Lambda GLSP Setup with QoS Requirements in Optical Internet

SuKyoung Lee and David Griffith
National Institute of Standards and Technology
100 Bureau Drive, Stop 8920, Gaithersburg, MD 20899

Abstract— **GMPLS is currently being specified as the control plane for next generation optical networks. While one merit of GMPLS is that network operators are able to control the traffic of Internet Protocol (IP) and thus offer performance superior to that of traditional Internet Gateway Protocols (IGP), it has been a challenge to develop a efficient Traffic Engineering (TE) technique to satisfy the different traffic requirements of end-users in GMPLS domain. Several TE techniques have been thought of by creating a set of Generalized Label Switched Paths (GLSPs) in the optical network.**

In this paper we present a technique for establishing GLSP in optical Internet, that support Quality of Service (QoS) considering the traffic flows with delay QoS requirements. This approach accounts for the processing delay at the network layer that may be encountered in optical nodes that terminate lightpaths and perform some types of traffic grooming. Additionally, we extend the QoS routing framework by defining additional extensions that can be used in conjunction with Shared Risk Group (SRG) identifiers to reserve backup paths that do not share network resources with the working paths that they are protecting. We provide some simulation results to demonstrate the effectiveness of our routing algorithms.

I. Introduction

Recent years have seen a growing need to support applications that require a high QoS on networks that use the IP, voice traffic being a classic example. The response to this need has been the development of a range of tools, such as Integrated Services (IntServ) and Differentiated Services (DiffServ), to support various levels of QoS. In addition, it is now possible to implement TE in IP networks using tools such as Multi-Protocol Label Switching (MPLS) [1]. Using TE extensions to OSPF in combination with MPLS, it is possible to route connections across the network in order to optimize network utilization, avoid islands of stranded bandwidth, and balance load.

The rapid growth in data traffic currently being experienced is fueled by the Internet, posing new challenges for transport network providers. To meet the bandwidth demand of the Internet, it is natural to use optical technologies with Wavelength Division Multiplexing (WDM) to offer the capability of building very large wide area networks with throughput of the order of terabits per second.

Recently, with the advent of new traffic engineering protocols like MPLS, there has been considerable activity in several standards groups to integrate MPLS and WDM networking technologies into a unified structure for optical Internet [2]. This extended version of MPLS that is so-called GMPLS will allow many carriers to deploy optical Internets where lambdas (wavelengths) are treated as very low level point to point links for the transmission of packets between high performance routers. In this optical Internet, optical switches translate label assignments into corresponding wavelength assignments and setup GLSP using local control interfaces to switch devices. Subsequent to GLSP setup, no explicit label lookup/processing operations are performed by Optical Labeling Switched Routers (O-LSRs).

One of the benefits of GMPLS is that it supports TE by allowing the node at the network ingress to specify the route that a GLSP will take by using explicit routing (ER). An explicit route is specified by the ingress as a sequence of hops that must be used to reach the egress, which is different from the hop-by-hop routing that is usually associated with packet-switch capable networks. ER features can also be used to facilitate QoS support for multiple class of services in GMPLS networks (e.g. DiffServ). There have been several efforts directed towards using QoS information to make routing decisions, but their focus is different from ours. A method for supporting QoS using the IETF's Open Shortest Path First (OSPF) routing protocol can be found in [3]. This approach advertises available bandwidth and delay metrics in OSPF Link State Advertisements (LSAs), but does not use the delay metric in computing paths. Some centralized routing schemes have been proposed, notably that in [14], which seek to find the transiting path whose delay is less than a prescribed limit and whose bandwidth is greater than a specified lower bound. That approach uses Dijkstra's algorithm to find a suitable path but does not specifically account for delays at layer 3 or for relative throughput, as our algorithm does.

Routing to support QoS requirements could be also applied to optical networks [5]. QoS routing selects a path offline for each flow or connection to satisfy diverse performance requirements and optimize resource usage. Even though network resource is still available, uneven distribution of traffic prevents from meeting QoS constraints, that is, leads to worse delay performance in terms of delay QoS. In this case, TE arranges traffic flows so that congestion caused by uneven network utilization can be avoided.

