MPLS: Managing the New Internet

Junaid Ahmed Zubairi
Department of Math and Computer Science
SUNY at Fredonia, Fredonia NY 14063-2330, USA
zubairi@cs.fredonia.edu
and
Wajdi Al-Khateeb ,
Electrical and Computer Engineering Department, Kulliyyah of Engineering
International Islamic University Malaysia, Kuala Lumpur, Malaysia
wajdi@iiu.edu.my

ABSTRACT

Traffic requirements on the Internet have changed from reliability to timeliness of delivery as different types of applications are being deployed on it. This situation has prompted the industry and IETF (Internet Engineering Task Force) to develop new protocols and techniques to meet the challenges. One of the important protocols developed is MPLS (Multi Protocol Label Switching) that allows network operators to implement traffic engineering, an important aspect of network operation that was neglected earlier. In this paper, we discuss why traffic engineering is a necessary tool in next generation Internet and discuss several examples in which MPLS traffic engineering implementation results in enhanced network performance. We also look at the recent development of GMPLS (Generalized MPLS) that provides the control plane for lambda switching, waveband switching and fiber switching.

1 INTRODUCTION

Internet was started primarily as a research network and its TCP/IP suite supported the reliable transport of data over unreliable network. The fault tolerant protocols made sure that the transmission of data was completed even if some nodes crashed in the middle. In the early 1990's, the deployment of HTTP in the Internet and HTML-capable GUI browsers on user workstations transformed the Internet into a global information network. This transformation accelerated the growth of Internet and expanded the user base in a major way.

The transformation of Internet into a general information network also caused the introduction of different Internet enabled applications. Some of the applications such as file-transfer and e-mail still expected reliability in communication. However, an increasing number of applications demanded timeliness in delivery of data. For example, the e-mail application can wait for a random amount of time for delivery of messages. However, a telemedicine application or a database update transaction must be finished within a bounded time period. The TCP/IP suite was not ready for such time-sensitive services as it had been developed with the target of assured delivery of data. The network part in TCP/IP suite makes its best effort to deliver the data in a reliable and timely way. However, if the data is delayed or discarded, the network cannot alleviate this problem and the upper layer (i.e. TCP) has to take the corrective action. This was later labeled as BE (best effort) behavior of the network.

The BE behavior of the Internet results directly from the way the routers function. Please refer to Figure 1 that depicts a router with its internal building blocks. It shows that a router has multiple input ports and a switching fabric that connects input ports to output ports. Each port has a queue associated with it and the input ports also have access to the routing table. The next hop for a packet is decided based on the longest-prefix-match rule consulting the local forwarding table associated with the input port that has received the packet. If the processing rate for an input port is less than the rate of receiving packets, the queue associated with that input port fills up, causing delay variations for the queued packets. Additional delays can result from the aggregation of packets on a single output port when the routing algorithm prefers one outgoing link over the others. If this trend continues, the queue associated with the preferred output port fills up, causing packet loss and additional variation in delays for the queued packets. With traditional link-state or distance-vector routing, some packets may get delayed more than others depending on the queuing delays experienced. Packets belonging to time-sensitive traffic are required to reach the destination within specified time bounds otherwise the performance may be degraded or the delayed packets may become totally useless. As an example, voice call is transmitted in real-time and the delayed traffic loses its utility because the words spoken by a person have to be played back in exact sequence. A delay of maximum 0.4 seconds is tolerable during the conversation. Any delay

beyond 0.4 seconds makes the speech unintelligible. Best effort Internet is unable to provide any guarantees of delivering the data within 0.4 seconds or delivering all of the data. Internet routers may also reduce or increase the delays between talk spurts as other traffic mingles with the voice packets in the network core. This variation of delay between talk spurts may reduce the effectiveness of speech when it is played back at the receiver end.

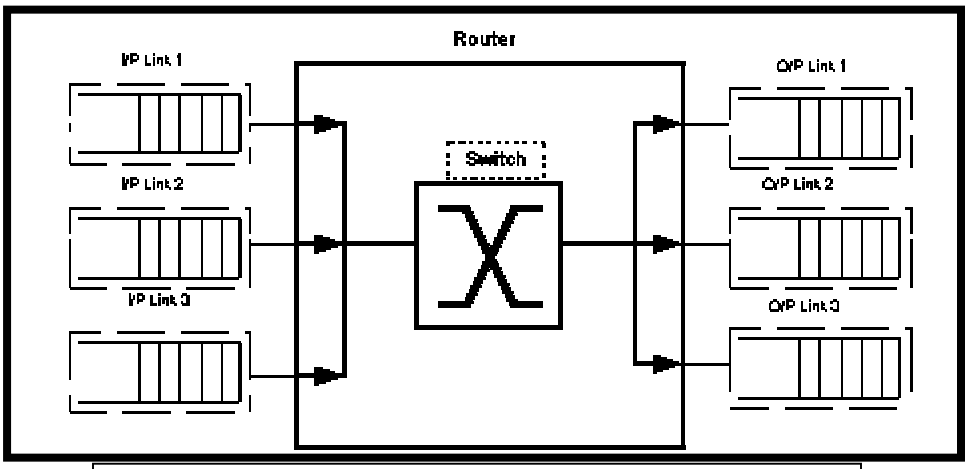

Figure 1: The Internal Configuration of a Router

New techniques and protocols are being developed to meet the demands of the newer applications on the Internet. These protocols are expected to cater to the time-sensitive applications while keeping the network fair and efficient for all. IETF (Internet Engineering Task Force) has been working on developing protocols for supporting time-sensitive service in the Internet. Among the new protocols, RSVP (Resource Reservation Protocol) [1,2,3,4] provides quantitative guarantees per flow via reservations whereas Diffserv (Differentiated Services) [5,6,7,8] provides qualitative assurances by using PHB (per hop behavior) for packets marked with DSCP (DS codepoints) in the IP headers. MPLS has been developed to accelerate routing of traffic aggregates destined towards a common point by using Label Switched Paths (LSP). Let us look at the above-mentioned protocols briefly in order to get the complete picture.

