Investigation and Comparison of MPLS QoS Solution and Differentiated Services QoS Solutions

Steve Gennaoui, Jianhua Yin, Samuel Swinton, and *Vasil Hnatyshin

Department of Computer Science
Rowan University
Glassboro, NJ 08028
*E-mail: {gennao83, yinj90, swinto22}@students.rowan.edu, *hnatyshin@rowan.edu*

Abstract
This paper examines the performance of the Differentiated Services and MPLS approaches for providing Quality of Service (QoS) guarantees in the network. The first set of scenarios had the FTP, voice, and video traffic sources mapped into various DiffServ classes and processed by the routers using different queuing disciplines, i.e., FIFO, priority queuing, DWRR, and WFQ. In the second set of scenarios we deployed Multiprotocol Label Switching (MPLS) and mapped traffic sources into different Label-Switched Paths (LSP). We also varied the link capacities in the network to create scenarios where the traffic flows have to contend with congestion. Simulation results collected using OPNET IT Guru 17.5 showed that in the case of congestion DiffServ is unable to provide QoS guarantees. MPLS, on the other hand, can route traffic over uncongested paths which help the flows achieve their desired levels of QoS.

1. Introduction

With the recent rapid increase in the number of network based applications, there have been numerous efforts to meet quality of service (QoS) demands from these applications without increasing the network capacity. Among the most prominent approaches for providing Quality of Service are Integrated Services [1-2] (IntServ) and Differentiated Services [3] (DiffServ). While each approach offers its own benefits, there are times when IntServ and DiffServ are insufficient to satisfy desired QoS requirements.

The Integrated Services architecture [1-2] provides fine-grained per-flow guarantees. To achieve this level of QoS, IntServ requires all the routers on the path traversed by a flow to reserve and manage available resources such as available queue space and outgoing link capacity. The Internet typically deals with billions of traffic flows, many of which may travel through the same core routers. Maintaining and managing resource reservations for all the flows that travel through the core routers creates enormous processing and storage overheads. That is why the Integrated Services architecture does not scale well to large networks such as the Internet and is deployed only on a small scale in private networks.

The Differentiated Services [1] architecture addresses the issue of scalability by supporting coarse-grained, per-class Quality of Service requirements. In the Differentiated Services architecture the flows with similar QoS requirements are combined into traffic aggregates or traffic classes. Each aggregate or class is identified by its differentiated services code point (DSCP). The DSCP value is recorded in the Type of Service (ToS) field of the packet's IP header and is typically set in the network edges, before the packet enters the network core. The Differentiated Service compliant core routers treat arriving packets based on the pre-configured per-hop behavior (PHB) which specifies how the packets that belong to a certain aggregate are to be treated (i.e., queued, forwarded, scheduled, etc). Unmarked packets that do not belong to any class are processed according to the default PHB specification. The Differentiated Services architecture provides a scalable solution to the QoS problem. However, the DiffServ-provided QoS guarantees are closely tied to network provisioning. If the path a traffic aggregate travels on does not have adequate resources, then the DiffServ approach won't be able to satisfy desired QoS requirements.

Multiprotocol Label Switching (MPLS) [4-5] is an approach for forwarding the data through the network based on the path label rather than the network address. Each label identifies a virtual link between the nodes and the forwarding decision is made based on the packet's label. By specifying a predefined path for the traffic flows to follow, MPLS allows for load-balancing and an effective traffic distribution in the network. When deployed together with DiffServ, MPLS can also provide QoS support: MPLS is responsible for traffic distribution on non-shortest paths in an effort to provide efficient utilization of network resources, while DiffServ provides service differentiation for traffic aggregates at the individual routers [5].

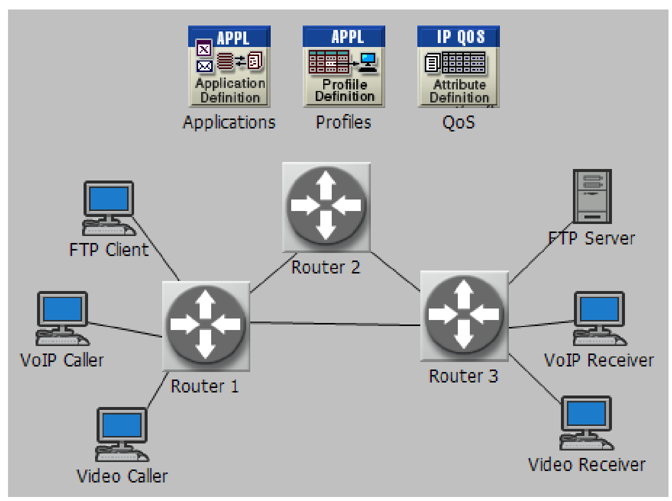

Figure 1: Network Topology

In this paper we examine the performance of various queuing mechanisms used together with the Differentiated Services and MPLS approaches for providing Quality of Service (QoS) guarantees. In our study we examined the performance of FTP, voice, and video applications when sending traffic through the network with the First-In-First-Out (FIFO), Priority Queuing (PQ), Deficit Weighted Round Robin (DWRR), and Weighted Fair Queuing (WFQ) queuing disciplines deployed at the router interface connected to the bottleneck link. We examined two scenarios one with MPLS disabled and another one with MPLS enabled. In the second scenario we deployed Multiprotocol

