GLASS (GMPLS Lightwave Agile Switching Simulator) - A Scalable Discrete Event Network Simulator for GMPLS-based Optical Internet

Youngtak Kim, Eunhyuk Lim, Chul Kim, Kwangil Lee, Douglas Montgomery,
Oliver Borchert, Richard Rouil, David Su

Advanced Network Technologies Division (ANTD),
National Institute of Standards and Technology (NIST)
820 West Diamond Avenue, Gaithersburg, MD 20899, U.S.A.
(Tel: +1-301-975-3613; Fax: +1-301-590-0932; E-mail ytkim@yu.ac.kr)

Abstract – In this paper, we explain the design philosophy and the overall architecture of a scalable discrete event network simulator for GMPLS-based Optical Internet, called ***GLASS*** *(GMPLS Lightwave Agile Switching Simulator).* GLASS has been developed to support the R&D works in the area of Next Generation Internet (NGI) networking with GMPLS-based WDM optical network, and Internet traffic engineering with DiffServ-over-MPLS. It supports discrete-event simulations of various DiffServ packet classification, per-hop-behavior (PHB) processing with class-based-queuing, MPLS traffic engineering, MPLS OAM functions that provide performance monitoring and fault notification, GMPLS-based signaling for WDM optical network, link/node failure model, and fast restoration from an optical link failure.

The NIST GLASS is implemented with Java programming language on the SSFNet (Scalable Simulation Framework Network) simulation platform. It has been designed and implemented with open interfaces to support future expansions or replacements of protocol modules by users. It also provides DML description input file interface to support the users' flexible definition and modification of simulation parameters and configuration of protocol modules. The simulation outputs are generated in text file format that can be used by Excel to produce various graph or charts. Recently, a GUI-based topology and simulation creator has been developed to support a visual environment from which one can configure and run the simulator.

Keywords – MPLS/GMPLS, WDM Optical Network, DiffServ, Network Simulation, Traffic Engineering

I. Introduction

A. Motivation

In order to manage the explosively increasing Internet traffic more effectively, various traffic engineering and networking technologies have been proposed, developed and implemented. The physical link bandwidth has been expanded with DWDM optical transmission technology and Optical Add-Drop (OADM) & Optical Cross Connect (OXC) switching technologies [1]. MPLS (Multi-Protocol Label Switching) has been introduced to enhance the packet forwarding & switching performance by using faster fixed-label switching at layer 2.5 [2]. By using the connection-oriented, bandwidth reserved MPLS LSP (Label Switched Path) among the core routers, the traffic engineering has been more flexible and predictable. MPLS architecture, which had been basically designed upon packet switching capability, recently has been generalized into Generalized MPLS (GMPLS) to include other switching capabilities, such as TDM circuit switching, fiber/lambda switching with generalized label [3]. The implementation of IP-based control plane for the next generation optical network with the GMPLS control architecture has been received great interests recently; and it has been accepted by the optical network equipment vendors and network operators. The DiffServ technology has been developed to provide differentiated quality-of-service (QoS) according to the user's requirements or necessity [4]. Especially, the protocol structure of "DiffServ-aware-MPLS with GMPLS-based WDM Optical Network" has been emphasized as a promising technical solution for Next Generation Internet.

These newly proposed and developed Internet networking and traffic engineering technologies are currently standardized individually by IETF, ITU-T, OIF and other related forums. As a result, the inter-operability, complexity, scalability and effectiveness of the integrated operations with various new protocol modules have become the major concerns of Internet Service Providers (ISP) and network operators, as well as the system vendors.

To test and evaluate the inter-operability and effectiveness of the newly proposed protocol functions, the implementation of prototype systems and configuration of a trial test-bed network is one possible approach; but it usually takes long time and is costly. As a more practical approach, the network simulation with the configurable node protocol structure and the scalable network size is used in popular by many researchers and system developer. Network Simulator (ns) [6], JavaSim [7], SSFNet [8], and OPNET [9] are the most popularly used network simulators. But, these network simulators do not support the integrated simulation of "DiffServ-aware-MPLS" on the "GMPLS-based WDM Optical Network" with OAM functions and fault restoration functions.

B. Network Simulation for DiffServ-aware-MPLS on the GMPLS-based WDM Optical Network

The GLASS has been developed for the integrated simulations of Next Generation Internet (NGI) networking with GMPLS-based WDM optical network, and Internet traffic engineering with DiffServ-over-MPLS [10]. It supports the discrete-event simulations of various DiffServ packet classification, per-hop-behavior (PHB) processing with class-based-queuing, MPLS traffic engineering, MPLA OAM for performance monitoring and fault restoration, GMPLS-based signaling for WDM optical network, link/node failure model, and fast restoration from link or node failure.

GLASS has been implemented on the SSFNet (Scalable Simulation Framework Network) simulation platform. It has been designed and implemented with open interfaces to support future expansion and replacement of protocol modules by users. It also provides DML description input file interface to support the users' flexible definition/modification of simulation parameters and configuration of protocol modules.

Recently, a GUI-based topology and simulation creator (GLASS-TSC) has been developed to support a visual environment from which one can configure and run the simulator [10].

The rest of this paper is organized as follows. Section II describes the basic concept and the operations of "DiffServ-aware-MPLS Traffic Engineering", and "GMPLS-based WDM Optical Networking." Section III explains the target scalable network simulation of the Internet networking and traffic engineering on the GMPLS-based optical network. Section IV describes the architecture of GLASS, and Section V summarizes the paper.

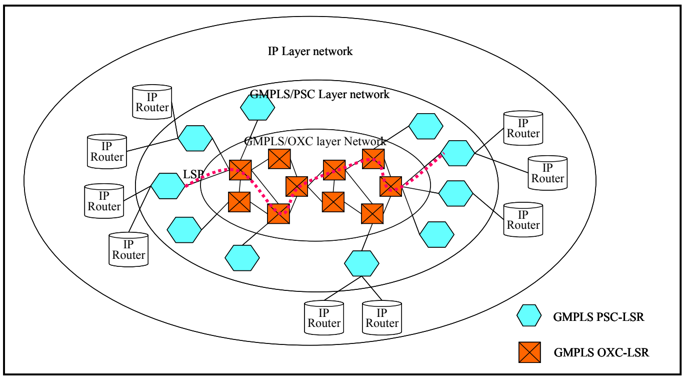
II. DiffServ-aware-MPLS traffic engineering and Generalized Multi-Protocol Label Switching (GMPLS)

A. Networking Model of Next Generation Internet

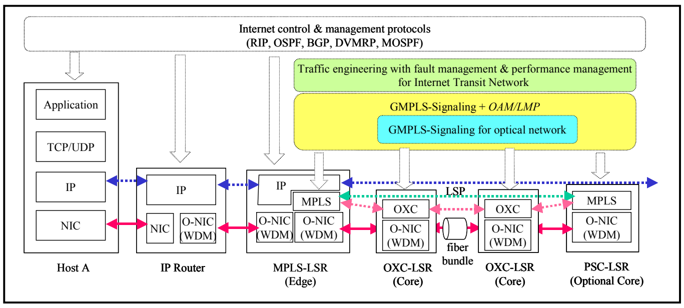
One of the most important functional requirements of Next Generation Internet is an efficient traffic engineering mechanism to manage the explosively increasing Internet traffic and to provide QoS-guaranteed services to end users [11-14]. Also, in order to provide the sufficient bandwidth required for the multimedia applications, the DWDM optical network has been developed and deployed as the backbone transit network. For more flexible internetworking scenarios and easy implementations, the IP-based optical network control with the GMPLS (Generalized Multi-Protocol Label Switching) [15, 17, 18] architecture has been designed and implemented recently. GMPLS-based control plane for the optical transport network provides great flexibility in the inter-networking of IP/MPLS network and optical network.

Figure 1(a) shows the domain Interworking model of next generation Internet with IP layer network, MPLS layer network and optical layer network. In GMPLS architecture, the MPLS layer network and the optical layer network can be interconnected in overlay model or peer-to-peer model. In overlay model, the MPLS layer network is the client of optical layer network, and MPLS layer network sends requests of the optical path setup through the O-UNI (optical user-network interface) signaling [16]. The routing information of the optical domain is not provided to the client MPLS layer network. In peer-to-peer model, the optical lambda channel is modeled as just another LSP with generalized label (fiber ID and lambda ID) that has bigger bandwidth, and the routing information of optical network's OXC-LSR is provided to the MPLS packet switching capable (PSC) router (PSC-LSR) to be used in the calculation of a routing decision.

As shown in Figure 1(b), the MPLS layer network and optical layer network can both have the same control plane functions based on the GMPLS-signaling architecture. This unified control plane provides various advantages, such as various inter-working interface model across domains, integrated traffic engineering [19-29], and efficient integrated fault restoration.



(a) Domain Interworking Model


Internet control & management protocols
(RIP, OSPF, BGP, DVMRP, MOSPF)
Traffic engineering with fault management & performance management
for Internet Transit Network
GMPLS-Signaling + OAM/LMP
GMPLS-Signaling for optical network
Application
TCP/UDP
IP
NIC
IP
NIC
O-NIC (WDM)
IP
MPLS
O-NIC (WDM)
O-NIC (WDM)
LSP
OXC
O-NIC (WDM)
fiber bundle
OXC
O-NIC (WDM)
MPLS
O-NIC (WDM)
Host A
IP Router
MPLS-LSR (Edge)
OXC-LSR (Core)
OXC-LSR (Core)
PSC-LSR (Optional Core)

(b) Protocol Structure of GMPLS

Figure 1. Networking model of Next Generation Internet

B. Internet Traffic Engineering

A major goal of Internet traffic engineering is to facilitate efficient and reliable network operations while simultaneously optimizing network resource utilization and maximizing traffic performance [20-29]. The key performance objectives associated with traffic engineering (TE) are either traffic-oriented or resource-oriented. The traffic-oriented performance objectives include the aspects that enhance the QoS of traffic stream, such as minimization of packet loss, minimization of delay, maximization of throughput, and enforcement of service level agreements. The resource-oriented performance objectives include the aspects pertaining to the optimization of resource utilization.

In order to accomplish the objectives of traffic engineering, we must consider the service level specification/agreement, the Internet traffic engineering with DiffServ which manages the micro-flow of each service class, the DiffServ-aware-MPLS traffic engineering with traffic & QoS parameters that manages the MPLS LSP for the aggregated flow of one or more DiffServ class-types. Figure 2 shows the overall traffic engineering architecture.