WHAT IS RSVP?

RSVP (Resource Reservation Protocol) is a receiver-based advance reservation protocol. RSVP attempts to eliminate surprise overloads by requiring reservations before any transmission can be done. Airlines and other long-haul transportation companies that require reservations in advance of travel have used this concept successfully. However, RSVP suffers from the problem of poor scalability because it requires the routers to maintain knowledge of every microflow that passes through them. The number of microflows passing through the core of the Internet can easily reach into tens of thousands. Tracking every microflow can overwhelm the backbone routers and the performance may degrade instead of improving. Recently, efforts have been made to introduce aggregation of microflows in RSVP to improve its scalability.

WHAT IS DIFFSERV?

Diffserv is a mechanism developed by IETF in order to let a router choose the most suitable service for each of the different traffic streams passing through it. As an example of Diffserv, consider filling the gas tank of your car. If you go to a gasoline station, you can choose the type of gas and pay accordingly. High-octane gas is expensive but it optimizes the performance of your car. In a similar way, when you connect to the Internet, with Diffserv you would be able to choose the type of service desired and you will be charged accordingly. EF (Expedited Forwarding) is the most expensive "premium" service available under Diffserv. Other services include AF (Assured Forwarding) and DF (traditional Best Effort or Default Forwarding). An analogy can be established with the postal system's various services. An ordinary letter is the like the traditional BE service, a registered letter can be considered to receive AF service whereas the letter dispatched using overnight courier enjoys EF service. Diffserv enabled routers use class based

queuing to offer differentiated services to Diffserv coded traffic. For example, a Diffserv enabled router expedites the EF (Expedited Forwarding) labeled packets and discards DF (Default Forwarding) packets in case of congestion.

WHAT IS MPLS?

MPLS manages a network domain by introducing connection oriented LSP's (Label Switched Paths) in a connectionless network. Traditional Internet routers are connectionless devices. When a packet arrives at an intermediate router, the router selects the best route for the packet and then forwards it on the link that falls on the selected route. In MPLS, the route for a flow is decided in advance of transmitting it. There are a number of benefits in deciding the path of a flow in advance. The intermediate routers can forward the traffic on the fly without any processing delay. In MPLS, the paths can be mapped on ATM (Asynchronous Transfer Mode) network easily because ATM also selects paths before start of transmission. The network administrators can manually lay down paths through the MPLS domain based on some policy. The LSP's are allocated to requesting flows that may enter the domain via an ingress node and exit via an egress node. An ingress node is the entry door and an egress node is the exit door of the domain. Once an LSP (Label Switched Path) is installed, it is monitored and terminated when the flow transmission has been completed. All packets within this LSP are treated in a similar manner. One of the most important capabilities of MPLS is to map traffic on paths established with traffic engineering principles, resulting in load balancing and fault tolerance of the underlying network.

We examine the traffic engineering principles and their potential uses in MPLS in detail in later sections. Rest of the paper is organized in four sections. In the next section, we introduce the concept of traffic engineering and discuss its capabilities and principles. In section 3, we discuss MPLS and its signaling and label distribution protocols. The traffic engineering principles as applied in MPLS are also discussed. In section 4, we explain the application of traffic engineering principles in some MPLS domains and show the resulting benefits. Towards the end, we discuss the GMPLS that is evolving as a unifying protocol for diverse switching technologies.

2 TRAFFIC ENGINEERING

Traffic engineering is the name of mapping traffic flows onto the physical topology to enhance overall network utilization and create a uniform distribution of traffic throughout the network [9,10,11,12]. Traffic engineering optimizes network efficiency through mapping and distribution of traffic. The objectives of traffic engineering include satisfactory service delivery, maximum resource efficiency and avoidance of congestion on any single path. Traffic engineering also uses methods to control a network's response to traffic demands and other stimuli, such as link or node failures [13].

Traffic engineering in the traditional Internet was achieved by manipulating routing metrics, such as monetary cost, hop-count, bandwidth, reliability and delay. Control based on above metrics was considered adequate as long as Internet backbones were much smaller in terms of the number of routers, links, and amount of traffic. Since IGP (Interior Gateway Protocol) route calculation was topology driven and based on a simple additive metric such as the hop count or an administrative value, it did not consider other important dynamic criteria such as bandwidth availability or prevailing traffic characteristics. As a result, traffic was unevenly distributed across the network causing inefficient use of resources. As the Internet and its sub-domains expanded, it became increasingly difficult to ensure that a metric adjustment in one part of a huge network did not create a new problem in another part of the network. If congestion occurred, selected links were over-sized to resolve the problem. Congestion occurrence was a result of lack of network resources and uneven distribution of traffic leaving some parts of the network heavily overloaded and other parts only lightly loaded [10, 14, 15].

The problem of lack of network resources is usually addressed by providing more resources such as over-provisioning or upgrading the infrastructure. Uneven distribution of traffic is more complicated since it can be the product of the dynamic routing protocols such as OSPF and IS-IS, that select the shortest paths to forward packets. Hence a solution was required that takes into account more factors than the common path-metrics. While using shortest path conserves network resources, it may cause some other problems, such as congestion on some paths and under utilization of other paths [9,14].

TE (Traffic Engineering) includes methods for congestion avoidance, utilization improvement, fairness, reliability and QoS support. It also provides for evaluation of network performance and compliance to TE rules. Results from the evaluation can be used to improve the network topology and structure [10,13,14,15].

3 MPLS AND CONSTRAINED ROUTING

MPLS identifies an end-to-end path before starting to transmit the data . It combines the L3 routing and L2 switching into "L2.5 forwarding" and provides a way to define connections in a connectionless network. MPLS attaches its own header (known as the “shim” header) between the headers of layer 2 and layer 3 protocols. It works on the principle of providing a "virtual path" from an ingress node to an egress node in an MPLS domain. This idea of providing a virtual path is not new and it has been used in the Internet to provide VPN tunnels across the public network and IPv6 tunnels across IPv4 networks. In tunneling, the packets originating at a router and destined for a specific node are labeled in such a way that the intermediate routers forward them towards a common destination through the same path. Thus, tunneling implicitly introduces the notion of a connection because all packets in the tunnel follow the same path and experience the same routing through the network.