Label Switching (MPLS) and mapped traffic sources into different Label-Switched Paths (LSP). We varied the link capacities in the network to create scenarios where the traffic flows have to contend with congestion. Simulation results collected using OPNET IT Guru version 17.5 [6] showed that in the case of severe congestion, DiffServ is unable to provide QoS guarantees. MPLS, on the other hand, can route traffic over uncongested paths which help the flows achieve their desired levels of QoS.

The rest of the paper is organized as follows. Section 2 provides a summary of a study in which we examined the application performance in the Differentiated Services network with MPLS disabled. In Section 3 we examined application performance in the network with MPLS enabled and illustrated that MPLS can help the applications achieve their desired level of QoS in the scenarios where the Differentiated Services approach fails to do so. The paper concludes in Section 4.

2. Application Performance in the Differentiated Services Network with MPLS Disabled

2.1. Simulation Set-up

In our study we used the network topology shown in Figure 1, where the client nodes (i.e., *FTP Client*, *VoIP Caller*, and *Video Caller*) send the FTP, Voice, and Video traffic to their respective destinations (i.e., *FTP Server*, *VoIP Receiver*, and *Video Receiver*). In the DiffServ without MPLS scenario all the traffic travels on the shortest path through the Router1 – Router 3 link, which is configured to be the bottleneck. In the MPLS scenario the traffic flows can utilize an alternative path Router 1 – Router 2 – Router 3 which will allow them to better utilize network resources and achieve higher levels of QoS satisfaction.

Table 1 shows configuration of the FTP, Voice, and Video applications and their DSCP markings. We summarize the configuration of various DiffServ queuing disciplines in Table 2. All queuing mechanisms were configured as global QoS profiles and deployed on the interfaces attached to the bottleneck link between Router 1 and Router 2. To simplify analysis and comparison of collected results, we disabled RED and used constant traffic transmission rates.

Table 1: Application Configuration

Appl. Name	Configuration: Attribute	Configuration: Value
FTP	Command Mix (Get/Total):	0%
	Inter-request Time (seconds)	constant(2)
	File Size (bytes)	100,000
	Type of Service	**AF21**
Voice	Application Type:	IP Telephony
	Type of Service	**AF41**
Video	Application Type:	Low Resolution Video
	Type of Service	**EF**

We set the capacity of the links connecting the end nodes to their gateways (i.e., Router 1 and Router 2) to that of a DS3 line. We varied the capacity on the bottleneck link Router 1 – Router 2 by setting it to 1.0 Mbps, 1.5 Mbps, and 2.0 Mbps. Such configuration resulted in various levels of network congestion as the total traffic arrival rate exceeded the capacity of the bottleneck link.

Table 2: Configuration of Queuing Disciplines

Queuing Discipline	Configuration: Attribute	Configuration: Value
FIFO	Maximum Queue Size (pkts)	500
	RED parameters	Disabled
PQ	Priority Label Max Queue Size (pkts) Classification Scheme RED Parameters	1 (Normal) 200 ToS = AF21 Disabled
	Priority Label Max Queue Size (pkts) Classification Scheme RED Parameters	2 (Medium) 60 ToS = AF41 Disabled
	Priority Label Max Queue Size (pkts) Classification Scheme RED Parameters	3 (High) 40 ToS = EF Disabled
DWRR	Weight Max Queue Size (pkts) Classification Scheme RED Parameters	15 200 ToS = AF21 Disabled
	Weight Max Queue Size (pkts) Classification Scheme RED Parameters	30 60 ToS = AF41 Disabled
	Weight Max Queue Size (pkts) Classification Scheme RED Parameters	55 40 ToS = EF Disabled
WFQ	Buffer Capacity	300
	Weight Max Queue Size (pkts) Classification Scheme RED Parameters	15 200 ToS = AF21 Disabled
	Weight Max Queue Size (pkts) Classification Scheme RED Parameters	30 60 ToS = AF41 Disabled
	Weight Max Queue Size (pkts) Classification Scheme RED Parameters	55 40 ToS = EF Disabled

