AF11 and UDP traffic is marked EF. Node 2 is another TCP source whose traffic is marked Best Effort (BE). The DiffServ components introduced in the simulation are Class Based Queuing (CBQ) schedulers, DiffServ conditioners with AF and EF profiles. The EF peak-rate is set at 0.8Mbps with AF peak-rate at 0.3 Mbps. The 1Mbps link is divided in the CBQ as 4:4:2 for EF, AF, and BE respectively.

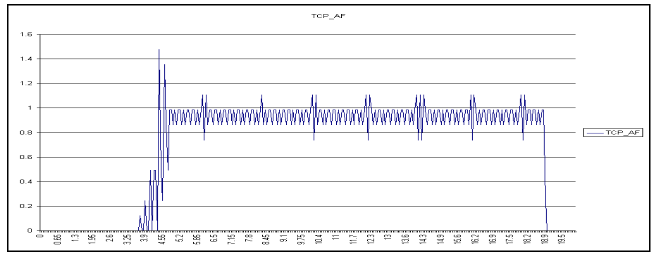

Figure 4.2a. TCP traffic (no UDP)

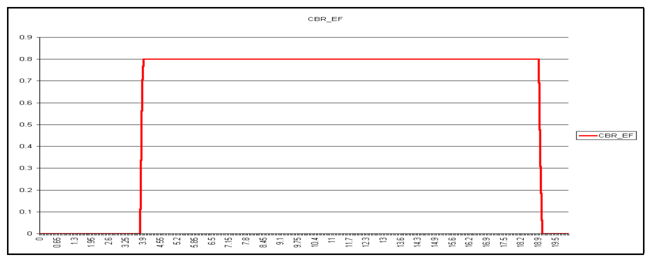

Figure 4.2b. UDP traffic (no TCP)

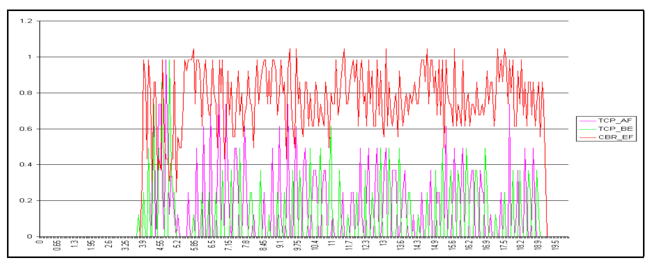

Figure 4.3. TCP with high UDP traffic

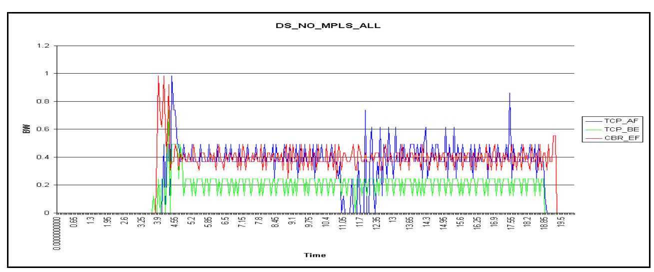

Figure 4.4. DiffServ enabled network

Figure 4.5 displays the results of a simulation run with DiffServ enabled but with link failure in the link between node7 and node 9. This shows that even with DiffServ enabled, both UDP and TCP traffic are negatively impacted. The IP layer rerouting will act as a solution here but the rerouting time or duration for the rerouting to take place depends upon the IP routing protocol being followed in the network.

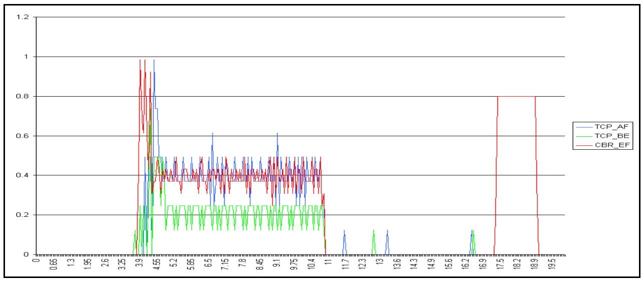

Figure 4.5. Link failure in the DiffServ network

Figure 4.6 shows the effect of using MPLS-based rerouting in the above case. In this case, both MPLS and DiffServ are enabled. MPLS rerouting is enabled as a pre-established Explicitly Routed Label Switched Path (ER-LSP) called the protection-LSP from node 3 to node 11 via node 4. The rerouting algorithm being followed is the one proposed by Haskin. Here, the nodes 5,7, and 9 along the working-LSP frequently check for link failures in any of its interfaces and if link failures happen, they route the traffic through the pre-configured protection-LSP. The main overhead behind using MPLS is the signaling aspect. Depending upon the signaling protocol used, this could be an area of concern. RSVP-TE, with its soft-state signaling causes higher overhead than what CR-LDP does. The MPLS signaling processing overhead and memory storage overhead at the LSRs for maintaining per-LSP state information needs to be considered before deciding on MPLS and DiffServ integration.