In MPLS, the connection between the source and destination router can be defined in a variety of ways. It can be defined specifically, as in ER-LSP (Explicitly routed label switched path) or it can be loosely defined as a path that passes through several nodes some of which are specified while other nodes may be chosen later. In case of ER-LSP, it may be called an LSP tunnel [16]. For defining the LSP, constrained routing with traffic engineering principles is used in MPLS.

Constrained routing avoids congestion and uneven network utilization by optimized arrangement of traffic flows through the network. Routes that are subject to constraints such as bandwidth, delay, jitters, and administrative policy are computed considering the dynamic traffic load conditions in addition to the common metrics [9, 10, 14, 15]. It may prefer a longer but lightly loaded path over the heavily loaded shortest path. Given the QoS request of a flow or an aggregation of flows, QoS-routing returns the route that is most likely able to meet the QoS requirements. Constrained routing extends the conventional routing by defining extensions to the IGP (IS-IS and OSPF). These extensions may include maximum link bandwidth, maximum reservable link bandwidth, current bandwidth reservation and current bandwidth usage.

When a path creation request arrives, the MPLS performs *constrained routing* in order to find a suitable label switched path (LSP). Constrained routing applies the extended IGP parameters reported in link state advertisements to the overall tree to find a suitable path. Link state advertisements include reservable bandwidth, static link colors indicating capacities of links and TE-specific metrics [9]. Constrained Shortest Path First (CSPF) routing looks for a LSP that will meet the given constrains. The information of the constraints is used to prune (exclude) links from the LSP database that do not meet the specified constraints. Dijkstra's algorithm is then applied to the remaining graph to produce constrained shortest path between the ingress and the egress. This path must be recalculated whenever the constraints are changed. Path selection can follow a narrowing down of available choices by using Boolean operators. Computing optimal routes subject to two or more constraints is a NP-complete problem . Mostly, the algorithms work on "bandwidth available" and "hop count" for selecting a path between a source and destination. A constrained routing scheme can choose one of the followings as the route for a destination:

1. Shortest path, if multiple found select the widest one (the one with most available bandwidth)
2. Widest path, if multiple found, select shortest one
3. The shortest-distance path. Here the bandwidth is replaced by its inverse value that makes it possible to express the total distance of a k-hop path p as the sum of this inverse value:

$$\text{dist}(p) = \Sigma\,(1/n_i) \text{ where } n_i \text{ is the bandwidth of the link } i \text{ and } i=1 \rightarrow k$$

The last approach favors shortest paths when network load is heavy and widest paths when network load is moderate. However, this scheme does not differentiate between various classes of traffic as its only measure of the cost is the available bandwidth. In general, it is preferred over the first and second methods [14].

In the example shown in Figure 2, the desired LSP is to be installed between R4 and R11. In IGP-based computation of paths, all links are considered in the shortest path calculation, including links R5-R6, R4-R7, R4-R11 and R8-R9.

These links are excluded from the possible paths in the CSPF because they do not meet the constraints. The CSPF algorithm prepares a database of links that meet the constraints. Please refer to the inset in Figure 2 to see the set TE_LSDB. Dijkstra's algorithm can now be invoked on TE-LSDB to compute the shortest path.

Once an LSP has been selected, it is installed in the MPLS domain by signaling protocols such as RSVP-TE(Traffic Engineering extensions to RSVP) or CR-LDP(Constrained routing Label Distribution Protocol). An ER-LSP (Explicitly routed label switched path) can be established by sending a setup message with CR-LDP or RSVP-TE that can pin down the path through the routers. It is also possible that the setup message may specify only two nodes and the intermediate nodes can be set as per availability etc. Each LSP has a number of parameters that include its resilience, fault tolerance, pre-emptivity and associated QoS features.

A network may face nodal or link failures during the course of its operation. Availability of nodes and links in a network can be enhanced with the detection and handling of failures in a timely manner [17, 18]. MPLS provides for LSP failure detection with techniques built in the protocol. High availability and minimum system downtime in software are achieved by software fault-tolerance and the use of online software upgrades as well as suitably designed software implementation. Fault tolerance of hardware relies on hardware fault detection, reporting of failures, and on the availability of hardware backup. LDP HELLO and KEEPALIVE messages are used to validate the LSRs and links. Rerouting is the process of providing a new route for an LSP after notification of a failure or a topology change. Pre-programming of alternate paths for an LSP is known as LSP protection[17]. Failure at some point of a strictly specified explicitly routed LSP must be reported to the ingress. The ingress will establish a new LSP that can exclude the faulty links.