2.2. Analysis of Results

Figure 2 illustrates the total amount of traffic generated by individual applications in this study. Specifically, Video traffic was generated at the constant rate of 1.4 Mbps, VoIP traffic was generated at the constant rate of 45.6 Kbps, and FTP traffic was sent at the average rate of about 420 Kbps. These are typical transmission rate for these applications. In Figure 2, there are two lines for the FTP application traffic: one showing transmission rate of about 840 Kbps and another one showing rate of 0 Kbps, which represent the average transmission rate of about 420 Kbps. Figures 3 – 5 illustrate how various queuing techniques distribute available bandwidth on the bottleneck link Router 1 – Router 3 among individual application. Each figure contains four graphs, one for each queuing mechanism (i.e., WFQ, DWRR, PQ, and FIFO). Each graph contains three lines, each line representing the throughput for the examined application (i.e., Video, FTP, VoIP) at three different values of bottleneck link capacity. For example, the top left panel in Figure 3 illustrates the bandwidth allocated to the video traffic using Weighted Fair Queuing (WFQ) when the bottleneck link capacity was set to 1.0 Mbps, 1.5 Mbps, and 2.0 Mbps.

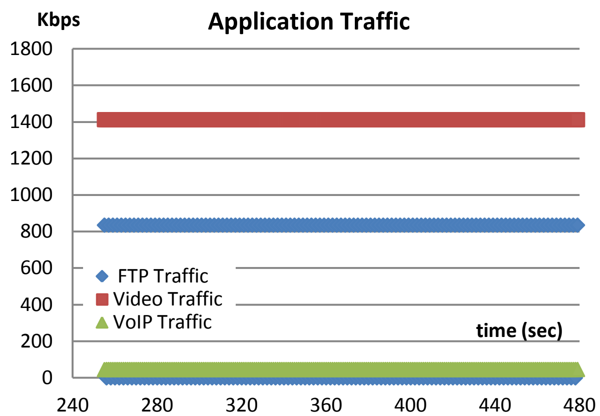

Figure 2: Application Traffic Generation Rate

In the scenarios where the bottleneck capacity is set to 2.0 Mbps there is no congestion and as a result all applications were able to receive the amount of bandwidth close to what was needed for their applications. However, when the bottleneck link capacity is reduced, the applications were unable to achieve the desired QoS levels.

The WFQ mechanism distributes available bandwidth among individual flows according to their weights, shown in Table 2. In the scenarios where the bottleneck capacity was set to 1.5 Mbps and to 1.0 Mbps neither Video nor FTP application was able to achieve the desired amount of bandwidth and as a result experience significant loss and delay. The voice application (also referred to as VoIP), on the other hand, performed reasonably well and was able to obtain the necessary amount of resources. This is primarily due to the VoIP application requiring significantly less bandwidth than its allocated WFQ share.

The achieved level of quality of service using DWRR is almost identical to that when using Weighted Fair Queuing. WFQ provides a fine-grained fair resource distribution on a per-bit basis. The Deficit Weighted Round Robin mechanism provides a more coarse resource distribution. DWRR relies on a deficit counter, which specifies the amount of data in bytes that can be serviced during each round. During each round the queue forwards the packet onto the outgoing interface as long as the value of the deficit counter is greater than the size of that packet. As a result, DWRR can service a different number of packets during each round, which leads to a bit more variability in achieved bandwidth than when using WFQ.