In this paper, we propose a mechanism to to provide better delay performance and improve network utilization in the optical network. We do this by using a linear programming approach that seeks to minimize the total path delay, including delay introduced by packet processing at layers above optical. While other researchers, specifically [6], have introduced linear programs that account for the bottleneck effect at layer 3, their approach seeks to maximize total network throughput, while ours is focused on meeting QoS requirements.

The optimization routine that we are proposing will take path delay, including queueing delay at layer 3, into account. In opti-

This work was supported, in part, by the National Institute of Standards and Technology (NIST), the Advanced R&E Activity (ARDA), the Laboratory for Telecommunications Sciences (LTS) MENTER project, the Defense Advanced Research Projects Agency (DARPA) Fault Tolerant Networks (FTN) program, and the National Communications System (NCS).

cal Internet using lambda labeling, processing at layer 3 would constitute a significant bottleneck whose impact we wish to minimize. In transparent optical networks where it is possible to create lightpath connections between every pair of edge routers, this is not an issue. But in large networks with thousands of devices at the edge, creating a virtual topology that is a full mesh is impractical. Thus, we can expect that intermediate nodes with layer 3 functionality will exist in a lambda labeling network (most likely at the edges of optical subnetworks in the core). Thus it would be useful to account for the impact of layer 3 processing when the ingress computes the explicit route. The layer 3 processing overhead at intermediate nodes is accounted for in the proposed path set-up algorithm.

The rest of paper is structured as follows. In section II we introduce the background. We describe our algorithm in section III where an integer linear program (ILP) based solution is used. In section IV we present simulation results, and finally in section VI conclusion is drawn.

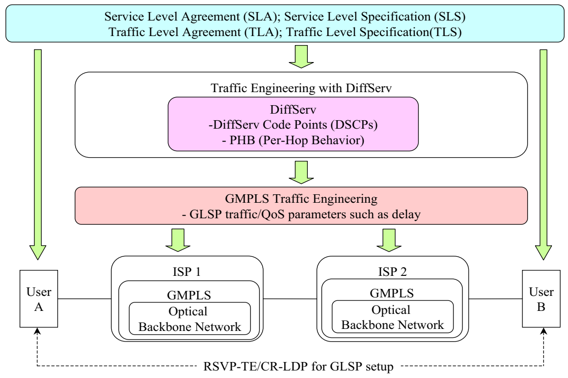

Fig. 1. Traffic Engineering with QoS Requirements

II. Background

Core optical network consists of the Optical Cross-Connects (OXCs) and fiber links interconnecting OXCs. Each OXC can route an input wavelength to an output wavelength. These OXCs, in most of the cases, can preclude the electronic processing, and thus a optical connection can be established between edge nodes. In lambda labeling network, GMPLS control plane is attached onto OXC devices and treats them as GMPLS nodes, hereafter called as O-LSR nodes, allowing GMPLS signaling to compute and set up GLSPs between O-LSR nodes in a similar way as MPLS LSPs.

O-LSRs can be viewed as a combination of a router and an OXC. The router component is responsible for all the layer 3 functions such as addressing, routing, and global topology discovery. It is also responsible for optimizing network performance, which can be carried out via TE with QoS support, management of optical resources (i.e. wavelength assignment in coordination with the optical channel sublayer), and restoration. That is, the most important advantage of GMPLS turns out to be its ability to do TE of IP networks although it was originally devised with other goals such as faster forwarding of IP datagrams. Even if GMPLS is very useful as protocols for establishing traffic engineered paths through optical Internet, many researchers have recently begun working on the mechanisms of computing these paths [7]. Therefore, the objective of this paper is to formulate the TE problems in mathematical form to motivate more work towards the consideration of multiple QoS classes. As can be seen in Fig. 1, the wavelengths can be assigned to the GLSPs in accordance with QoS class [5]. GMPLS would be applicable to differentiated services based networks as the growing trend is to use differentiated services in the edge O-LSRs and lightpaths in the core. In this paper, ingress O-LSRs determine the GLSPs that IP traffic flows will be routed through, in terms of the QoS requirements such as the maximum acceptable delay. Additionally, this path computation based on delay requirement can be also applied to establishing a backup path with reservations every time a primary path is established across the network. Seeing the objective of TE in terms of priorities, although we consider only a single level of priority for delay QoS in this paper, the proposed approach naturally can extend to multiple priority levels. As a consequence, the performance of the QoS service classes is oblivious to the presence of the delay-sensitive traffic.