In service level specification, the objective QoS parameters of the requested service traffic flow should be specified, and the specification must be agreed or contracted by both the service client and the network service provider. ITU-T recommendation Y.1541 provides a good example of the service level specification [22]. In order to guarantee the required QoS and to provide better bandwidth utilization, DiffServ defines the Per-Hop-Behavior (PHB) at each IP/MPLS router node. The PHB includes the class-based-queuing with specific metering/ measuring and coloring, dropping policy, queuing, packet scheduling and optional traffic shaping.

MPLS provides various attractive features of traffic engineering based on explicitly labeled & switched path. The explicitly labeled paths are not constrained by the destination-based forwarding paradigm, but it can potentially be efficiently managed by their traffic parameters. The traffic trunk of an aggregation of traffic flows of the same class can be easily mapped onto LSPs, and a set of attributes can be associated with traffic trunks that modulate their behavioral characteristics. Also, a set of attributes can be associated with resources that constrain the placement of LSPs and traffic trunks across them. MPLS allows flexible traffic aggregation, and it is relatively easy to integrate a constraint-based routing with lower overhead.

To provide the traffic engineering capability, the existing signaling and routing protocol modules must be expanded. As routing and signaling protocol with traffic engineering extensions, OSPF-TE [23-26], RSVP-TE[17], IS-IS-TE[27], CR-LDP[18] are under standardization in IETF. To support the inter-domain traffic engineering, the TE extensions to BGP-4 protocol have been proposed [44-48]. The signaling and routing protocols with TE extensions basically provide mechanisms of maintaining the link state information database according to the specified TE parameters, such as physical distance, available bandwidth, allocated bandwidth, residual error rate, resource color, shared risk link group (SRLG) identifier, etc.

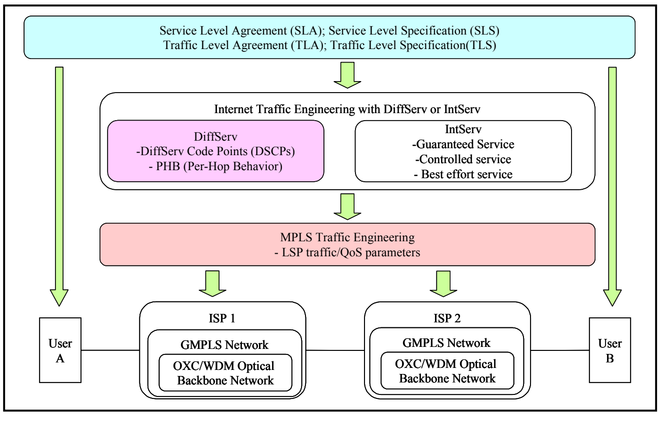

Service Level Agreement (SLA); Service Level Specification (SLS)
Traffic Level Agreement (TLA); Traffic Level Specification(TLS)
Internet Traffic Engineering with DiffServ or IntServ
DiffServ
-DiffServ Code Points (DSCPs)
- PHB (Per-Hop Behavior)
IntServ
-Guaranteed Service
-Controlled service
- Best effort service
MPLS Traffic Engineering
- LSP traffic/QoS parameters
ISP 1
GMPLS Network
OXC/WDM Optical Backbone Network
ISP 2
GMPLS Network
OXC/WDM Optical Backbone Network
User A
User B

Figure 2. Internet traffic engineering

C. DiffServ-aware MPLS Traffic Engineering

Differentiated Service (DiffServ) with Per-Hop-Behavior (PHB) has been developed to provide a QoS-guaranteed packet transmission [30-43]. According to the source/destination address, service type, and protocol ID of IP packet header, we can define 64 different classes with distinct DiffServ Code Points (DSCP) [30]. In order to simplify the classification of DiffServ, a set of DiffServ classes is defined as a *class-type* where the classes in the same class-type possess common aggregate maximum and minimum bandwidth requirements to guarantee the required performance level. Even though there is no maximum or minimum bandwidth requirement to be enforced at the level of an individual class within the class-type, we can use the priority polices for classes within the same class-type in terms of accessing the class-type bandwidth (e.g. via the use of preemption priorities). Table 1 shows an example definition of class-type and their performance objectives.

Table 1. Example of DiffServ Class-type and performance objectives

Class-type Nature	Objective	Example	Delay	Jitter	packet Loss Ratio	Bandwidth definition
NCT1/ NCT0	Minimized error high priority	RIP, OSPF, BGP-4	100 msec	U	10^{-3}	Peak rate
EF	Jitter sensitive real-time high interaction	VoIP	100 msec	10 msec	10^{-3}	Peak rate
AF4	Jitter sensitive real-time high interaction	Video conference	250 msec	20 msec	10^{-3}	Committed rate
AF3	Transaction data interactive	Terminal session Custom app	250 msec	U	10^{-3}	Committed rate
AF2	Transaction data	Data base Web	250 msec	U	10^{-3}	Committed rate
AF1	Low loss bulk data	FTP E-mail	1 sec	U	10^{-3}	Committed rate
BE	Best effort	Best effort service	U	U	10^{-3}	U

The mapping of DiffServ-class-types into MPLS LSP (Label Switched Path) can be implemented in either E-LSP (Exp-inferred-LSP) or L-LSP (Label-only-inferred LSPs) model. In E-LSP model, LSPs can transport multiple class-types (ordered aggregates), and the EXP field of the MPLS shim header conveys the PHB to be applied to the packet (conveying both information about the packet's scheduling treatment and its drop precedence) at each LSR. In L-LSP model, each LSP only transports a single class-type, so the packet's treatment is inferred exclusively from the packet's label value, while the packet's drop precedence is conveyed in the EXP field of the MPLS shim header.

E-LSP model has merit of easier connection handling and protection; the creation of a single LSP for end-to-end services for a customer is easier that the setting up, maintaining, administering and monitoring multiple LSPs for each class-type. Also, E-LSP model requires reduced number of LSPs needed to deploy end-to-end services in a network. The path protection and switching mechanisms are more easily applied to a single LSP that a group of related LSPs. Finally, the bandwidth borrowing among the class-types of a customer is much easier.

In MPLS networking, the guaranteed provisioning of bandwidth is controlled by the per-LSP queue and the MPLS packet scheduler. Several LSPs can be encapsulated by an outer LSP using the hierarchical LSP stacking, and this hierarchical LSP stacking can be applied recursively. Each outer LSP is also specified with its own traffic parameters as explained above. If there is any available excess bandwidth in the outer LSP, the excess bandwidth is allocated to the inner LSPs in proportion to their weights in addition to their committed data rates. Since the LSP stacking is organized in recursive manner hierarchically, the available excess bandwidth in the outmost LSP should be recursively allocated to the inner LSPs according to the inner LSPs' weights. By this reallocation of the excess bandwidth, we can increase the utilization of network resources.

D. GMPLS-based WDM Optical Networking

The major equipments for WDM optical networking are optical cross-connect (OXC) with optical add-drop multiplexing (OADM), and the DWDM network interface card (DWDM-NIC) for the client nodes (such as MPLS LSR or IP Router). OXC/OADM provides wavelength routing & switching, wavelength conversion, fiber/port switching, and waveband switching. The optical switching functions are implemented in either all-optical switching architecture or with Optical-Electrical-Optical (O-E-O) architecture. According to the architecture and the modules, the lambda conversion and lambda switching may have different limitations, such as the number of wavelength converters in a OXC and the range of wavelength conversion. Through the add-drop ports of OXC/OADM, the optical frames are delivered to the upper protocol layers, such as MPLS or IP layer. Figure 2 shows a typical optical domain model.

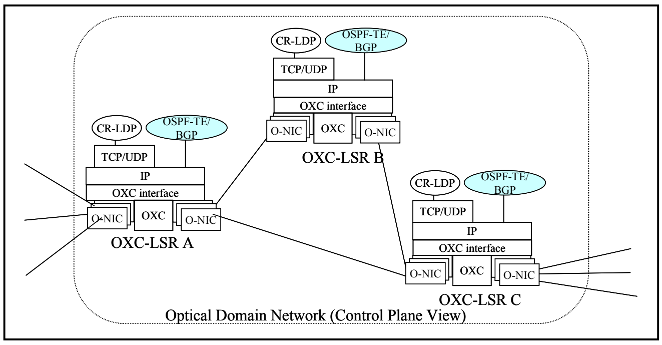


Figure 3. Optical domain model

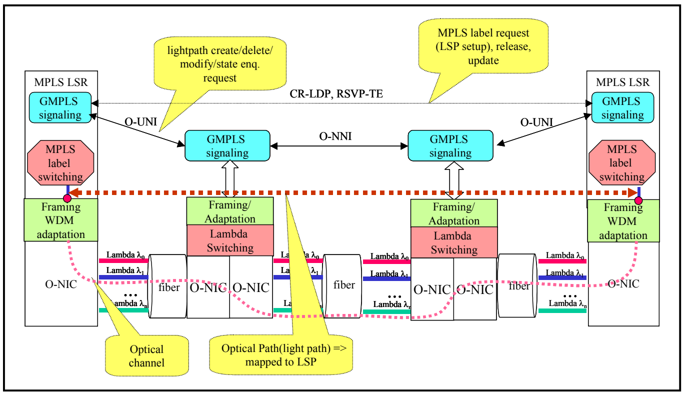


Figure 4. MPLS-over-Optical Network

The most important role of the control plane of WDM optical network is to set up light paths according to the requests via O-UNI signaling, modify, delete or status enquiry of the light paths. For this light path handling, the control plane should have the functional modules of signaling (such as CR-LDP [18] or RSVP [17], BGP[44-48]), routing (such as OSPF or ISIS), and wavelength assignment. Recently, Optical Internetworking Forum (OIF) generated the O-UNI 1.0 standard document [16] and IETF generated the LMP (Link Management Protocol) draft document for the control channel management and the link property correlation [19].

As shown in Figure 3, the protocol modules of control plane for OXC/OADM are IP-based; the interactions with its peer nodes are done through IP packet forwarding. Figure 4 shows the O-UNI between MPLS-LSR and OXC-LSR, and the O-NNI signaling among OXC-LSRs.