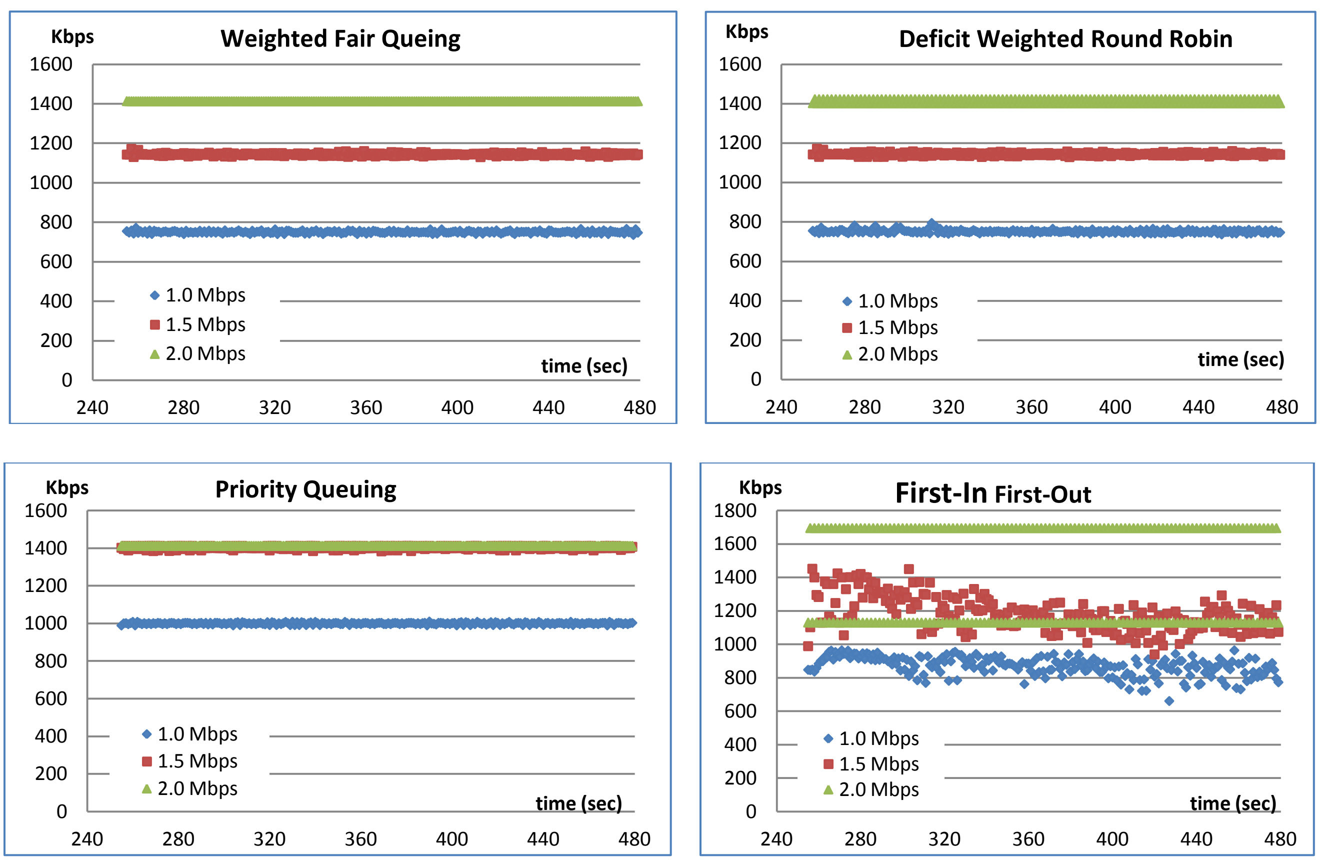

Figure 3: Video Traffic Distribution with different queuing techniques

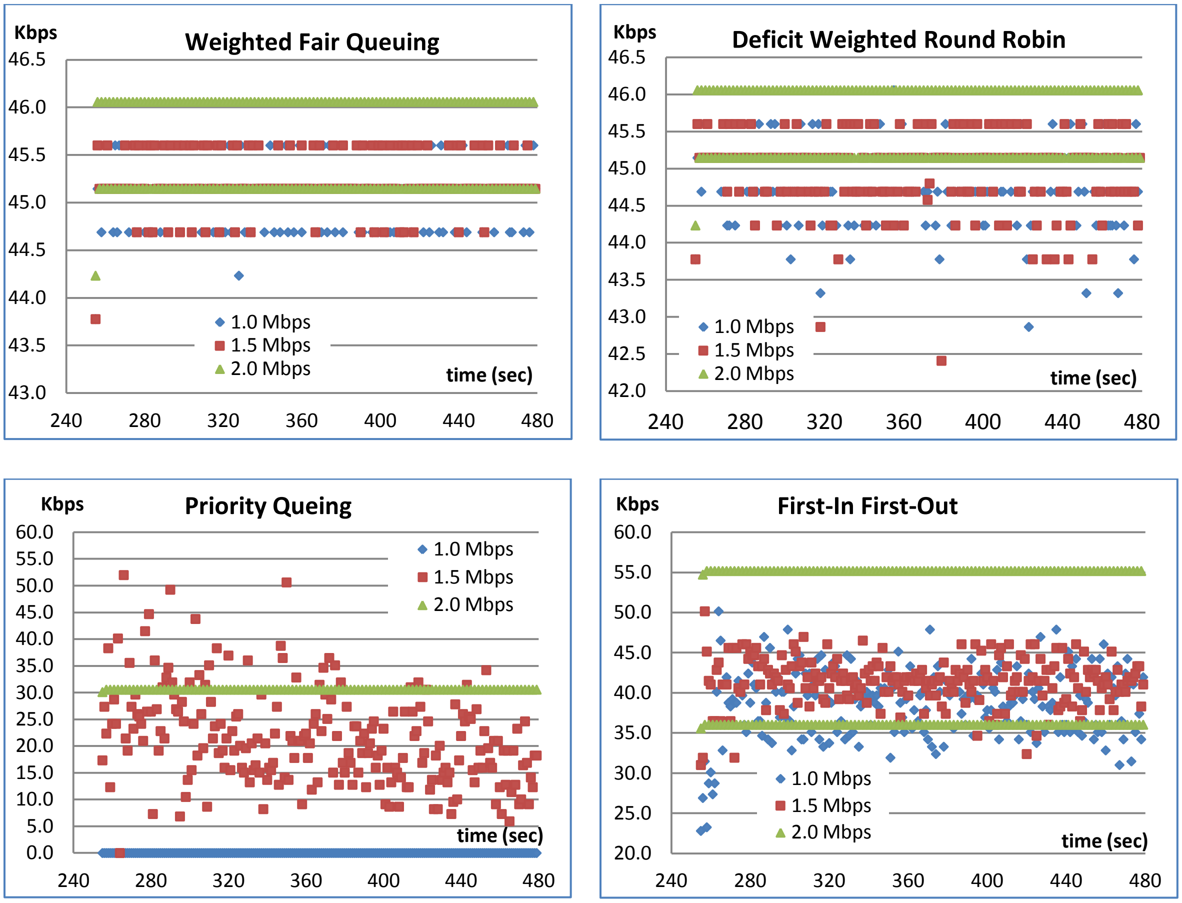

Figure 4: Voice Traffic Distribution with different queuing techniques

Figure 5: FTP Traffic Distribution with different queuing techniques

In scenarios where the priority queuing mechanism was deployed, the video traffic (highest priority) was allocated either the required 1.4 Mbps or entire available bandwidth on the link. When the bottleneck link was set to 1.5 Mbps the FTP application experienced severe performance degradation (unacceptable levels of loss), while the VoIP application experienced significant packet delay variation (the graph is not shown due to space limitations), which is also highly undesirable for voice traffic. Furthermore, when the bottleneck link capacity was set to 1.0 Mbps, both the FTP and VoIP applications were unable to deliver any data at all.

The FIFO queuing does not provide any service differentiation or QoS support. As result, when the FIFO queuing mechanism was deployed on the bottleneck link, the applications had to compete against one another directly. Since the video and VoIP applications run over UDP, they do not reduce their transmission rates when packets are lost. FTP, on the other hand, runs over TCP which throttles the traffic flows when congestion occurs (i.e., a packet loss has been detected). As a result, the video and VoIP applications, unfairly gain a larger share of available bandwidth while the FTP traffic has to be satisfied with the leftovers.