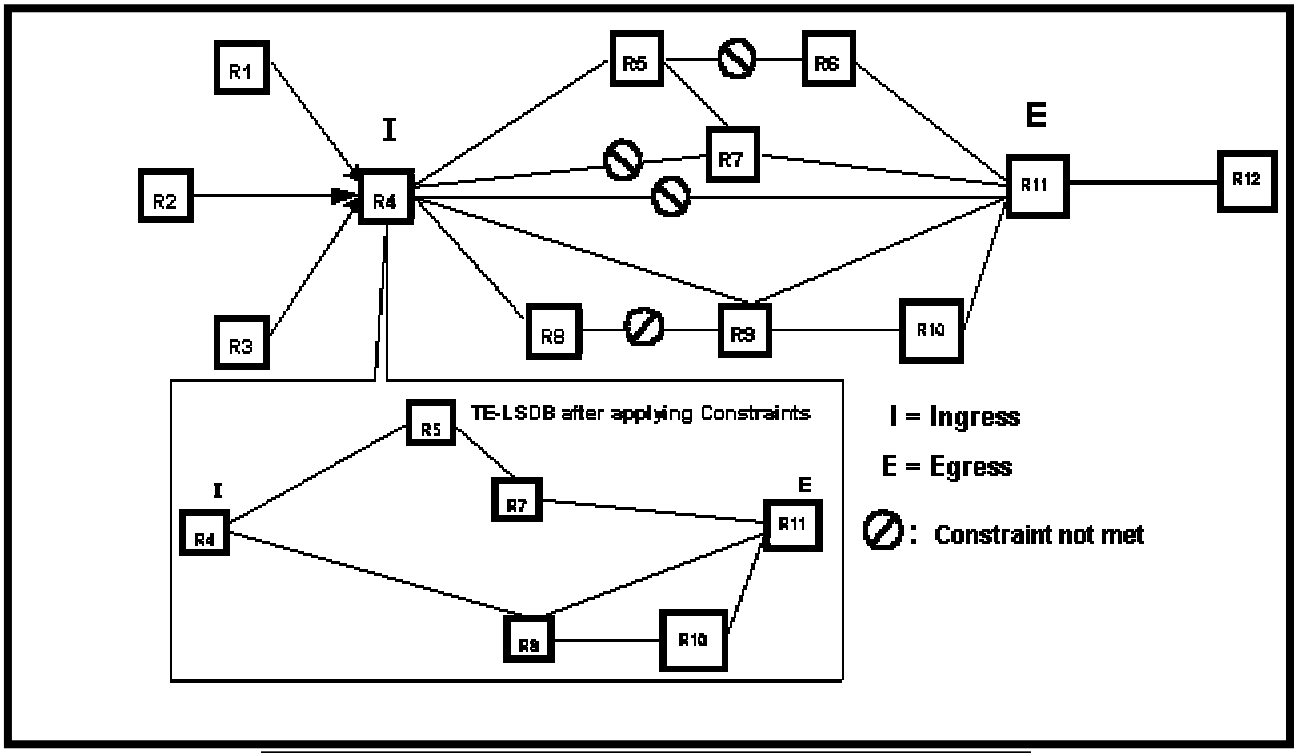

Figure 2: Constraint-Based Routing Using CSPF

4 EXAMPLES OF TRAFFIC ENGINEERING

In this section, we discuss some examples that demonstrate the application of traffic engineering principles in various MPLS domains. Figure 3 shows route selection based on metrics. In this example, the respective metrics m3 and m1 identify links of OC3 and OC12. In this example no single route among the three possible ones can accommodate the

traffic transmitted from the three sources R1, R2, and R3, each of which is sending at a rate of 100Mbps. However, using conventional shortest path routing, all three traffic streams are allocated the shortest path, that is R4-R5-R10, leading to congestion on the selected route while the other two routes R4-R6-R9-R10 and R4-R7-R8-R9-R10 remain unutilized.

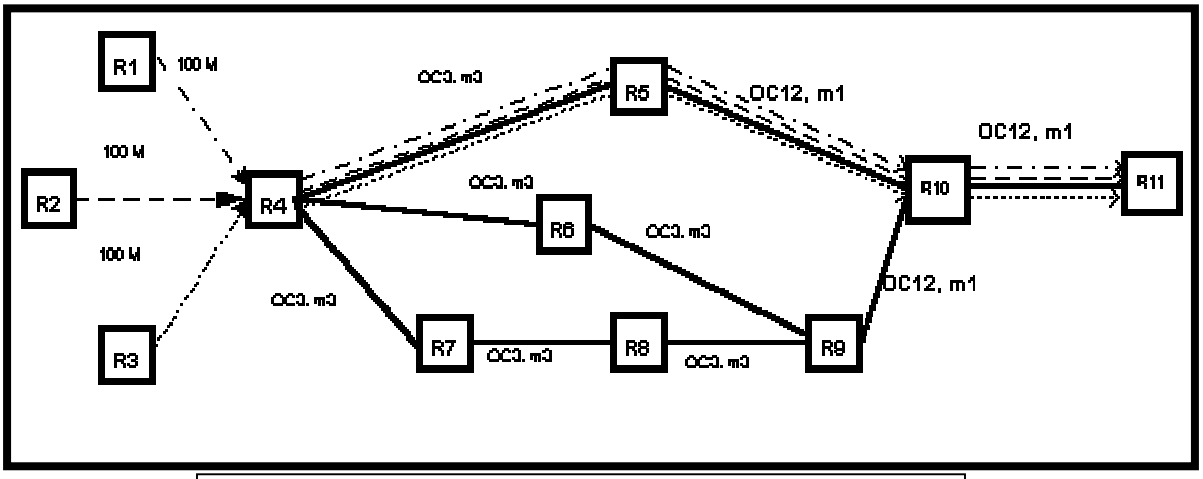

Figure 3: Inefficient Use of Resources in Traditional Routing

Traffic engineering's contribution to solve this dilemma is that it does not necessarily select the shortest path between two devices, moreover it is even possible that packets of two data flows may traverse completely different paths even though their originating node and the final destination node are the same. [19]

Figure 4 shows how traffic engineering approach will solve the above problem. To start with, the traffic from R1 will be placed on the path R4-R5-R10. By utilizing constraint-based routing, the traffic arriving from R2 is placed on the longer path R4-R6-R9-R10 and the traffic from R3 has to be placed on even longer route R4-R7-R8-R9-R10. Hence the routing selection is primarily done on the basis of the constraints on the individual links.