E. Interworking Models of Optical Internet

The internetworking of IP/MPLS network and optical network can be considered in service model, interaction model and routing approaches [13,14]. In service model, the IP layer network and optical layer network can operate in either client-server service model or in integrated service model. In client-server model, the optical network primarily provides a set of bigger bandwidth pipes to the client IP/MPLS layer, while in integrated service model, the IP layer network and the optical layer networks are treated as a single network and there is no distinction between the optical switches and the IP/MPLS routers as far as the control plane goes.

In IP/MPLS-over-optical interaction model, three scenarios have been suggested: overlay model, peer model and augmented model [13, 14]. In overlay model, the optical network provides point-to-point connection to the IP/MPLS domain. The IP/MPLS routing protocols are independent of the routing and signaling protocols of the optical layer. When the network operators of the IP/MPLS layer network and the optical layer network are different, this overlay model would be used in consideration of the privacy and security of network status information. In peer model, the optical routers and optical switches act as peer nodes and there is only one instance of a routing protocol running across the optical domain and the IP/MPLS domain. A common IGP like OSPF or IS-IS may be used to exchange topology information. OPSF opaque Link State Advertisement (LSA) and extended type-length-value (TLV) encoded fields may be used to in the case of IS-IS. The assumption in this model is that all the optical switches and the IP/MPLS routers have a common addressing scheme. In augmented model, there are actually separate routing instances in the IP/MPLS and optical domains, but information from one routing instance is provided into the other routing instance. For example, IP addresses could be assigned to optical network elements and carried by optical routing protocols to allow reachability information to be shared with the IP domain to support some degree of automated discovery.

Three routing models (fully peered routing, domain specific routing and overlay routing) have been suggested as the routing approaches [13,14]. The fully peered routing model is used for the peer interaction model, where one instance of the routing protocol running in the IP/MPLS and optical domains. The domain specific routing model supports the augmented interaction model where the routing instances are separated for the IP/MPLS domain and the IP domain. The inter-domain routing protocols like BGP may be used to exchange information between the IP/MPLS and optical domain. OSPF areas may also be used to exchange routing information across the two domains [14]. The overlay routing model is much like the IP-over-ATM that supports the overlay interaction model. The optical paths for the IP packet delivery are set up across the optical network. Address resolution similar to that in the IP-over-ATM is required. The optical domain network can maintain a registry of IP addresses and client (e.g. IP/MPLS router or virtual private network (VPN) [44]) identifiers it is connected to. On querying the database for an external IP address it would return the appropriate ingress/egress port address on the OXC. Once optical paths are created, the client layer network routing adjacencies can be formed using OSPF. The IP/MPLS network would then be "overlayed" on the underlying optical network that may have an independent routing function.

III. Scalable Discrete Event Simulation of GMPLS-based Next Generation Internet

A. Objectives of Scalable Simulation for Internet

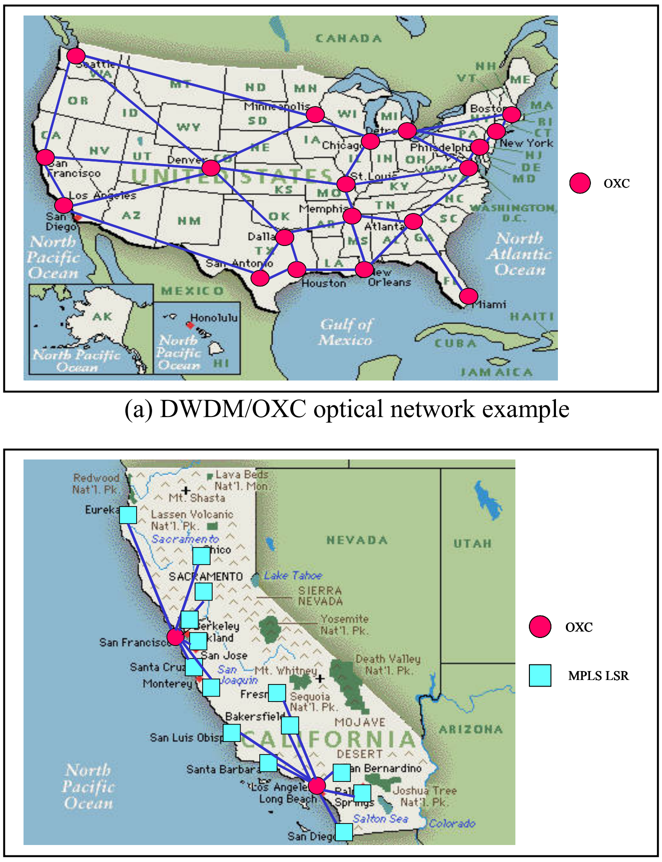
In order to model the dynamic feature of the Internet, and to analyze the global effect of Internet traffic engineering at a large network scale, a scalable network-modeling framework with discrete-event simulator is essential. To correctly model the physical network topology and the capacities of nodes and links, the simulator should be able to generate the simulation result for a network of at least greater than 100s of nodes with arbitrary link connectivity, in a reasonable time. For the simulation of Next Generation Internet with DiffServ, MPLS, and Optical Switching capability, we need at least following four node models: (a) user host with Internet application (e.g. Web, FTP, Telnet, VoIP and/or Teleconference) traffic generation/reception, (b) IP router with IP packet forwarding, (c) MPLS-LSR with IP packet forwarding, optional DiffServ packet classification and class-based-queuing, MPLS label switching, and WDM optical network interface, (d) OXC node with lambda conversion/ switching, lambda add-drop, and optional fiber/wave-band switching.

As an example of large-scale Internet networking with physical network topology, Figure 5 shows the possible Internet configuration for United State with 10~50 OXC nodes, 5~10 MPLS-LSR nodes per each OXC nodes, and 5~10 IP routers per each MPLS-LSR node. As a result, this example Internet should consider 10~50 OXC nodes, 50~500 IP/MPLS routers, and 100 ~ 1000 user hosts.

Since the Internet protocols are continuously proposed and updated in various IETF working groups, the integrated operation of protocol modules either inside of a specific node or among multiple peer nodes in distribution should be tested with network simulator. For example, the integrated fault restoration mechanism in MPLS-over-Optical network is one of the most important research topics. Especially in the GMPLS-based control plane for WDM optical network, the interactions among related protocol modules for traffic engineering, such as OSPF-TE, BGP-TE, LMP, and CR-LDP/RSVP-TE, must be carefully evaluated.

The abstraction level of the protocol operations may be different according to the objectives of the simulation. For example, the user data traffic generation and transmission can be eliminated if the objective of the simulation is only the analysis of the operations of signaling protocol that sets up, delete or modify the user data connections. If the major interest of the simulation is to test the effectiveness of the class-based-queuing mechanism for the DiffServ-aware-MPLS traffic engineering, the dynamic flooding of OSPF link state advertisement can be simplified in the simulation.

For better processing performance in time-consuming network simulation of a large-scale Internet, a multi-processing or parallel/distributed processing will be useful to reduce the processing time. This efficient parallel processing is usually supported by the simulator kernel platform, such as SSF.



(a) DWDM/OXC optical network example

(b) Regional MPLS network topology

Figure 5. Example network configuration for United State

B. Discrete Event Simulation

By the discrete event simulation of Internet, we usually check the proper sequencing and timing of packet forwarding at each protocol layer module, event handling by signaling and management protocols, fault notification and restoration, and congestion control. For the discrete event simulation, the actual packets of user application data are generated at the user host, and the IP packets of signaling message are generated at IP/MPLS routers and OXC/OADM nodes. The simulation time of each event is recorded according to the simulation configuration.

Since the packet generation and forwarding requires processing burden in simulation, some part of the messaging in network operation is abstracted if it is not the major concern. In that case, the message passing among remote nodes may be simplified by direct parameter reading in a centralized process without effecting the major discrete event simulation.

Usually, the content of user application message does not have significance when the major concern of the network simulation is the verification of the operation of network nodes and the protocol modules. Thus the IP/MPLS routers and OXC optical switching node do not consider the payload part of user data messages.

C. SSF and SSFNET

The SSF (Scalable Simulation Framework) is a discrete event simulation platform for the construction and simulation of very large networks [8]. SSF can execute detailed simulations of complex topology networks with a large number of concurrent TCP/IP flows. SSF application programming interface provides a compact, high-level target for simulator implementers, hiding all details of simulator internals (threads, processors, event queues, and synchronization) from the modeler. SSF defines just five base classes: Entity, inChannel, outChannel, Process, and Event. These five classes form a self-contained design pattern for constructing process-oriented, event-oriented, and hybrid simulation.

SSFNET is the first collection of SSF-based models for simulating Internet protocols and networks [8]. SSFNET packages provide classes that can be used directly or extended to construct very large network models. The SSFNET libraries include component models for network elements (hosts, routers, network interface cards, local area networks) and network protocols (IP, UDP, TCP, BGP, and static OSPF). In SSFNET, TCP operates on top of IP; IP operates on top of one or more pseudo-protocols representing configured network interface cards. Each network interface card (NIC) maintains a pair of buffered SSF channels for exchanging IP packet events with the outside world. A NIC may be connected to another NIC of the same link type, or to a LAN, and supports self-configuration options for physical link characteristics. Figure 6 shows the protocol graph of SSFNET.

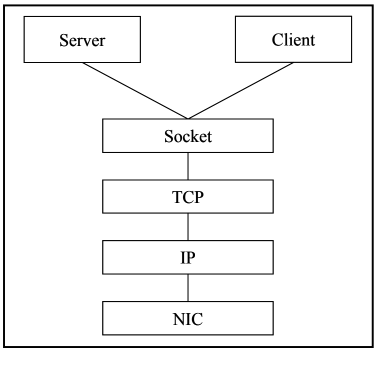


Figure 6. SSFNET protocol graph

The SSFNET does not include network protocols for traffic engineering and GMPLS-related functions of Next Generation Internet, such as DiffServ, MPLS, Optical Network, and GMPLS signaling. In order to provide the WDM optical network interface where multiple fiber link are used per NIC, and multiple lambda channels are available in each fiber link, the SSFNET NIC module cannot be used as it is. Also, the MPLS and GMPLS-signaling related protocols, such as CR-LDP/RSVP-TE, MPLS-OAM, and LMP must be implemented. The traffic engineering (TE) extensions in CR-LDP/RSVP, OSPF, BGP, and O-UNI should be newly implemented. The current OSPF module of SSFNET is static version that does not dynamically update link state database by the link state advertisement (LSA).