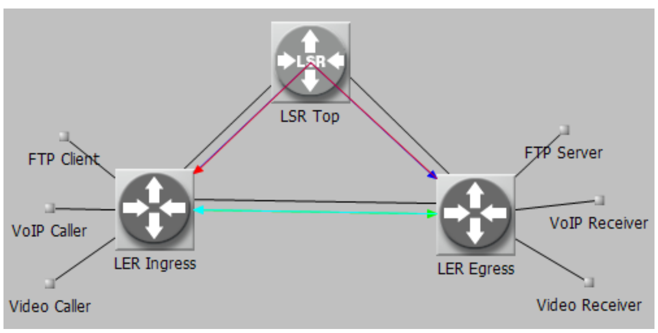

Figure 6: Network Topology for MPLS study

Overall, while some queuing mechanisms can provide better service differentiation then others, none of them are able to provide the desired levels of Quality of Service when the network is not properly provisioned (i.e., the links of the shortest path do not have enough capacity to carry the traffic). MPLS is an alternative and supplementary mechanism which allows the traffic to be routed over the non-shortest paths, utilizing the resources on the links that in traditional networks remain unused, which may lead to higher levels of QoS satisfaction.

3. Application Performance in the Network with MPLS Enabled

3.1. Simulation Set-up

To illustrate how MPLS influences the application performance, we deployed the same three applications defined in Table 1 into the network shown in Figure 1. To follow MPLS terminology we renamed the routers as LER Ingress, LSR Top, and LER Egress as shown in Figure 6, while the rest of the network topology remained unchanged. We also varied the capacity of the links between the MPLS routers differently than in the DiffServ study. Since in the MPLS scenario the traffic will follow different paths, we set the capacity of the links in the MPLS domain to 1.0 Mbps, 1.2 Mbps, and 1.5 Mbps. Such configuration ensured that while individual links in the network are unable the carry all of the application's traffic, if routed over different paths, the network will be able to provide the desired level of QoS to individual applications.