Reliable network operation is another important aspect of TE. Failure recovery scenarios must be designed to ensure continuity of service following network impairments. Therefore, the operational capability must exist to reroute traffic through the remnant capacity when failures occur. While this rerouting procedure should make more effective use of the residual post-failure capacity, it should establish a backup GLSP to still satisfy the QoS requirement even after the traffic is recovered onto the backup path.

As a signaling protocol to support the QoS, there are two methods in GMPLS. . One uses TE extensions to the Resource Reservation Protocol (RSVP-TE) [8],[9] and the other one utilizes extensions to the label distribution protocol (LDP) called Constraint-based Routing LDP (CR-LDP) [8], [10]. IETF have been developing both signaling protocols. Either RSVP-TE or CR-LDP can be used allowing a GLSP to be explicitly specified across the optical core (Fig. 1) since both signaling protocols were also recently extended to support GMPLS.

III. GLSP Set-up Algorithm

The main problem of the proposed algorithm is to establish a GLSP that meets the delay requirement (We call this GLSP as CR-GLSP: Constraint-based Routed GLSP). The algorithm routes the GLSPs over the topology, and assigns wavelengths optimally to the various lightpaths. This assignment problem very likely belongs to the class of NP-hard problems [11], since its sub-problem, the static lightpath establishment has been shown to be NP-hard in [12]. Therefore, it is righteous to formulate the problem as an ILP.

We consider a connected network with N OXCs and L directed links such as fibers. Each fiber carries a certain number of wavelengths. A subset of these nodes are assumed to be edge O-LSRs between which lightpaths can be set up. We are given all the information on the virtual topology and status through a link state protocol. That is, Interior Gateway Protocol (IGP), such as

OSPF with extensions for optical and TE attributes will allow nodes to exchange information about optical network topology, resource availability, and even policy information.

Let $G = (\mathcal{N}, \mathcal{L})$ describe the given network with node set $\mathcal{N} = n_1, n_2, \ldots, n_{|\mathcal{N}|}$ and link set $\mathcal{L} = l_1, l_2, \ldots, l_{|\mathcal{L}|}$, where $|\mathcal{N}|$ and $|\mathcal{L}|$ denote the cardinalities of the node set and the link set, respectively. A link will be referred to by the ordered node pair it connects, (n_1, n_2). Let $\mathbf{D} = \{D_{ij}\}$ be the distance matrix from node i to node j that means a propagation delay from node i to node $j (i \neq j)$. Let $\mathcal{K} = k_1, k_2, \ldots, k_{|\mathcal{K}|}$ be the set of traffic demands belonging to a specific service class between a pair of edge O-LSRs. Typically there are several QoS service classes, each having characteristic specifications, such as delay. Here, we assume that the service class belongs to delay-sensitive class, since the algorithm focuses on delay QoS. Each traffic demand $k \in \mathcal{K}$ is defined by the ordered triple $(I_k, E_k, \varepsilon_k)$, where I_k is the ingress OXC, E_k is the egress OXC, and ε_k is the maximum edge-to-edge delay that is allowed for request k. For traffic flow, We define $\lambda^{I_k E_k}$ to be the average flow associated with the k^{th} traffic demand belonging to the delay-sensitive class requesting CR-GLSP set-up. These demands are assumed to arrive one at a time. Like $\lambda^{I_k E_k}$, λ_{ij}^{IE} denotes the traffic from ingress I to egress E that flows over an intermediate virtual link (i, j).

The unknown variables that need to be determined based on optimizing a certain objective function and satisfying a set of constraints are the following. These binary valued variables indicate a set of virtual connectivities:

$$v_{ij} = \begin{cases} 1 & \text{if the virtual topology has a direct fiber link } (i,j), \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $i, j = 1, 2, \ldots, |\mathcal{N}|$, $i \neq j$. If the TE function performs CR-GLSP setup online when a new CR-GLSP request arrives after a virtual topology is determined initially, these variables would assume the role of constants via virtual topology matrix. On the other hand, in case that the virtual topology is being formally modeled over the physical topology offline, the TE tool should compute the variables to overlay the virtual GLSPs. Thus, the online TE tool could be used to solve, in real time, problems such as the QoS based routing and rerouting of GLSPs.