SSF models use a simple hierarchical attribute tree notation (DML: Domain Modeling Language) to specify a tree of configuration parameters for each of the components that makes up a lager model. The SSFNET provides a very simple simulation output function that is based on programmer-defined log file without on-the-fly graphic function. The simulation data in the log file can be used as the input to the users' own graphic tool.

IV. Architecture of GLASS

A. Target Simulation Features

The NIST GLASS is a GMPLS-based Optical Internet simulator. It has been developed to support the R&D works in the area of Next Generation Internet (NGI) networking with GMPLS-based WDM optical network, and Internet traffic engineering with DiffServ-over-MPLS. It supports the discrete-event simulations of various DiffServ packet classification and per-hop-behavior (PHB) processing with class-based-queuing, MPLS traffic engineering, GMPLS-based signaling for WDM optical network, link/node failure model, and fast restoration from an optical link failure.

The GLASS is implemented on the SSFNet (Scalable Simulation Framework Network) simulation platform. It has been designed and implemented to provide open interfaces to support future expansion or replacement of protocol modules by its users. It also provides DML description input file interface to support the users' flexible definition/ modification of simulation parameters and configuration of protocol modules. Currently, it provides log output in Excel file format by which the user can easily generate various graphs and charts for his research documents.

One of the important design goals of the GLASS has been the modularity. By implementing the simulation modules in modular structure, the user can easily choose what he/she requires, and configure the node models according to his/her major interests of analysis.

The most important functional blocks are basic Internet networking block, DiffServ block, MPLS networking block, and optical networking block. The basic Internet networking block is composed of the usual Internet host functions with Internet application, socket with TCP and UDP, IP protocol and its routing protocols (OSPF and BGP). Most of the protocol modules of the basic Internet networking block are based on the SSFNET modules.

The other three functional blocks (DiffServ, MPLS and optical networking) have been augmented to the SSFNET modules. The detailed functional architecture and their operations will be explained in next subsections.

Based on the modular structure of the NIST GLASS, various simulation scenarios with any combination of functional blocks are possible; the basic Internet networking block should be used always for user-to-user communications. For example, Internet networking with DiffServ, MPLS, or Optical networking individually is possible. Also, as the most complex scenario, the integrated Internet networking with DiffServ, MPLS networking and optical network components (O-NIC modules and OXC node) is possible.

B. DiffServ packet processing

Differentiated service provisioning has been proposed and implemented to protect the premium service traffic under the network congestion by giving relatively higher queuing and forwarding priority than other usual best-effort traffic [31]. For the differentiated or classified processing of packets at each IP routers, IETF documents define the differentiated service code points (DSCP), DiffServ class-types, metering and coloring, class-based-queuing with algorithmic packet drop, packet scheduling, and optional traffic shaping [30-43]. The overall DiffServ packet processing model is as shown in Figure 7.

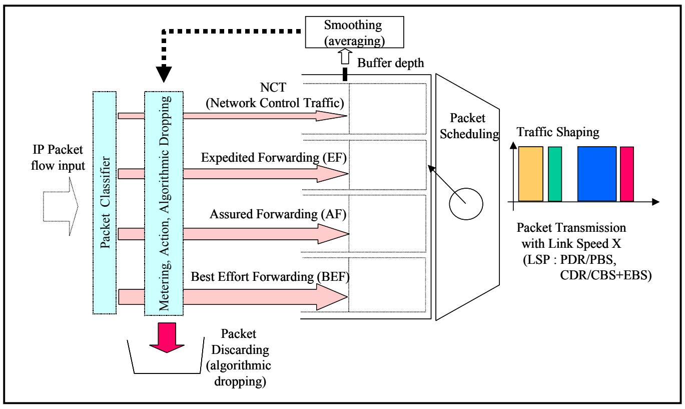

Smoothing (averaging)
Buffer depth
NCT (Network Control Traffic)
Packet Scheduling
Traffic Shaping
IP Packet flow input
Packet Classifier
Metering, Action, Algorithmic Dropping
Expedited Forwarding (EF)
Assured Forwarding (AF)
Best Effort Forwarding (BEF)
Packet Transmission with Link Speed X (LSP : PDR/PBS, CDR/CBS+EBS)
Packet Discarding (algorithmic dropping)

Figure 7. DiffServ Packet Processing Model

The packet classification is implemented with multi-field classifier that uses multiple fields in the IP packet header to determine the differentiated per-hop-behavior (PHB). The IP source address prefix (address and prefix length), destination address prefix, upper-layer protocol, TCP/UDP/SCTP source and destination port range, and ToS (Type of Service) or DSCP (DiffServ Code Point) fields are used in the packet classifier.

The DiffServ class-types are defined according to the performance objectives of end-user service traffic, as shown in Table 1. The DiffServ class-types that are proposed in IETF can be grouped into 4 categories: network control traffic (NCT), expedited forwarding (EF), assured forwarding (AF), and best-effort (BE) forwarding. In the NIST GLASS, 8 class-types, as shown in Table 2, are defined and their class-based-queuing mechanisms are specified in the DML file.

Table 2. DiffServ Class-Type in Simulator

DiffServ Class-type	DSCP	Remark
NCT 1	111 000	
NCT 0	110 000	
EF	101 110	
AF4	100 000	Drop precedence value : 010, 100, 110
AF3	011 000	
AF2	010 000	
AF1	001 000	
BE	000 000	

In order to protect the premium or higher priority traffic flow in network congestion, the packet flow must be firstly measured according to the traffic parameters allocated to each class-type. To measure the arrival intervals of packet, Token Bucket Meter (TBM) or Time Sliding Window (TSW) meters are used with single rate three color marker (SRTCM) or two rates three color marker (TRTCM). In the simulator, TBM with SRTCM is used for NCT and EF class-type where the packet rate is defined by peak information rate (PIR) with peak burst size (PBS), while TBM with TRTCM is used for AF class-type where the packet rate is defined by PDR/PBS and committed information rate (CIR) with committed burst size (CBS). According to the result of the data rate measurement, each packet is colored to Green (conforming PIR/PBS and CIR/CBS), Yellow (conforming PIR/PBS and CIR, but exceeding CBS), or Red (exceeding PIR/PBS).

The class-based-queuing functions include packet discarding according to the drop precedence and priority of the class-type, and packet buffering. Packet dropping at each class-base-queue is implemented in either simple tail-dropping or algorithmic random dropping as in RED (Random Early Detection) or RIO (RED with In/Out-Profile). These three dropping mechanisms are provided in the GLASS. For the algorithmic random dropping, the smoothed queue lengths of each class-base-queue are continuously measured with the exponentially weighted moving average calculation.

The packet scheduler determines the selection of a packet to be transmitted. The packet is selected either by the priority of the queue (in priority-scheduler) or by the relative weight of the queue (in weighted scheduler). In priority scheduling, the queue(s) with higher-priority exclusively use the bandwidth regardless of the lower-priority queue status. In weighted scheduler, the weight for each queue is allocated respectively, and the relative portion of the bandwidth is allocated to the queue by weighted round robin (WRR) or weighted fair queuing (WFQ) mechanism. Also, various combinations of the priority scheduler and the weighted scheduler are possible. For example, we can use the WFQ for all AF traffic flows with specific weight for each AF class-type, while the overall scheduling is handled by a priority scheduler, as shown in Figure 8.

As an optional function, a traffic shaper module can be implemented to control the aggregated packet flow, by controlling the packet interval spaces according to the traffic parameters specified to the aggregated packet flow. This traffic shaping guarantees the conformance of the packet flow, and thus reduces the packet drop probability at the next down stream routers.

When the differentiated service is supported by MPLS networking with DiffServ-over-MPLS structure, the packet flows are mapped to LSP in either E-LSP or L-LSP mechanism. In E-LSP where multiple class-types are mapped into a LSP, the relative administrative priority of each class-type is specified in the EXP field of the MPLS LSP shim header. In L-LSP structure, each DiffServ class-type is supported by separated LSP, and the EXP field represents the relative drop precedence in a class-type. For AF class-type with L-LSP, the lower 3 bits of DSCP (drop precedence) are copied into the EXP field.

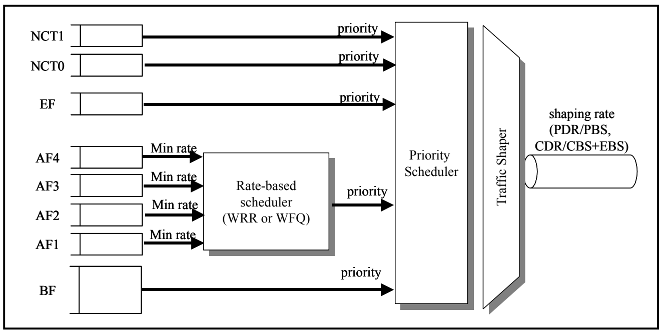


Figure 8. Packet Scheduling

C. MPLS LSR

MPLS networking is based on the explicit connection setup and bandwidth management with signaling protocol (CR-LDP or RSVP), routing protocol (OSPF or IS-IS, BGP), and OAM (Operation, Administration and Maintenance). Figure 9 shows the protocol organization of MPLS-LSR in the GLASS.

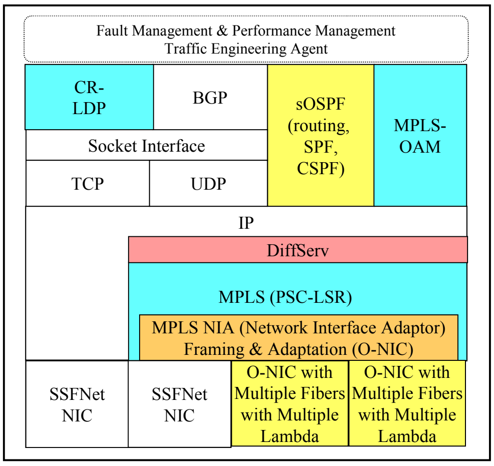

Fault Management & Performance Management
Traffic Engineering Agent
CR-LDP
BGP
sOSPF (routing, SPF, CSPF)
MPLS-OAM
Socket Interface
TCP
UDP
IP
DiffServ
MPLS (PSC-LSR)
MPLS NIA (Network Interface Adaptor)
Framing & Adaptation (O-NIC)
SSFNet NIC
SSFNet NIC
O-NIC with Multiple Fibers with Multiple Lambda
O-NIC with Multiple Fibers with Multiple Lambda

Figure 9. MPLS-LSR in the GLASS

The primary traffic engineering capability offered by MPLS is the ability to configure constraint-route label switched path (CR-LSP) routes with the traffic engineering constraints associated with the LSPs. The ability to set up explicit routes with QoS constraints can be supported by one of two signaling protocols: RSVP-TE (resource ReSerVation Protocol with traffic engineering extension) or constraint-route label distribution protocol (CR-LDP). In the NIST GMPLS Simulator, both CR-LDP and RSVP-TE are implemented. The traffic parameters TLV (type-length-value) of CR-LDP are Peak Data Rate (PDR), Peak Burst Size (PBS), Committed Data Rate (CDR), Committed Burst Size (CBS), and Excess Burst Size (EBS). Additional TE constraints, such as backup path type (1:1, 1+1, 1:N, M:N, link-disjoint or path-disjoint, SRLG (shared risk link group)-disjoint), resource color, and residual error, are defined as additional TLV in the CR-LDP signaling message and processed by the LSRs.