Table 3: FEC and Traffic Trunk Profiles

FEC	Traffic Trunk Profile
Name: FTP FEC DHCP: AF21 Protocol: TCP	Name: FTP Trunk Max Bit Rate: 850 Kbps Avg Bit Rate : 480 Kbps Peak Burst Size: 800 Kbps Max Burst Size : 800 Kbps Out of Profile: Discard Traffic Class: AF21
Name: VoIP FEC DHCP: AF41 Protocol: UDP	Name: VoIP Trunk Max Bit Rate: 64 Kbps Avg Bit Rate : 48 Kbps Peak Burst Size: 8 Kbps Max Burst Size : 8 Kbps Out of Profile: Discard Traffic Class: AF41
Name: Video FEC DHCP: EF Protocol: UDP	Name: Video Trunk Max Bit Rate: 1.5 Mbps Avg Bit Rate: 1.4 Mbps Peak Burst Size: 700 Kbps Max Burst Size: 700 Kbps Out of Profile: Discard Traffic Class: EF

In MPLS, the Label Edge Routers (LER) are responsible for labeling incoming packets based on available routing information before they are forwarded into the MPLS domain. The Label Switch Routers (LSR) are responsible for switching incoming packets based on their label and updating the label before the packet is forwarded to the next hop. The MPLS Label Switch Paths (LSPs) are the paths through the MPLS network. LSPs are set-up based on the requirements in the Forwarding Equivalence Classes (FEC) that the traffic flows are mapped into. In addition to matching class marking such as DSCP or ToS byte, the traffic flows mapped into an FEC must also satisfy its traffic trunk profile, typically used for Traffic Engineering. Table 3 illustrates the summary of FEC and Traffic Trunk Profile configuration specified in IT Guru via the *mpls_config_object*.

To deploy MPLS into a network, first we defined four LSPs (model *MPLS_E-LSP_DYNAMIC*) as shown in Table 4. Next, we configured LER Ingress and LER Egress routers to map incoming traffic flows into their corresponding FEC, traffic trunk profiles, and LSPs. We set up LER routers to forward all the FTP and VoIP traffic over the longer path: LER Ingress – LSR Top – LER Egress, while the video traffic was forwarded over two paths. Specifically, 65% of the video traffic was sent over the path LER Ingress – LER Egress, while remaining 35% of the video traffic was sent over the LER Ingress – LSR Top – LER Egress path. Summary of LER configuration is shown in Table 5.

Table 4: LSP Definitions

Name	Path
Ingress -Top - Egress	LER Ingress -> LSR Top -> LER Egress
Egress -Top - Ingress	LER Egress -> LSR Top -> LER Ingress
Ingress –Egress	LER Ingress -> LER Egress
Egress – Ingress	LER Egress -> LER Ingress

Finally, we configured LSR router to define the mapping between FECs and the traffic trunk profiles. Specifically, traffic flows that belong to FTP FEC, VoIP FEC, and Video FEC were mapped into FTP Traffic Trunk, VoIP Traffic Trunk, and Video Traffic Trunk profiles, respectively.

Table 5: LER Configuration

	Application	MPLS Traffic Mapping Configuration	
LER Ingress	FTP	Interface: FEC: Traffic Trunk: Primary LSP: LSP Weight:	4 `FTP FEC FTP Trunk Ingress -Top- Egress 100%
	VoIP	Interface: FEC: Traffic Trunk: Primary LSP: LSP Weight:	3 VoIP FEC VoIP Trunk Ingress -Top- Egress 100%
	Video	Interface: FEC: Traffic Trunk: Primary LSP: LSP Weight: Primary LSP: LSP Weight:	2 Video FEC Video Trunk Ingress - Egress 65% Ingress -Top- Egress 35%
LER Egress	FTP	Interface: FEC: Traffic Trunk: Primary LSP: LSP Weight:	2 FTP FEC FTP Trunk Egress-Top-Ingress 100%
	VoIP	Interface: FEC: Traffic Trunk: Primary LSP: LSP Weight:	3 VoIP FEC VoIP Trunk Egress-Top-Ingress 100%
	Video	Interface: FEC: Traffic Trunk: Primary LSP: LSP Weight: Primary LSP: LSP Weight:	4 Video FEC Video Trunk Egress-Ingress 65% Egress-Top-Ingress 35%

3.2. Analysis of Results

In our study of the application performance in the MPLS-enabled network, we set the capacity of MPLS domain links to 1.0 Mbps, 1.2 Mbps, and 1.5 Mbps. Such provisioning in the DiffServ network with MPLS-disabled resulted in severe congestion and the application failing to achieve the desired levels of QoS as discussed in Section 2. Our study showed that such capacity allocation in an MPLS-enabled network is sufficient to satisfy bandwidth requirements of all applications.