When the network manager gets a request for a new CR-GLSP with a certain QoS requirement, it should determine whether this CR-GLSP can be routed to meet the QoS. The decision variables assume a binary value and denote which virtual link the GLSP select in the network. In mathematical form,

$$e_{ij} = \begin{cases} 1 & \text{if the CR-GLSP has a lightpath on link } (i, j) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where $i, j = 1, 2, \ldots, |\mathcal{N}|$ and $i \neq j$. That is, e_{ij} indicates whether the CR-GLSP is routed over the virtual link from node i to node j as in [13].

x_i indicates the layer 3 routing capability of the node as follows:

$$x_i = \begin{cases} 0 & \text{if node } i \text{ has no layer 3 processing} \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

The set of variables has to satisfy the following constraints.

$$\sum_j v_{ij} \leq T_i, \qquad \sum_i v_{ij} \leq R_j \qquad \text{for all } i, j. \quad (4)$$

where T_i and R_j are the number of transmitters and receivers, respectively, at node i and j $(i, j \in N)$.

$$\lambda_{ij} = (v_{ij} \sum_{I,E} \lambda_{ij}^{IE}) + e_{ij} \lambda^{I_k E_k} \qquad \text{for all } i, j. \quad (5)$$

$$\sum_j \lambda_{ij}^{I_k E_k} - \sum_j \lambda_{ji}^{I_k E_k} = \begin{cases} \lambda^{I_k E_k}, & i = I_k \\ -\lambda^{I_k E_k}, & i = E_k \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

$$\lambda_{ij} \leq W_{ij} C, \quad (7)$$

where C is the capacity of each wavelength on a fiber and W_{ij} denotes the number of wavelengths per link in the virtual topology between the nodes i and j for all i and j.

$$\lambda_{ij}^{I_k E_k} = e_{ij} \lambda^{I_k E_k}. \quad (8)$$

$$e_{ij} \leq v_{ij}. \quad (9)$$

The constraint 4 ensures that we can set up only single lightpath per port at each node. The constraints 5 and 6 are related to the traffic flow on virtual link for all i and j. When the new traffic demand is routed, the definition of the total traffic over link (i, j), λ_{ij} is expressed in the constraint 5. The constraint 6 assures that the flow of traffic is conserved at each node. Although splitting is used for load balancing purposes by routing demands over multiple GLSPs at the ingress point, it is not permissible for the routing algorithm to always split traffic in an arbitrary manner since the traffic being routed may be inherently unsplittable. In this paper, the traffic flow $\lambda^{I_k E_k}$ will not bifurcate either at any point in the network from the fact that the proposed TE algorithm is trying to route a CR-GLSP request explicitly to meet the QoS. Thus, the constraint 6 is brought in so the traffic flowing into an OXC to be tantamount to that flowing out of the OXC for any OXC other than the ingress and egress O-LSRs. In terms of the constraints, the link capacity should not be exceeded with the addition of the new CR-GLSP along its path through the network, that is presented in the constraint 7. The constraint 8 specifies that if the link (i, j) does not exist on the CR-GLSP, no traffic can be routed on that link. Finally, constraint 9 keeps the new CR-GLSP from being set up between two nodes if there is no virtual link connecting them. Additionally, we can incorporate the layer 3 port throughput into the constraint 7. When the traffic flowing through a link is going forward an intermediate O-LSR, the traffic demand should not be larger than the sum of the maximum throughput supported by IP router port:

$$\lambda_{ij} \leq W_{ij} C \{1 + (\alpha - 1) Q_j\}, \quad (10)$$

where $\alpha \leq 1$ denotes the maximum layer 3 port throughput, expressed as a fraction of the optical layer throughput.