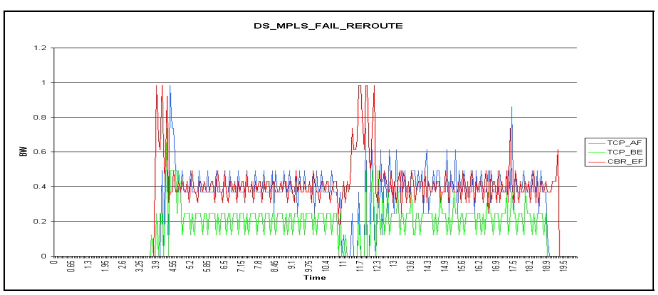

Figure 4.6. MPLS-based fast rerouting

5 Conclusions

Concepts of DiffServ + MPLS was presented, focusing on the operational procedures of E-LSP and L-LSP. This was followed by some simulation results to underline the need of DiffServ + MPLS based on MPLS-based rerouting.

Currently, only unicast operation is defined in [7], while multicast communication requires further study. Use of the EXP field to support Explicit Congestion Notification (ECN) together with DiffServ is also under development. Finally, there is not a cookbook approach as to how to combine both E-LSP and L-LSP efficiently, leaving open the research opportunities on the topic of DiffServ + MPLS.

Future work should concentrate on simulations with more realistic topologies and different TCP sources like HTTP web traffic as observed in [14]. MPLS-based traffic engineering is another important advantage to DiffServ. Similar experiments can be done to highlight these advantages.

Acknowledgement

We would like to acknowledge Dr. L. A. DaSilva's opinions and comments on our ideas.

References

[1] B. Davie, and Y. Rekhter, *MPLS Technology and Applications*, Morgan Kaufmann, San Francisco, CA, 2000
[2] U. Black, *MPLS and Label Switching Networks*, Prentice Hall, Upper Saddle River, NJ, 2001
[3] S. Blake et al, *An Architecture for Differentiated Services*, RFC 2475, December 1998
[4] E. Rosen, A. Viswanathan, and R. Callon, *Multiprotocol Label Switching Architecture*, RFC 3031, January 2001
[5] D. Grossman, *New Terminology for DiffServ*, IETF Internet Draft, draft-ietf-DiffServ-new-terms-04.txt, March 2001
[6] F. Faucheur et al, *MPLS Support of Differentiated Services*, IETF Internet Draft, draft-ietf-mpls-diff-ext-08.txt, February 2001
[7] IETF, *Differentiated Services (DiffServ)*, http://www.ietf.org/html.charters/DiffServ-charter.html (current 29 Apr. 2001).
[8] IETF, *Multiprotocol Label Swtiching (mpls)*, http://www.ietf.org/html.charters/mpls-charter.html (current 29 Apr. 2001)
[9] Ns-2, "The Network Simulator – ns-2", http://www.isi.edu/nsnam/ns/ (current 29 Apr. 2001)
[10] F. L. Faucheur et al., "MPLS Support of Differentiated Services", draft-ietf-mpls-diff-ext-09, Apr. 2001.
[11] S. Murphy, "The ns MPLS/DiffServ patch", http://www.teltec.dcu.ie/~murphys/ns-work/mpls-DiffServ/ (current 29 Apr 2001)
[12] G. Ahn, "MPLS Network Simulator", http://www.raonet.com/introduction.shtml (current 29 Apr 2001)
[13] D. Haskin, "A Method for Setting an Alternative Label Switched Paths to Handle Fast Reroute", draft-haskin-mpls-fast-reroute-05, May 2001.
[14] K. Nichols, "An Opinionated View of the Current State of IP DiffServ", ftp://ftp-eng.cisco.com/ftp/kmn-group/docs/kmn_ucbmm.pdf (current 29 Apr 2001).