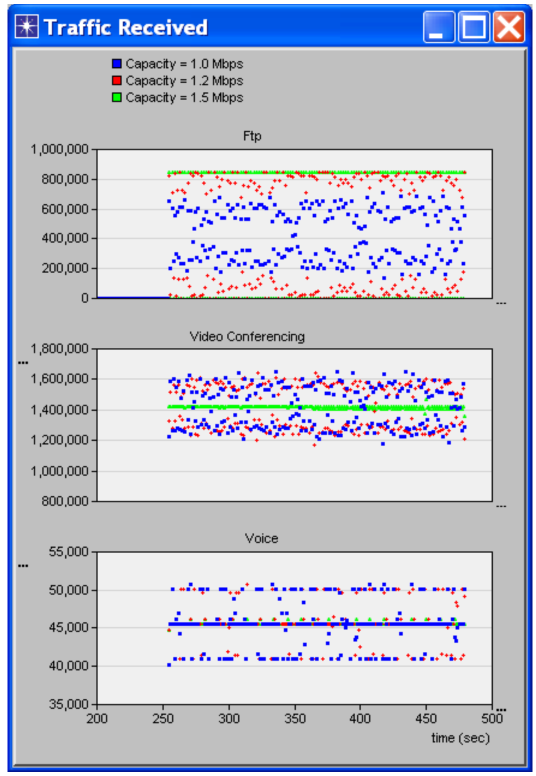

Figure 6: Throughput in MPLS study

As shown in Figure 6, all applications were able to achieve their desired amount of bandwidth. The main difference in the application performance was the delay. Figure 7 illustrates the average delay experiences by the applications in the MPLS-enabled network. While the end-to-end delay varied from application to application, it was always within the range of acceptable values. MPLS is able to provide better QoS support because it routes traffic over less utilized paths that may not necessarily be the shortest, while DiffServ is not capable of such load-balancing since it relies on shortest path routing.

The application performance in an MPLS-enabled network can be improved even more by deploying queuing mechanisms on the LER routers. We modified the MPLS scenario and deployed WFQ on the interfaces that connect the LER Ingress and LER Egress routers to the LSR Top router. These are the only links that carry a mixture of FTP, video, and voice traffic and thus can benefit from a more sophisticated queuing discipline than the default FIFO queues. The LER Ingress – LER Egress path only carries video traffic and thus does not require any mechanism for traffic differentiation.

The WFQ configuration was similar to that used in the DiffServ scenario summarized in Table 2. It should be noted that traffic distribution in the MPLS scenario is different from that in the DiffServ scenario. Specifically, in the MPLS scenario only 35% of video traffic is traveling through the bottleneck link, which now is located between the LER Ingress and LSR Top routers. That is why we allocated different WFQ weights to traffic classes. Specifically, FTP traffic weight was set to 70, video traffic weight was set to 30, and voice traffic was sent into a Low Latency Queue, which operates similar to priority queuing, i.e., the traffic in the Low Latency Queue is processed ahead of all the other traffic. The traffic in the other queues is processed only when the Low Latency Queue is empty.

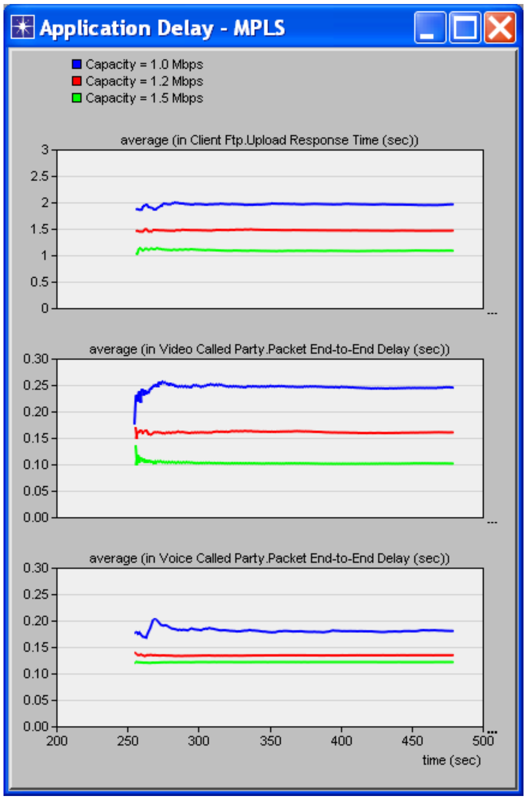

Figure 7: Delay in MPLS study

The results of the application performance in the DiffServ network with MPLS enabled are shown in Figures 8 and 9. Adding WFQ with Low Latency Queue reduced the end-to-end delay experienced by the video and voice applications. This improvement resulted in an increase in the FTP application's loss and delay, when the link capacity was set to 1.0 and 1.2 Mbps.