The constraint-based routing is supported by OSPF-TE module. The conventional OSPF only considers the aggregated cost metric of each link in the shortest path calculation; usually it considers the installed capacity and/or the physical distance of each link without any dynamic link utilization information. Thus, OSPF must be extended to support the constraint-based shortest path first (CSPF) routing in the Link State Advertisement (LSA), Link State Data Base, and the shortest path calculation algorithm. More detail explanation will be given following subsection. MPLS CR-LSP setup procedure is shown in Figure 10.

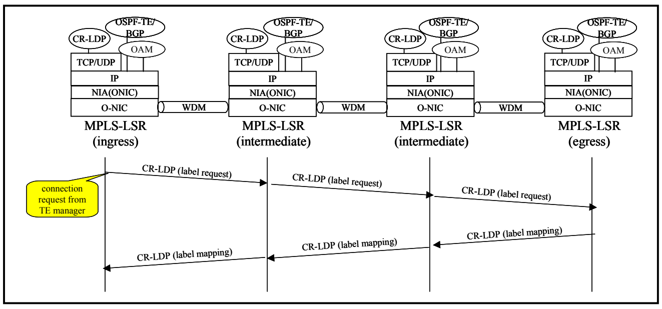


Figure 10. Constraint-based LSP setup

The ingress LSR (Label Switching Router) maintains the forwarding equivalence class (FEC) that defines the packet forwarding through the LSP (Label Switched Path) indexed by the NHLFE (Next Hop Label Forwarding Entry). The FEC is defined by following fields of IP packet header: source address prefix, destination address prefix, source and destination port range, ToS, and protocol number. The NHLFE is used to forward a labeled packet, and it contains following information: the packet's next hop, the operation to perform on the packet's label stack with a specified new label (replacing or popping), data link encapsulation in packet transmission, the way to encode label stack, and any other information needed in order to properly dispose the packet. FEC-to-NHLFE (FTN) maps each FEC to a set of NHLFE(s), and is used to forward packets that arrive unlabeled at the ingress node. According to the information of NHLFE, the packet is labeled, and forwarded. The FTN may map a particular label to a set of NHLFEs that contains more than one element; in other words, the packet flow of a FEC can be delivered by multiple LSPs. This multiple LSPs for a FEC may be useful in load balancing over multiple equal-cost paths. At the intermediate MPLS LSR, the Incoming Label Map (ILM) specifies the mapping of arriving labeled packet to a set of NHLFEs.

Using the hierarchical label stacking, a tunnel-LSP with multiple client CR-LSPs can be configured, as shown in Figure 11.

This tunnel-LSP can support various traffic engineering concepts, such as resource color, abstraction of various physical link characteristics and multiplexing. In the inter-networking between MPLS network and optical network in overlay model, the tunnel-LSP can be established across the optical domain network that is supported by an optical path.

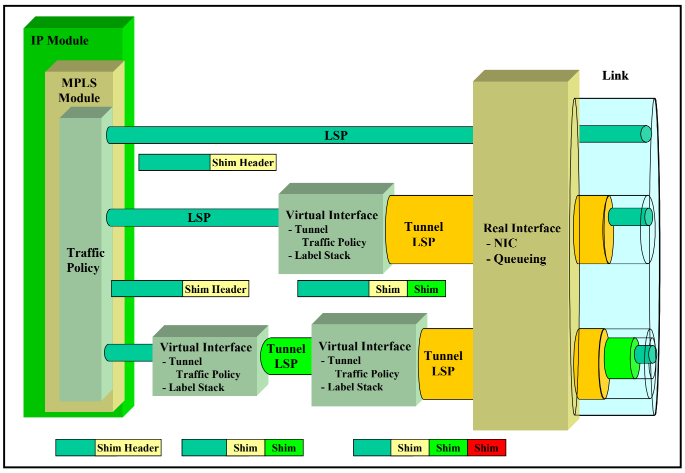


Figure 11. Hierarchical Label Stacking and tunnel LSP

Traffic policing of CR-LSP is necessary to guarantee the traffic engineering constrains associated with the CR-LSP by preventing any unauthorized overuse of link resources. To measure the bandwidth utilization by each CR-LSP, dual token bucket meter is used; a token bucket for checking peak data rate (PDR) and PBS, and another token bucket for checking committed data rate (CDR) with CBS and EBS. According to the measurement result and the bandwidth over-utilization policy, the excess packets may be discarded, or the excess packets may be tagged and the dropping-decision is deferred to the upper-level traffic policing function of the outer tunnel LSP that may allow temporal over-utilization if there is un-used available bandwidth from the under-utilization of other CR-LSPs.

MPLS packet scheduling at the output ports can be implemented with similar structure of DiffServ packet scheduler that was explained in previous section. For each CR-LSP the relative priorities are defined as setup priority and holding priority. Also, the weight associated with each CR-LSP determines the relative share of the possible excess bandwidth above its committed rate. In the GLASS, we provide priority-based scheduler, weight-based scheduler and hierarchically combined priority-based MPLS packet scheduler with partial weight-based scheduling. According to the priority and the weight of LSP, the priority-based scheduler, the weight-based scheduler, or the hybrid priority-based scheduler with partial weight-based scheduling can allow relative share of the available bandwidth to each CR-LSP.

D. WDM O-NIC (Optical Network Interface Card)

The wavelength division multiplexing (WDM) Optical network interface card (O-NIC) has multiple fibers which has multiple lambda channels respectively. Each lambda channel is used as a separated transmission path that is terminated by the IP/MPLS router or the OADM port of optical switching node. The lambda channels in a fiber can send optical frames in parallel, so the O-NIC module provides parallel packet transmission capability with individual queue for each lambda channel. The NIC module of SSFNET uses only a single transmission queue at each NIC with an abstract link to its peer node; but in the O-NIC, the parallel transmission capability has been added.

A framing and adaptation module, as shown in Figure 12, is used to provide the interface between the IP/MPLS layer protocol modules and the O-NIC module. The framing and adaptation module generates the optical frame with specified optical frame header, forwards the optical frame through a specific lambda channel, extracts the IP/MPLS packet from the arrived optical frame, and delivers the packet to the upper protocol layer. In future extension, in-channel OAM functions for the optical path will be included.

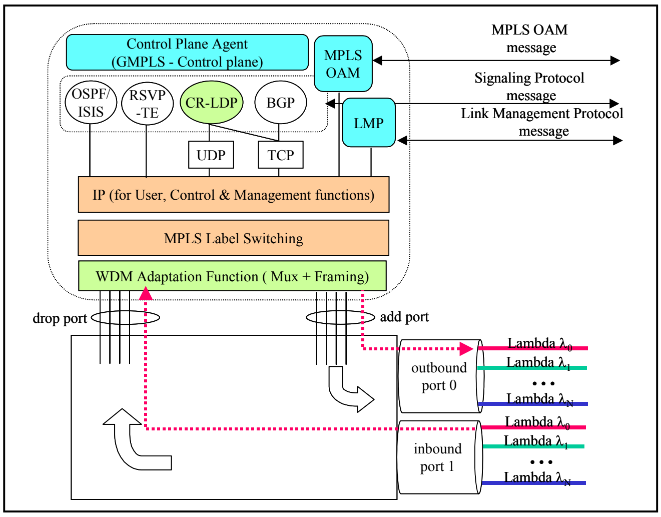


Figure 12. Optical framing and adaptation

The LMP (Link Management Protocol) supports the control channel management, link property correlation, and link connectivity verification. The LMP establishes and maintains the control channels connectivity between neighboring nodes by exchanging hello protocol message for fast keep-alive, control channel availability, and status monitoring. For link property correlation, LMP exchanges the *LinkSummary* message between the adjacent nodes to synchronize the link properties. To verify the link connectivity, in-band test messages may be transmitted over the data-bearing channel, and test status messages are transmitted over the control channel.

The lambda channel defines the optical frame structure that usually contains the frame header and trailer for channel identification, sub-channel multiplexing, and channel status management. Two major optical frame structures have been proposed: SONET-like optical frame structure and digital wrapper with simple header and trailer. The SONET-like optical frame can inherit the well-defined sub-rate multiplexing and channel management functions of SONET hierarchical multiplexing architecture. In the GLASS, to make the implementation simple, we do not provide the sub-channeling with sub-rate multiplexing at WDM O-NIC in the first stage. And, we use simplified optical frame structure as shown in Figure 13. In this simple optical frame header, there is no channel management information, and no sub-channeling. In the current implementation of the GLASS, the fiber ID (16-bit) and lambda ID (16-bit) are used as the 32-bit generalized label for optical network in GMPLS-based signaling. This GMPLS label is locally unique.

The parameters of optical interface, such as number of fibers in a O-NIC, the number of lambda channels in a fiber, and the signal type (or transmission rate), are defined in the DML configuration file. The parameters related to the TE-extensions, such as residual error rate of the fiber link, SRLG identifier, and resource color, are also defined in the DML file.