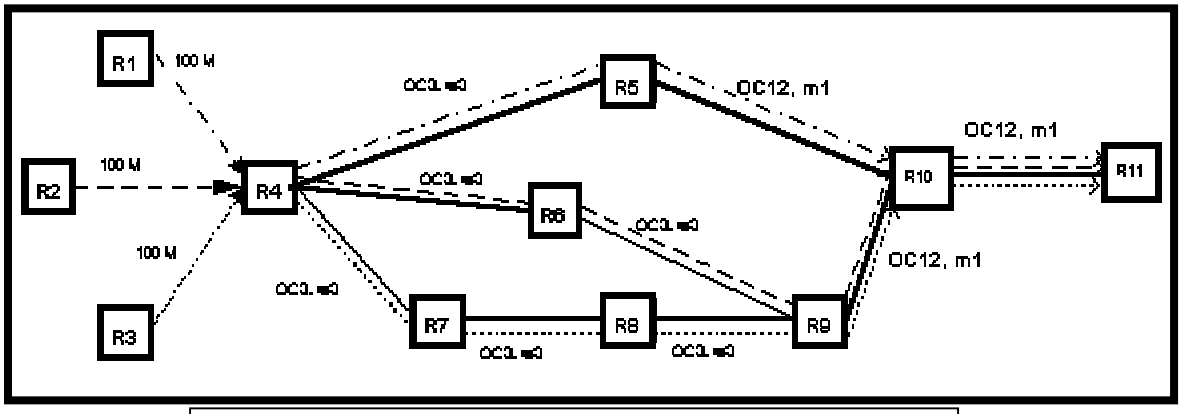

Figure 4: Congestion Avoidance Using TE Resource Utilization

Consider another scenario as depicted in Figure 5. If the traffic from any of the routers R1, R2 and R3 exceeds the capacity of any single path from R4 to R10, then multiple LSPs need to be configured between R4 and R10. Load ratio of these LSPs can be specified as desired, so that load can be distributed optimally. By assuming that R1 is already delivering CBR traffic at a rate of 500 Mbps to R11 on LSP R4-R5-R10, with links R4-R5 and R5-R10 both having

equal OC12 capacities, 500 M will be reserved on these two links. When R2 delivers new traffic of 200Mbps data rate, none of the three routes between R4 and R10 is capable of satisfying the new requirement placed by R2. Hence it is possible to share the new load in an appropriate ratio between R4-R5-R10 (122 M reservable capacity), R4-R6-R9-R10 (155 M reservable capacity), and R4-R7-R8-R9-R10 (155M reservable capacity). Figure 5 shows load sharing for the traffic from R2 among R4-R5-R10 and R4-R6-R9-R10. Similarly the traffic from R3 will split among R4-R6-R9-R10 and R4-R7-R8-R9-R10.

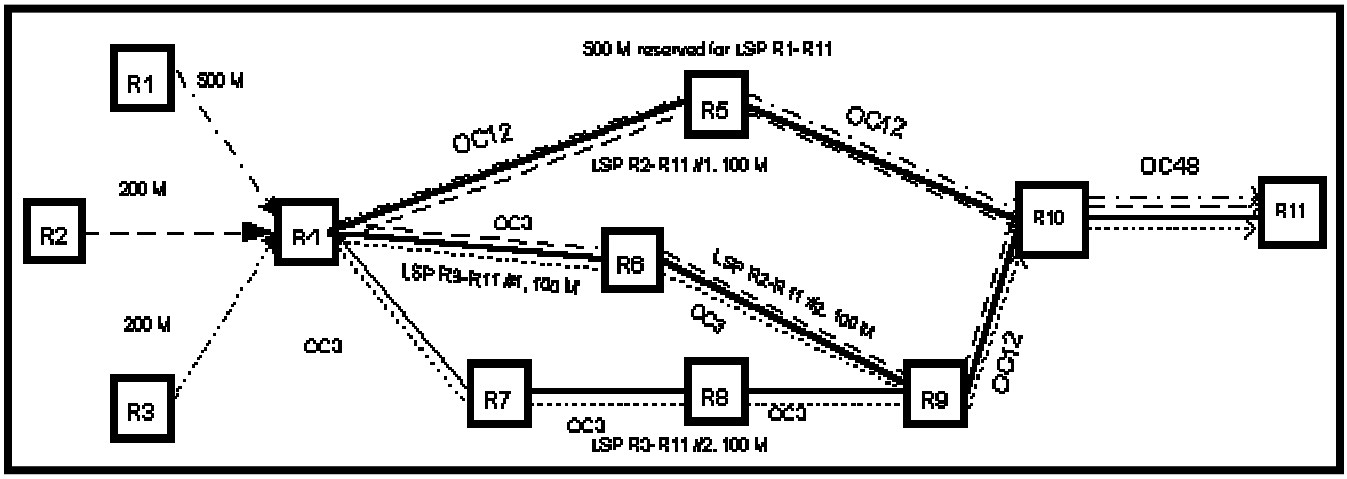

Figure 5: Multiple Route Load Sharing

In the optimum condition the load ratio is automatically derived from their bandwidth specification. Here we see that with a constraint-based routing, not only load sharing can be done among multiple paths of different cost, but also load ratio can be specified as desired. This will solve the problem mentioned earlier, i.e. the traffic from a source to a destination exceeds the capacity of the shortest path, while a longer path between these two routers is under-utilized.

5 GENERALIZED MPLS (GMPLS): THE LAMBDA FACTOR

Traffic on the Internet is increasing at a much faster rate than the provisioned capacities. Under this situation, the backbone of the Internet must be converted to all optical communication otherwise it will not be able to support the offered load that will soon run into several Terabits per second. The current Internet slows down when the photons are converted to electrons so that the traffic can be routed by the electronic modules in backbone routers. Experiments are underway to install all-optical modules in the routers so that the traffic can be routed while it is still in the form of light. The recent development of an artificial crystal (KLTN) [22] is an important step in this direction. Since the forwarding decision in optical devices is based on wavelengths instead of cell headers, it is known as lambda switching. KLTN will be able to deflect (or route) one out of hundreds of wavelengths present in a beam within a fiber to a specific direction. There will be no moving parts involved and no conversion to electrons will be needed. Therefore, it is expected that the switching will be 10,000 times faster than the current rate.