4. Conclusions

This paper compares the application performance achieved using various queuing mechanisms in the context of DiffServ architecture against the performance achieved in the network with MPLS. The simulation study conducted using OPNET IT Guru ver. 17.5 software package [6] showed that while the Differentiated Services architecture can provide a certain level of QoS assurance, if the links on the path taken by the traffic flows are not properly provisioned then the applications will be unable to achieve the desired level of QoS. MPLS, on the other hand, is more flexible and can route the traffic over alternative non-shortest paths which contain sufficient amount of resources to satisfy QoS requirements. The network configuration can be further refined by combining MPLS and DiffServ approaches based on the QoS requirements which may lead to even better application performance.

References

[1] R. Braden , D. Clark and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," IETF RFC 1633, June 1994, http://www.ietf.org/rfc/rfc1633.txt

[2] P.P. White, "RSVP and integrated services in the Internet: a tutorial," IEEE Communications Magazine, Volume 35, Issue 5, pp. 100-106, May 1997, DOI: 10.1109/35.592102

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, "An Architecture for Differentiated Services, " IETF RFC 2475, December 1998, http://www.ietf.org/rfc/rfc2475.txt

[4] E. Mannie, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture," IETF RFC 6002, October 2004, http://www.ietf.org/rfc/rfc3945.txt

[5] B. Davie, A, Farrel, "MPLS: Next Steps," Morgan Kaufmann series in networking, 2008, ISBN-13: 978-0-12-374400-5

[6] OPNET IT Guru 17.5, Riverbed Technology, Inc., 2013

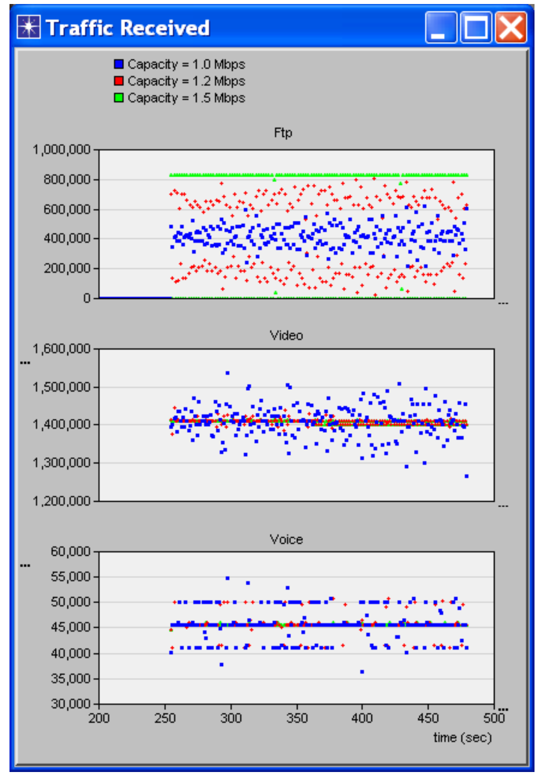

Figure 8: Throughput in MPLS with DiffServ study

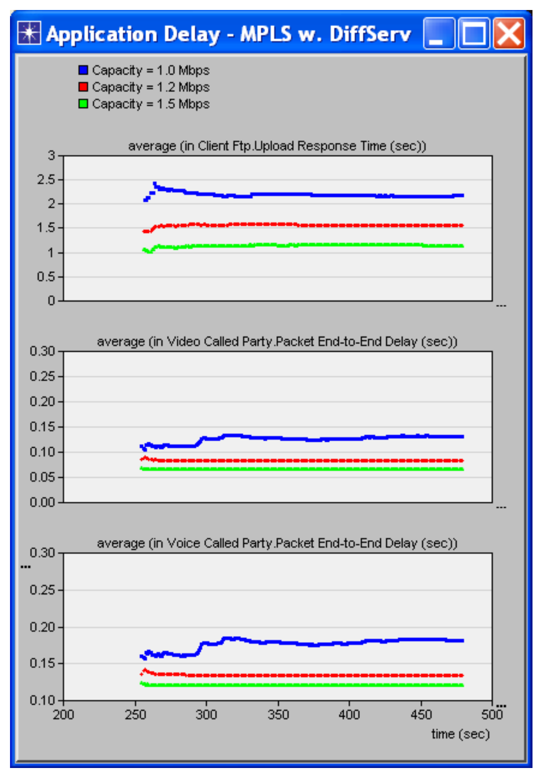

Figure 9: Delay in MPLS with DiffServ study