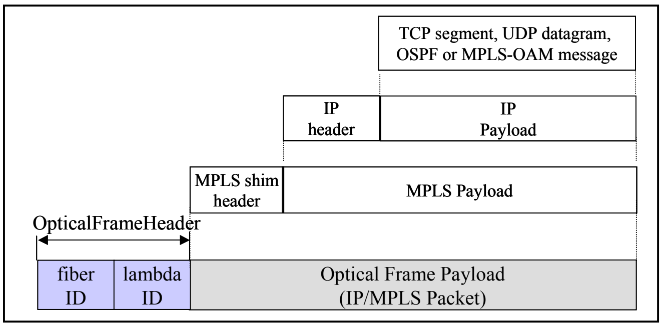

TCP segment, UDP datagram, OSPF or MPLS-OAM message
IP header
IP Payload
MPLS shim header
MPLS Payload
OpticalFrameHeader
fiber ID
lambda ID
Optical Frame Payload (IP/MPLS Packet)

Figure 13. Optical Frame Structure in simulator

E. OXC/OADM optical switch

OXC/OADM optical switch node provides basically wavelength routing by forwarding the arrival optical frame at a lambda channel of an input fiber link in an O-NIC to the specific output lambda channel according to the switching table. OXC node may have lambda conversion (or wavelength translation) function that translates any input wavelength into different output wavelength signal. Some limitations on the lambda conversion, such as the number of converters and the range of lambda conversion, can be specified at each optical switching node. Optical switching is implemented by all-optical devices or by optical-electrical-optical devices. In the simulation of optical networking, the detailed implementation methods of optical switching are not modeled in detail; the optical switching is simply controlled by the switching table. Figure 14 shows a typical OXC/OADM node model.

As explained in previous section, each fiber link contains multiple wavelength channels whose bandwidths are 2.5 Gbps ~ 10 Gbps. In the GLASS, the lambda 0 channel of each fiber link is used for the control plane function, such as CR-LDP signaling, LMP, OSPF and BGP. The lambda 0 channels are dropped at each OXC/OADM node to be connected to the upper layer control protocols. Between two peer nodes, multiple lambda 0 channels are available, and at least one of the lambda 0 channels is selected as an active signaling channel. All lambda 0 channels are managed by LMP.

OXC node can support waveband switching and fiber bundling. In waveband switching, as set of contiguous wavelengths is switched together to a new waveband as a unit, while in fiber bundling, a set of fiber links are switched together to a set of fiber links as a bundle. In the current implementation of NIST GLASS, to provide more flexibility, we support only the optical path setup for lambda channel, and do not provide waveband switching and fiber bundling. So, each lambda channel is set up one by one.

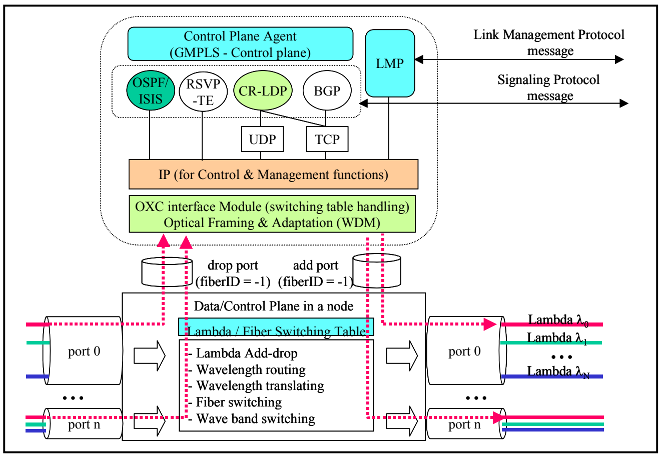

Control Plane Agent (GMPLS - Control plane)
LMP
Link Management Protocol message
Signaling Protocol message
OSPF/ ISIS
RSVP -TE
CR-LDP
BGP
UDP
TCP
IP (for Control & Management functions)
OXC interface Module (switching table handling)
Optical Framing & Adaptation (WDM)
drop port (fiberID = -1)
add port (fiberID = -1)
Data/Control Plane in a node
Lambda / Fiber Switching Table
- Lambda Add-drop
- Wavelength routing
- Wavelength translating
- Fiber switching
- Wave band switching
port 0
port n
port 0
port n
Lambda λ0
Lambda λ1
Lambda λN

Figure 14. OXC/OADM node model

The transport network assigned addresses (TNAs) are carried in O-UNI signaling messages to uniquely identify the endpoints of a connection. When a TNA corresponds to more than one physical or logical link, a decision has to be made on which physical interface the connection must be terminated. In O-UNI 1.0, two addressing schemes - IPv6-based and NSAP-based - are proposed. In IPv6-based addressing, 128-bit address with unique mapping of an IPv4 is used. In NSAP-based addressing, 160-bit (20 bytes) address with unique mapping of IPv4 address is used. Since both proposals assume unique mapping of IPv4 address in O-UNI, we implemented IPv4-based addressing in the GLASS.

Two addressing schemes, numbered ID addressing and un-numbered ID address, are available in GMPLS addressing [51]. In *numbered ID addressing*, unique IP address is assigned to each fiber, while the lambda ID is defined as an additional label. In *un-numbered ID addressing*, unique IP address is assigned to each O-NIC, and fiber ID and lambda ID are used as the additional label. Considering a large scale Internet networking, the GLASS supports only the un-numbered ID addressing where each O-NIC in an OXC/OADM node has its unique IP address. The OXC node ID is defined as one of the O-NIC IP address as a delegation ID.

The fault detection capability of optical link failure at OXC node is very important for fast fault restoration. The real implementations of the fault detection mechanism may differ according to the functionality of O-NIC, the optical switching and wavelength conversion mechanism in the OXC, and the in-band optical OAM functions. In the current GLASS, we assume the O-NIC has very simple optical link failure detector that can detect loss of light (LOL) at each lambda channel immediately and notify the detected fault to the traffic engineering agent (TE-agent) in the OXC node. For the simulation of optical link failure, DML file can specify the link failure at specific simulation time, and a timer in the optical link module is used to trigger the fault event. When TE-agent receives the fault notification with detailed information of the fiber link(s) and lambda channel(s), it asks the CR-LDP signaling module to send fault notification message to both the ingress OXC node and egress OXC node of the optical path. The detail fault restoration function with back optical path will be explained in the following section.

F. Traffic Engineering (TE) extensions to CR-LDP, OSPF and BGP

In order to provide the traffic engineering capability in the network simulator, the signaling, routing, link management and network management modules should support the TE-related parameters and mechanisms. Especially to guarantee the user-requested QoS, the signaling and routing must provide constraint-based routing and resource allocation at the connection establishment. Constraint-based routing refers to a class of routing systems that compute routes through a network subject to satisfaction of a set of constraints and requirements [21]. In general, the constraint-based routing may also seek to optimize the overall network performance while minimizing costs. The routing constraints are imposed by the network's administrative policies, or by the connection request for the user application as service level specification. Constraints usually include the bandwidth, hop count, end-to-end delay, delay variation limit (jitter), residual error rate, and policy-related parameters such as resource class and SRLG (Shared Risk Link Group) identifier.

LDP (Label Distribution Protocol) has been extended to CR-LDP (Constraint-based LDP) to support constraint-based routing and traffic engineering in MPLS network [18]. RSVP-TE is another signaling protocol used in MPLS with TE-extensions. In the GLASS, both the CR-LDP and the RSVP-TE are implemented, and can be configured as the signaling protocol module.

OSPF-TE includes the extended link attributes for traffic engineering by using opaque LSA (link state advertisement) [24]. The link TLV (type/length/value) of Traffic Engineering LSA contains traffic engineering metric (link metric for traffic engineering purposes), maximum bandwidth (link capacity), maximum available bandwidth (if over-subscription is allowed), unreserved bandwidth (amount of bandwidth not yet reserved at each of the eight priority levels), and resource class/color (administrative group membership for this link). The additional link attributes are used to build an extended link state database that is used in monitoring & reporting the extended link status, local constraint-based source routing, and global traffic engineering. In current GLASS implementation, the SSFNet's static OSPF has been extended to support TE link state parameters. Figure 15 shows the operations of OSPF-TE. OSPF-TE firstly collects the TE link state information of the network by visiting each LSR or OXC nodes, and builds the TE link state DB. This TE link state DB contains the detailed link state information of the pre-defined TE parameters on the link and its current status. For the constraint-based routing, this TE link state DB is pruned with the constraints specified in the connection request. Based on the pruned constraint-based link state DB, the shortest path can be calculated. In current GLASS implementation, the predefined TE link parameters are specified in the DML file; and the constraints of the LSP setup request are also specified in the DML file.

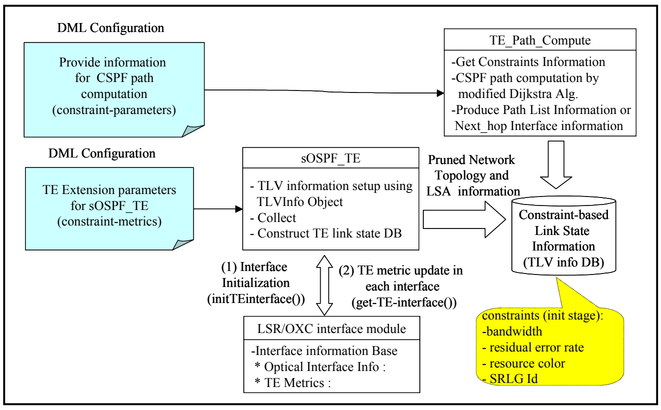

DML Configuration
Provide information for CSPF path computation (constraint-parameters)
TE_Path_Compute
-Get Constraints Information
-CSPF path computation by modified Dijkstra Alg.
-Produce Path List Information or Next_hop Interface information
DML Configuration
TE Extension parameters for sOSPF_TE (constraint-metrics)
sOSPF_TE
- TLV information setup using TLVInfo Object
- Collect
- Construct TE link state DB
Pruned Network Topology and LSA information
Constraint-based Link State Information (TLV info DB)
(1) Interface Initialization (initTEinterface())
(2) TE metric update in each interface (get-TE-interface())
LSR/OXC interface module
-Interface information Base
* Optical Interface Info :
* TE Metrics :
constraints (init stage):
-bandwidth
- residual error rate
- resource color
- SRLG Id

Figure 15. OSPF-TE functional architecture

In order to provide MPLS traffic engineering across multi-area (or multi-domain) networks, the OSPF area border router (OSPF-ABR) or the border gateway router with BGP should provide some mechanism to exchange the TE-related information as well as reachability information [45-46]. The extensions of BGP-4 protocol to support the inter-area/domain TE-related link state information is under discussion in IETF [44-49]. In current version of GLASS, the simulation functions of the multi-domain/area are not provided.