MPLS architecture as described above relies on labels carried within the "shim" headers to direct the traffic. Since the optical modules forward the data based on wavelength, MPLS does not fare well in optical communication. The earlier efforts to incorporate wavelength based switching resulted in a proposed name change from MPLS to MPλS. It was soon realized that a general protocol is needed to address the needs of devices that do not deal with individual packets. Later, the IETF released a draft [23] that suggested extending MPLS to GMPLS or Generalized MPLS. The target of GMPLS is to provide a generalized control plane that can be used to perform switching based on labels, wavelengths, time slots or ports. Thus GMPLS extends MPLS in order to provide support for modules that cannot identify cell and packet boundaries. The following types of modules are identified as part of LSR's in a GMPLS environment.

1) PSC (Packet-Switch Capable) For example, switching based on MPLS shim header or ATM's VPI/VCI

2) TDMSC (Time-Division-Multiplex-Switch-Capable) For example, SDH/SONET cross-connect or G.709 interface
3) LSC (Lambda -Switch-Capable) For example, interfaces that switch based on a wavelength or waveband
4) FSC (Fiber-Switch-Capable) For example, photonic cross-connects that switch one fiber to another

GMPLS consists of several building blocks most of which are the familiar Internet routing and signaling protocols that have been extended. A new specialized protocol LMP (Link Management Protocol) [24] has been developed for GMPLS. LMP automates the link provisioning and fault isolation tasks. It also provides bundling of links because maintaining link adjacencies is no more scalable when technologies like DWDM (Dense Wavelength Division Multiplexing) are used resulting in thousands of links between adjacent nodes.

GMPLS extends the labels used in MPLS by allowing information such as time-slots, wavelengths and port identification to be used in labels. Thus a generalized label can identify a single fiber within a bundle, a single wavelength or a time slot. Each LSP is established and maintained independently even though it may be nested within another LSP.

In short, GMPLS is expected to act as a unifying protocol for packet, circuit and optical switching technologies and resolve the control plane issues for managing diverse types of communications within the Internet.

6 EXPERIMENTING WITH TRAFFIC ENGINEERING IN AN MPLS DOMAIN

In this section, the work of one of the authors (Zubairi) is described. This work is being carried out in the department of mathematics and computer science of SUNY at Fredonia. Some experimental results are presented which have been obtained with a C++ software that was developed as part of this project. The contribution of student programmer Jason Beuckman, a senior student in computer science, is hereby acknowledged.

The target of the work is to develop efficient path allocation techniques that meet TE requirements. TE requirements are specified in a set of traffic trunks (flows) that are presented to an MPLS domain. TELIC (Traffic Engineering with Link Coloring) algorithm works to allocate LSP's in the domain to the flows in order to meet the constraints and enhance the overall utilization of the resources. TELIC has been discussed in detail in [25].

TELIC works with a configurable MPLS domain. It reads the domain specification from an input file "domain.dat" with the following format:

First line:	An integer representing total number of nodes in the domain
Subsequent lines:	Starting node <space> Destination node <space> Color

Using a domain specification file makes it very easy to simulate various topologies. TELIC then reads the traffic requests from a separate file named "traffic.dat". This file can have any number of lines, with each line having only two values: Diffserv class of traffic trunk, followed by bandwidth requested. Each request is processed and either an LSP is allocated or denied for the current request. TELIC's opening screen and menu are shown in Figure 6 below

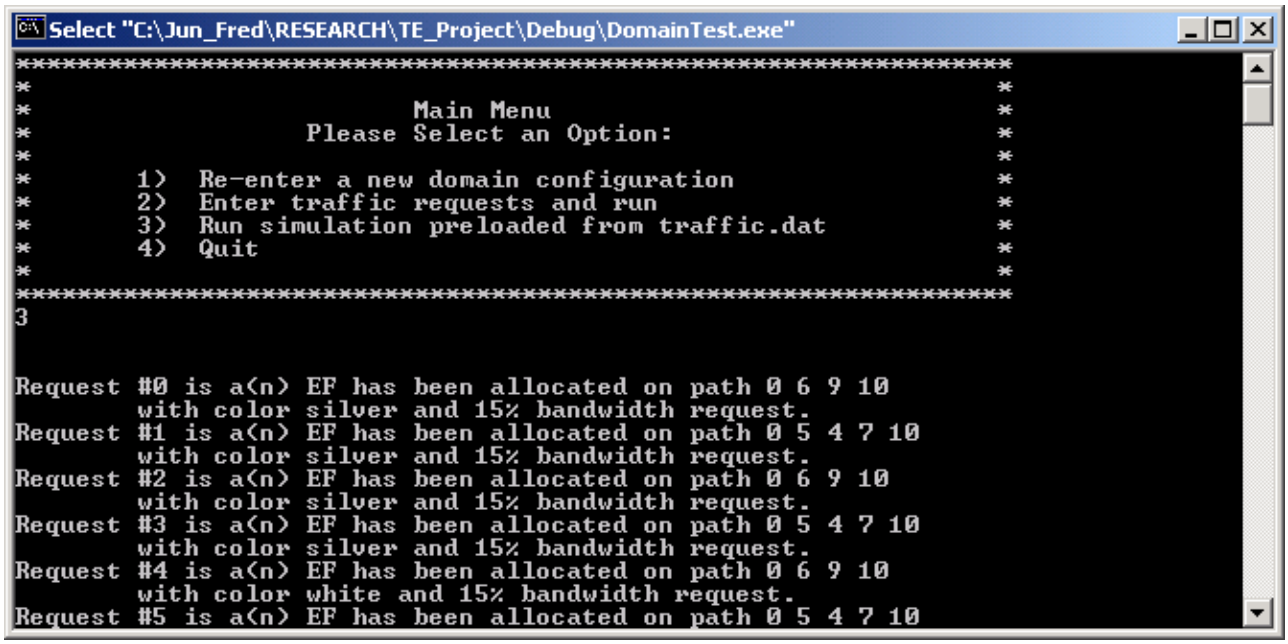

Figure 6: The Opening Screen of TELIC With Menu Options

As seen in Figure 6, TELIC is a flexible traffic engineering tool that can accept new topologies during run time and simulate traffic requests from a data file. The post simulation menu offers a display of final domain, allocated LSP table and remaining bandwidth. TELIC has been developed in Visual C++ using object oriented design. The link cost function and color range are configurable options.

We have used TELIC to process traffic requests in mesh, irregular ISP, disjoint multipath and regular ISP topologies in MPLS-Diffserv domains. Mesh topology includes nodes linked in a two dimensional grid. Disjoint multipath topology is characterized by the availability of multiple non-overlapping paths from an ingress node to the egress node. ISP and irregular ISP topologies include disjoint and shared multiple paths. In all test cases , class-based queuing was used, resulting in 100 percent success rate for EF requests. However, all the requests for AF and DF could not be satisfied. The results for ISP topology are shown in Figures 7 and 8. ISP topology contains 9 nodes and 12 links. One of the nodes was considered the ingress node on which TELIC was run. The target of TELIC was to find suitable paths through the domain to an egress node. The ratio of AF:DF was varied from set 1 to set 15 so that the DF requests are reduced and AF requests are increased. As seen in the graphs, there is a gap between the requested and allocated bandwidth for both Diffserv classes. This gap is due to the fact that some links become congested quickly and there is no possible path to the egress without those links. Therefore TELIC can be used indirectly to find an upper bound on the number of LSP's routable within a domain. In addition to this, it can also find and identify the bottleneck links in a network. The problem of blocked bottleneck links can be solved by adding parallel links or adding alternate paths in the domain for accommodating additional requests.

Currently, the shortest distance LSP allocation algorithm is being implemented in C++ as part of this research. This algorithm considers only the available bandwidth on the candidate links. Identical traffic sets are to be processed using the same topologies. Thus the performance of both algorithms can be measured and compared. The target of this study is to improve the features of TELIC. Future work includes incorporating fault tolerance and modeling traffic with statistical arrival rate and duration.