G. GMPLS-based O-UNI and O-NNI

For the establishment of an end-to-end MPLS LSP with TE constraints, the MPLS signaling protocol modules in each LSR should interact each other with the support of OSPF-TE module. In the MPLS layer network without any circuit-switched (or lambda/fiber switched) transit network, the TE link state information DB is constructed by the OSPF-TE module, and is used in the constraint-based shortest path first (CSPF) routing. If any circuit/lambda-switched network is used as the transit network of MPLS layer network, the connection-oriented path in the circuit/lambda-switched network domain must be established either in on-demand manner or in pre-establishment manner.

In peer-to-peer inter-networking model, the circuit/lambda-switched path is regarded as a special case of LSP that can be inter-connected with packet-switched LSP. So, on-demand connection setup is required across the packet switching network domain and circuit switching network domain. The signaling entities of MPLS LSR and OXC-LSR interact in peer-to-peer manner, and a single instance of OSPF-TE can support both MPLS LSRs and OXC-LSRs with optional area concept.

In server-client overlay networking model, the circuit/lambda-switched path is used as a transit link in the IP/MPLS layer network, and interconnects two IP/MPLS LSR nodes. This is the same concept of IP-over-ATM networking, and the MPLS layer network and the optical domain network may be owned and managed separately by different network operators. In this overlay model, there are two separated OSPF-TE instances: one for the MPLS layer network domain and the other one for the optical network domain. The establishments of the transit links in the circuit/lambda-switched network domain are determined by the traffic engineering manager of the MPLS layer network according to the traffic estimation among end users and the network utilization status.

In the current implementation of the GLASS, we support the overlay networking with O-UNI and O-NNI signaling based on CR-LDP and OSPF-TE. The establishments of the MPLS transit links are specified in the DML file, or initiated by the traffic engineering manager that manages the backbone trunk LSPs. Figure 16 shows the O-UNI signaling and O-NNI signaling in overlay inter-networking model. In this scenario, O-UNI is initiated by an MPLS-LSR that requires a trunk LSP to its neighbor MPLS-LSR, which is another client node of the optical domain network. So, the MPLS-LSR should have CR-LDP signaling function for MPLS layer network and also the O-UNI signaling client function.

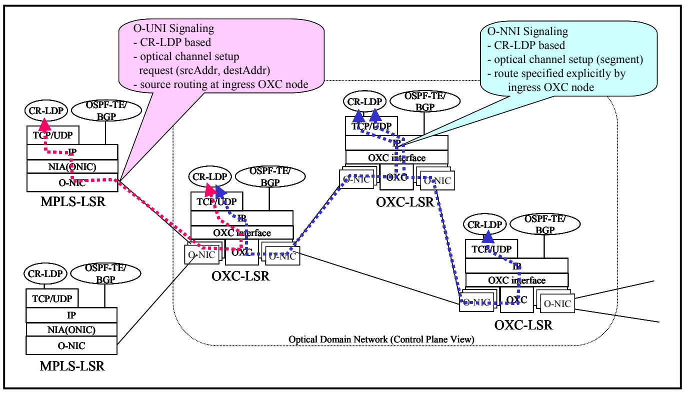


Figure 16. O-UNI and O-NNI signaling

The ingress OXC-LSR in the optical network domain receives the connection setup request through the O-UNI signaling, computes the shortest path to the destination (egress OXC-LSR) with the support of OSPF-TE module, and sends O-NNI signaling message (CR-LDP label request message) along the computed path. The intermediate OXC-LSR, when it receives an O-NNI signaling message, reserves the required resource (wavelength channel) according to the connection setup request.

The O-NNI signaling message is propagated to the egress OXC-LSR node that sends O-UNI signaling message to the destination MPLS-LSR to set up a lambda channel. When the destination MPLS-LSR accepts the request of the lambda channel establishment, the egress OXC-LSR allocates the lambda channel, and sends *CR-LDP label mapping* message back to the ingress OXC-LSR along the path. When each intermediate OXC-LSR receives CR-LDP label mapping message, it allocates the reserved resources (lambda channel). When the ingress OXC-LSR receives the CR-LDP label mapping message, it finally allocates the lambda channel between the connection requesting MPLS-LSR and itself, and replies the connection establishment. It any OXC-LSR cannot reserve or allocate the request resource, this connection setup procedure is terminated, and is crank-backed all the way to the ingress OXC-LSR that must re-compute the route for the optical path, and re-initiate the establishment procedure of the optical path.

For the optical path setup among the client nodes of optical domain network, each access optical link between client nodes (e.g. MPLS-LSR) and the OXC-LSR has pre-allocated address on it. As we briefly explained in previous section, the GLASS is supporting the un-numbered addressing for the optical links; each O-NIC is assigned with a unique IP address, and the fiber ID and lambda ID are used as the additional label that is unique within the O-NIC. The fiber ID (16-bit) and lambda ID (16-bit) are used as the 32-bit generalized label for the lambda channel that is a LSP in the optical network domain.

H. MPLS OAM (Operation, Administration and Maintenance)

In order to keep the MPLS LSP in good operational status, the performance monitoring and the fault management for fast restoration are essential. The performance monitoring should continuously monitor the packet delivery performance of the LSP according to the agreed traffic parameters, such as bandwidth, end-to-end packet transfer delay and jitter (delay variation) boundary. The fault management function should detect the occurrence of any fault condition in each protocol layer as soon as possible. When a link/node fault occurs in a specific layer, the fault management function must be able to minimize the spread of the effect of the fault to the upper layer, and swiftly switch the service traffic of the affected path to the alternative path if possible. A fast restoration is essential to increase the reliability of the quality of service of real time multimedia applications. For the fast restoration, prompt fault detection and fault notification functions are essential.

In GLASS, we implemented the MPLS OAM functions for the user data channel's performance monitoring and the fast fault restoration of LSP. For the efficient MPLS network management, we designed the MPLS OAM functions with full consideration of the network management architecture based on the TINA (Telecommunications Information Networking Architecture) network management architecture. Performance monitoring of LSP, fault detection, fault notification and fault localization functions are performed with the proposed MPLS OAM functions.

In the implementation of MPLS OAM functions, we use the mechanism of adding an LSP with reserved label value of 15 to distinguish the MPLS OAM packets from the user data packet. The return path of the MPLS OAM function is maintained using the reverse path of the bi-directional LSP. The results of performance monitoring and the backward fault notification OAM messages are delivered to the ingress LER through the return path. For fast fault detection and notification, we use both MPLS signaling and MPLS OAM. When a link/node failure has been detected by the lower protocol layer in the intermediate LSR the fault notification is done by signaling function, since the MPLS OAM packet transmission by the intermediate LSR is very difficult especially when the LSP is encapsulated by multiple label stacking. When the severe performance degradation is detected or the link/node failure is detected by the periodic performance monitoring mechanism by the egress LER, than the fault notification is done by the MPLS fault notification OAM packet from the egress LER to the ingress LER.

The loop-back test OAM function is used to identify the exact location of the failure. The loop-back test OAM function is performed after the protection switching or the fault restoration of the user data traffic. The loop-back test is optionally executed by the request from the network management system (NMS) before the erred working LSP is tear down. In this state, the erred working path is still working partially along the established path, so the ingress LER can send probe loop-back OAM packet through the LSP with special EXP field in the shim header. Each intermediate LSR can understand this loop-back OAM, and can report the reception of the loop-back OAM packet via the MPLS signaling function to the ingress LER. When the proper reply is not received from any LSR on the LSP route, the ingress LER can determine the location of the link/node fault.

I. Integrated Fault Restoration

Fast and efficient fault restorations in the Internet and the optical network are the challenging problems that have been researched aggressively by various research works [52, 53]. But, the integrated fault restoration in the IP/MPLS-over-Optical network has been studies only recently because of the immaturity of the control plane of the optical network and the MPLS-OAM functions.

In order to support the research works on the integrated fault restoration in the next generation optical Internet, the NIST GLASS provides simplified example models of the fault restoration in optical domain network and MPLS domain network. Figure 17 shows the overall procedure of the fault restoration in the optical domain.

In this scenario, the working optical path is protected with its 1:1 backup optical path from the ingress OXC-LSR and egress OXC-LSR. We assume the protection of the working light path is requested in the O-UNI signaling by a specific field of optical path protection type. The optical link fault is emulated by the O-NIC or NIC, and the programmed fault occurrence link and the programmed fault occurrence time are specified in the DML file.

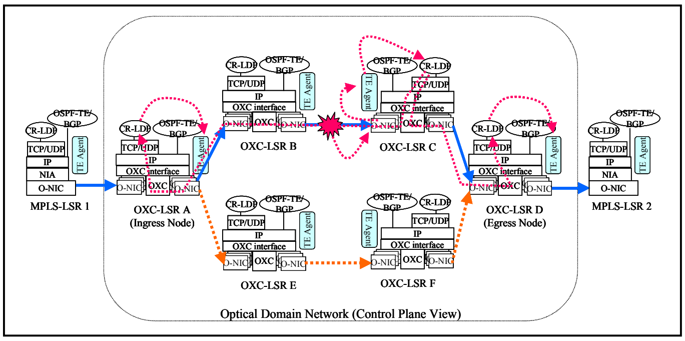


Figure 17. Integrated Fault Restoration

We assume that each NIC/O-NIC has the capability to detect Loss of electrical/light Signal (LOS). At the initialization stage of network simulation, each NIC/O-NIC module reads in the fault emulation specification from the DML file and sets a timer for the scheduled fault event. When the timer expires, NIC/O-NIC receives the notification of the link failure, blocks any packet transmission, and reports the fault detection to the TE agent that is in charge of the resource management operation in the packet/optical switching node. The TE agent requests the CR-LDP signaling module to send fault notification message to the ingress OXC-LSR and the egress OXC-LSR that re-route the data packet traffic from the erred working path to the backup path. The same mechanism is implemented in the MPLS layer network with NIC.

The dynamic error detection scheme with optical link OAM messaging is not supported currently. The integrated fault restoration mechanism with the restoration capability in both the optical layer network and the MPLS layer network can be implemented by using a traffic engineering manager that interacts with the TE agent in each switching node, supports the traffic engineering of the overall network with the network performance management and the network fault management. Figure 18 shows the overview of the traffic engineering manager.

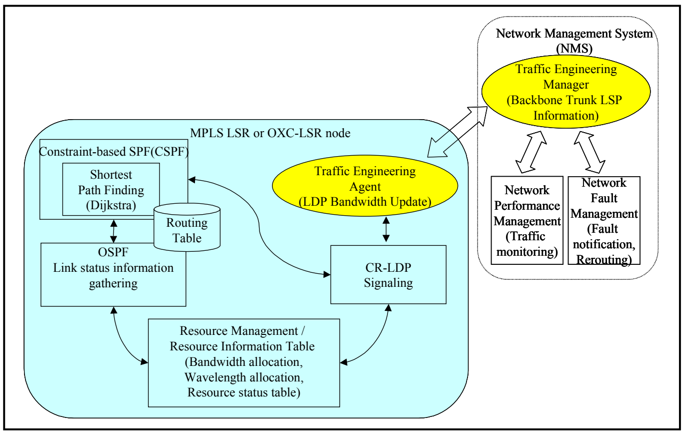

Network Management System (NMS)
Traffic Engineering Manager (Backbone Trunk LSP Information)
Network Performance Management (Traffic monitoring)
Network Fault Management (Fault notification, Rerouting)
MPLS LSR or OXC-LSR node
Constraint-based SPF(CSPF)
Shortest Path Finding (Dijkstra)
Routing Table
OSPF Link status information gathering
Traffic Engineering Agent (LDP Bandwidth Update)
CR-LDP Signaling
Resource Management / Resource Information Table (Bandwidth allocation, Wavelength allocation, Resource status table)

Figure 18. Traffic Engineering Manager

J. User Friendly Simulation Result Output Function

The output format of the network simulation results may be various according to the purpose of the simulation and the specific interest. For example, the network resource utilization will be best shown with a graph of traffic flows at each link, while the response time analysis in the integrated fault restoration may be shown by simple data list for each event handling.

The SSFNet simulator provides a simple log file mechanism to generate the output result, and the output file format is specified by the programmer who must insert print-out command at the specific location of source code.

In the current version of the NIST GMPLS Simulator, we support the Excel file format from which various graphs and charts can be generated according to the user's specific purpose. Also, the output of the simulation results can be controlled by the DML file, so as to support any network simulation with simple modification of the DML file without any source code modification.

V. Conclusion

In this paper, we explained the design goal and implementation architecture of a GMPLS-based Optical Internet simulator, called *GLASS (GMPLS Lightwave Agile Switching Simulator.* The NIST GLASS has been developed to support the R&D works in the area of Next Generation Internet (NGI) networking with GMPLS-based WDM optical network, and Internet traffic engineering with DiffServ-over-MPLS. It supports the discrete-event simulations of various DiffServ packet classification and per-hop-behavior (PHB) processing with class-based-queuing, MPLS traffic engineering, GMPLS-based signaling for WDM optical network, link/node failure model, and fast restoration from an optical link failure.

The NIST GLASS is implemented on the SSFNet (Scalable Simulation Framework Network) simulation platform. It has been designed and implemented with open interfaces to support future expansion or replacement of protocol modules by users. It also provides DML description input file interface to support the users' flexible definition/modification of simulation parameters and configuration of protocol modules.

The GLASS has been designed and implemented in modular structure so as to support the user to configure his/her major interests of simulation. Based on the modular structure, various simulation scenarios with any combination of the basic Internet networking block, DiffServ block, MPLS networking block, and optical networking block are possible.

Several examples of the simulation of the GMPLS-based optical Internet networking are provided at the homepage to show the functionality of the simulator. The example simulations include constraint-based routing with OSPF-TE, RWA (routing, wavelength and wavelength assignment) for optical network with CR-LDP and OSPF-TE, MPLS LSP setup and traffic & QoS measurement, hierarchical label stacking with MPLS tunnel LSP, DiffServ-over-MPLS, and fast restoration in IP/MPLS-over-Optical network.

References

[1] Antonio Rodrigues-Moral et al., "Optical Data Networking : Protocols, Technologies, and Architectures for Next Generation Optical Transport Networks and Optical Internetworks," Journal of Lightwave Technology, Vol. 18, No. 12, December 2000, pp. 1855~1870.
[2] Jeremy Lawrence, "Designing Multiprotocol Label Switching Networks," IEEE Communications Magazine, July 2001, pp. 134~142.
[3] Ayan Banerjee et al., "Generalized Multiprotocol Label Switching (GMPLS) : An Overview of Routing and Management Enhancements," IEEE Communications Magazine, July 2001, pp. 144~151.
[4] Panos Trimintzios et al., "A Management and Control Architecture for Providing IP Differentiated Services in MPLS-Based Networks," IEEE Communications Magazine, May 2001, pp. 80~88.
[5] Xipeng Xiao et al., "Traffic Engineering with MPLS in the Internet," IEEE Network, March/April 2000, pp. 28~33.
[6] The Network Simulator (ns), http://www.isi.edu/nsnam/ns/
[7] JavaSim, http://eepc117.eng.ohio-state.edu/javasim
[8] SSF (Scalable Simulation Foundation), http://www.ssfnet.org/exchangePage.html
[9] OPNET Modeler, http://www.opnet.com/
[10] NIST GMPLS Simulator (GLASS), http://www.antd.nist.gov/glass/
[11] Antonio Rodrigues Moral, Paul Bonenfant, and Murali Krishnaswamy, "The Optical Internet : Architectures and Protocols for the Global Infrastructure of Tomorrow," IEEE Communications Magazine, July 2001, pp. 152~159.
[12] Eve L. Varma et al., "Architecting the Services Optical Network," IEEE Communications Magazine, September 2001, pp. 80-87.
[13] IETF Draft, IP-over-Optical Networks : A Framework, Expires May 14, 2001.
[14] IETF Draft, IP over Optical Networks : A summary of Issues, S. Seetharaman et al., April 2001.
[15] IETF Draft, Generalized MPLS – Signaling Functional Description, October 2000.
[16] User Network Interface (UNI) 1.0 Signaling Specification, Optical Internetworking Forum (OIF), April 15, 2001.
[17] IETF Draft, Generalized MPLS Signaling – RSVP-TE Extensions, Nov. 2000.
[18] IETF Draft, Constraint-based LSP Setup using LDP, February 2001.
[19] IETF Draft, Link Management Protocol (LMP), Jonathan P. Lang et al.., Sept. 2001.
[20] IETF RFC 2702, Requirements for Traffic Engineering over MPLS, Awduche et al., September 1999.
[21] IETF Draft, A Framework for Internet Traffic Engineering, July 2000.
[22] ITU-T Draft Recommendation Y.1541, The network performance objectives for IP-based services, Oct. 2001.
[23] IETF Draft, Traffic Engineering Extensions to OSPF, August 2001.
[24] IETF Draft, The OSPF Opaque LSA option, July 1998.
[25] IETF RFC 2676, QoS Routing Mechanisms and OSPF extensions, G. Apostolopoulos et al., August 1999.
[26] IETF Draft, Alternative OSPF ABR (Area Border Router) Implementation, Alex Zinin et al., February 2001.
[27] IETF Draft, IS-IS extensions for Traffic Engineering, Sept. 2000.
[28] Project report, Functional Architecture Definition and Top Level Design, Danny Goderis, Editor, TEQUILA (Traffic Engineering for Quality of Service in the Internet, at Large Scale), September 2000.
[29] IETF Draft, OAM Functionality for MPLS Networks, Feb. 2001.
[30] IETF RFC 2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, K. Nichols et al., December 1998.
[31] IETF Draft, "MPLS Support of Differentiated Services," Francois Le Faucheur et al., April 2001.
[32] IETF Draft, "Requirements for support of Diff-Serv-aware MPLS Traffic Engineering," June 2001.
[33] IETF Draft, "MPLS Support of Differentiated Services using E-LSP," S. Ganti et al., April 2001.

[34] IETF RFC 2836, “Per-Hop-Behavior Identification Codes,” S. Brim et al., May 2000.

[35] IETF Draft, “An Expedited Forwarding PHB (Updates RFC 2598),” Bruce Davie et al., April 2001.

[36] IETF RFC 2597, “Assured Forwarding (AF) PHB Group,” J. Heinanen et al., June 1999.

[37] IETF RFC 2638, “A two-bit Differentiated Services Architecture for the Internet,” K. Nichols et al., July 1999.

[38] IETF RFC 2309, “Recommendations on Queue Management and Congestion Avoidance in the Internet,” B. Braden et al., April 1998.

[39] IETF Internet Draft, “Alternative Technical Solution for MPLS DiffServ TE,” Jerry Ash et al., July 2001.

[40] IETF RFC 2697, “A Single Rate Three Color Marker,” J. Heinanen et al., September 1999.

[41] IETF RFC 2698, “A Two Rate Three Color Marker,” J. Heinanen et al., September 1999.

[42] IETF Draft, “Management Information Base for the Differentiated Services Architecture,” F. Baker, K. Chan, A. Smith, August 2001.

[43] IETF Draft, “An Informal Management Model for DiffServ Routers,” Y. Bernet, S. Blake, D. Grossman, A. Smith, Feb. 2001.

[44] IETF RFC 2547, BGP/MPLS VPNs, E. Rosen et al., March 1999.

[45] IETF Draft, Providing Quality of Service Indication by the BGP-4 Protocol: the QOS_NLRI attribute, July 2001.

[46] IETF RFC 2842, Capability Advertisement with BGP-4, R. Chandra et al., May 2000.

[47] IETF Draft, A BGP/GMPLS Solution for Inter-domain optical networking, Yangguan Xu et. at, July 2001.

[48] IETF Draft, Multi-area MPLS Traffic Engineering, Kireeti Kompella et al., Expiration Date May 2002.

[49] IETF Draft, BGP/GMPLS Optical VPNs, Hamid Ould-Brahim et al., July 2001.

[50] IETF Draft, Link Bundling in MPLS Traffic Engineering, Kireeti Kompella et al., expiration date March 2002.

[51] IETF Draft, Signaling Unnumbered Links in CR-LDP, Kireeti Kompella et al., expiration date March 2002.

[52] Ornan Gerstel and Rajiv Ramaswami, “Optical Layer Survivability: A service Perspective,” IEEE Communications Magazine, March 2000, pp. 104-113.

[53] Yinghua Ye et al., “On Joint Protection/Restoration in IP-centric DWDM-based Optical Transport Networks,” IEEE Communications Magazine, June 2000, pp. 174-183.