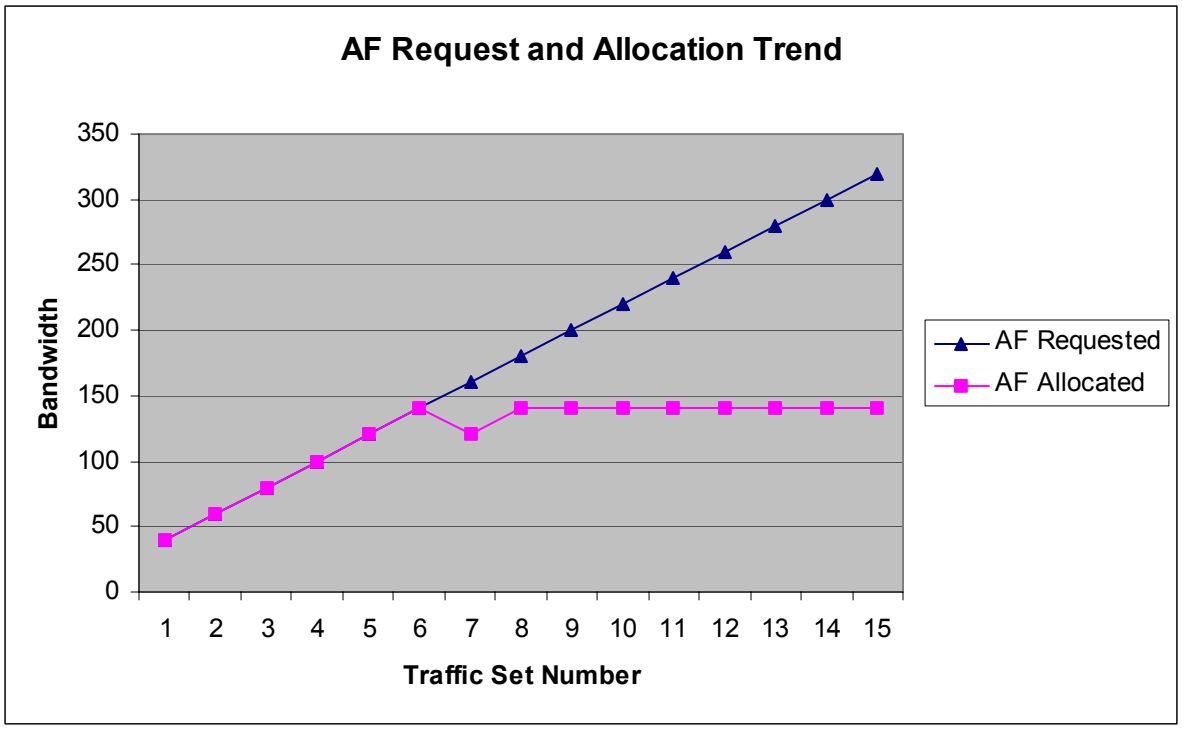

Figure 7: Simulation Results for AF Class Allocation Using ISP topology

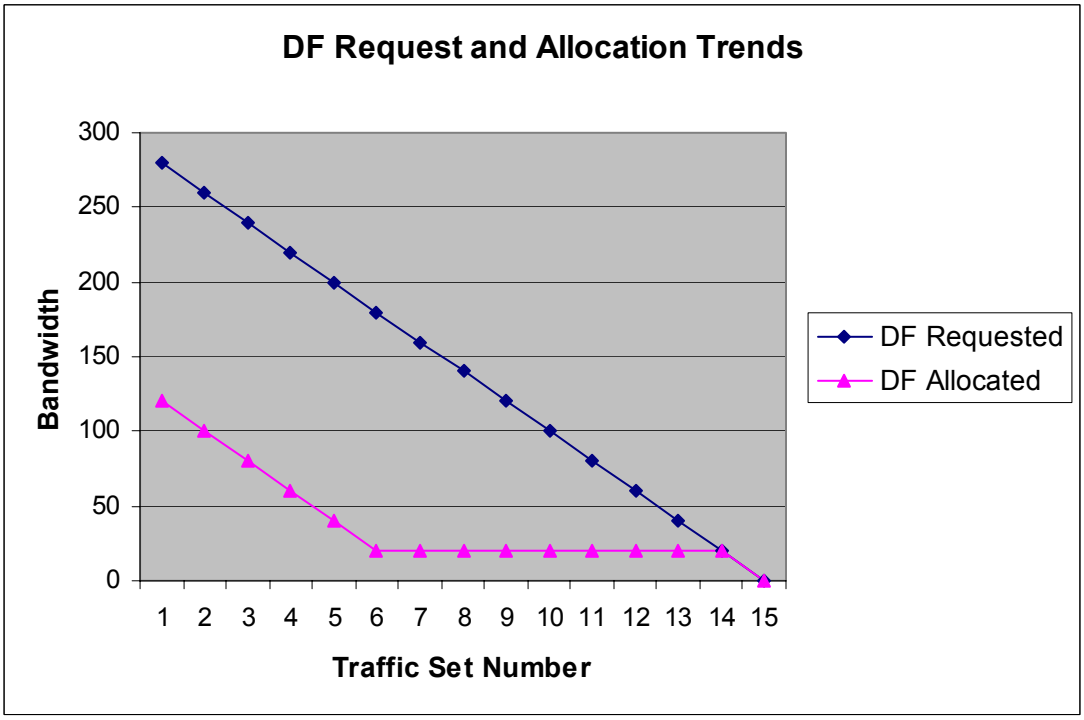

Figure 8: Simulation Results for DF Class Allocation Using ISP topology

CONCLUSION

We have discussed the traffic engineering requirements for the network domains and shown some examples of applying traffic engineering rules for load balancing, fault tolerance and optimized use of network resources. MPLS and its traffic engineering principles have been explained. We have highlighted an effort to build a flexible TE-based LSP allocation tool that can accept various traffic requests and work with different topologies. Results of using this tool on regular ISP topology are presented. We have also discussed the current efforts to define a generalized protocol named GMPLS that can handle optical, packet and circuit switching equally well.

REFERENCES

[1] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, *"RSVP: a new resource ReSerVation protocol,"* IEEE Network, vol. 7, pp. 8-18, Sept. 1993.

[2] B. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, *"Resource ReSerVation protocol (RSVP) - version 1 functional specification"* RFC 2205, IETF, Oct. 1997.

[3] J. Wroclawski, *"The use of RSVP with IETF integrated service"* RFC 2210, IETF, Oct. 1997.

[4] P. Pan and H. Schulzrinne, *"Staged refresh timers for RSVP"* in Proceedings of Global Internet, (Phoenix, Arizona), Nov. 1997. also IBM Re-search Technical Report TC20966.

[5] K. Nichols et. al. *"Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers"* in IETF, De. 1998, RFC 2474

[6] S. Blake et. al. *"An Architecture for Differentiated Services"* in IETF, Dec. 1998, RFC 2475

[7] V. Jacobson et. al. *"An Expedited Forwarding PHB"* in IETF, June 1999, RFC 2598

[8] J. Heinanen et. al. *"Assured Forwarding PHB Group"* in IETF, June 1999 RFC 2597

[9] C. Semeria *"Traffic Engineering for the New Public Network"* , White Paper, Juniper Network, 2000

[10] X. Xiao and L. Ni *"Traffic Engineering with MPLS in the Internet"*, Michigan State University, 2000

[11] D. Awduche et. al. *"Requirements for Traffic Engineering Over MPLS"*, RFC 2702, IETF, Sep 1999

[12] D. Awduche et. al. *"A Framework for Internet Traffic Engineering"*, IETF, Internet Draft, work in progress, Jul 2000 < draft-ietf-tewg-framework-02>

[13] G. Ash *"Traffic Engineering & QoS Methods for IP, ATM, & TDM-Based Multi-service Networks"*, , IETF Internet draft, work in progress, < draft-ash-te-qos-routing-01>, Jul 2000

[14] X. Xiao and L. Ni *"Internet QoS: A Big Picture"*, IEEE Network March/April 1999, pp8-18

[15] X. Xiao *"Providing Quality Of Service In The Internet"*, Department of Computer Science and Engineering, Michigan State University, Ph. D. thesis, 2000

[16] R. Callon et. al *"A Framework for Multiprotocol Label Switching"*, IETF Internet Draft, work in progress, , <draft-ietf-mpls-framework-05>, Sept. 1999

[17] P. Brittain and A. Farrel, *"MPLS Traffic Engineering:A Choice Of Signaling Protocols Analysis of the similarities and differences between the two primary MPLS label distribution protocols: RSVP and CR-LDP"*, Data Connection, 2000

[18] L. Li, et. al., *"IP Traffic Engineering Using MPLS Explicit Routing in Carrier Network, Between The Signaling Approaches: CR-LDP and RSVP"*, Nortel Networks, Jan. 2000

[19] L. Widjaja *"Communication Networks"* , McGraw-Hill, 2000

[20] White paper, *"Traffic Engineering with Multiprotocol Label Switching"*, Avici Systems, March 2000

[21] B. Jamoussi and P. Ashwood-Smith, *"MPLS and its Applications"* , Nortel Networks, April 2000

[22] Discover Innovation award in Communications, *"Electro-Holography Technology and All-Optical Intelligent Lambda Switch"*, Discover Magazine Vol.22 No.7, July 2001, page 81

[23] E. Mannie et. al., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture" IETF, Internet Draft, Work in progress, February 2001 <draft-many-gmpls-architecture-00.txt>

[24] A. Banerjee et. al., "Generalized Multiprotocol Label Switching: An Overview of Routing and Management Enhancements" IEEE Communications Magazine Vol.39 Issue 1, January 2001, pp 144-150.

[25] J. A. Zubairi, "An Automated Traffic Engineering Algorithm for MPLS-Diffserv Domain", in Proc. ATSC'02, San Diego, USA April 2002.