A packet traversing the CR-GLSP experiences an end-to-end delay of $\sum_{i,j} e_{ij} \lambda_{ij}^{I_k E_k} D_{ij} + d_{node}$, where d_{node} denotes the total waiting time at all nodes over the CR-GLSP. The dominant delay in light to moderately loaded high bandwidth optical

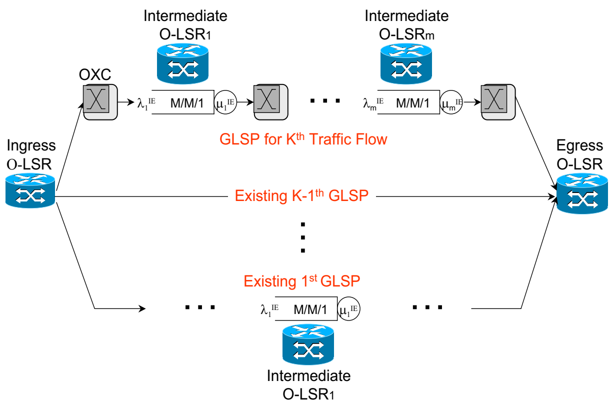

Fig. 2. Delay with Layer 3 Processing

networks is known as the propagation delay, which is uniquely determined by the length of the optical path over the physical WDM network. However, since wavelengths and fibers are finite, a lightpath is not always established between each source and destination IP routers. Moreover, if traffic grooming is available in some nodes, it is not always necessary to deploy a direct lightpath between two nodes to carry traffic between them. Instead, the traffic demand between two nodes may be carried by multiple lightpaths among some intermediate nodes without direct lightpath deployment between the two (end) nodes. Consequently, IP packets are forwarded to the destination by multi-hop IP routing. By this routing, layer 3 processing may still be a bottleneck of a network. Thus, d_{node} is defined to be the waiting and processing time at O-LSRs, since these nodes are the likely bottleneck points on any CR-GLSP. The delay increases with the number of hops and the expected processing delay at the intermediate nodes, since there can be an arbitrary number of intermediate O-LSRs on the GLSP.

we define λ_m^{IE} to be the aggregate input rate to the m^{th} ($m = 1, 2, \ldots, M$) O-LSR. Under the independence assumption on interarrivals we can model the delay experienced at layer 3 using an $M/M/1$ queueing model. Fig. 2 depicts the operation of such an optical Internet. Given that μ_m^{IE} be the service rate in each intermediate node, we can get the average layer 3 processing delay seen by the traffic flow as

$$\tau = \sum_{m=1}^{M} \frac{\rho_m^{IE}}{(1-\rho_m^{IE})\lambda_m^{IE}}, \tag{11}$$

where $\rho_m^{IE} = \lambda_m^{IE}/\mu_m^{IE}$ is the utilization. Using the local connectivity and processing rate variables, the above equation 11 becomes

$$\tau = \sum_{i,j} \frac{e_{ij} Q_j}{\mu_j - (\sum_{\ell} \lambda_{\ell j} + \lambda^{I_k E_k})}. \tag{12}$$

Thus, for the k^{th} traffic flow belonging to DS class, the objective function for delay can be written as

$$f(e_{ij}) = (\sum_{i,j} e_{ij} \lambda^{I_k E_k} D_{ij}) + \tau, \tag{13}$$

In Eq. 13, the second term τ will be zero if there is no layer 3 processing at any point on the GLSP. The objective function, Eq. 13, should be minimized in order to support the delay QoS requirement of the k^{th} flow belonging to the delay-sensitive class. If the minimum value of Eq.13 does not satisfy the delay QoS requirement, the values computed from the minimization would not be applied to the variable e_{ij}.

On top of the constraints in (4 - 9), an additional two constraints should be defined, that are related to QoS and layer 3 processing rate. One constraint for tolerable QoS can be defined as

$$0 \leq f(e_{ij}) \leq \varepsilon_k. \tag{14}$$

The other one for layer 3 processing rate can be expressed as

$$\mu_j \geq v_{ij} \sum_i \lambda_{ij}^{IE} + e_{ij} Q_j \lambda^{I_k E_k} \qquad \text{for all } I, E. \tag{15}$$

It is generally known that the multicommodity network flow problem with integer constraints is NP-hard [11]. Thus the k disjoint route problem which is NP-hard in [11] can be dealt with the same as that the k distinct egress node pairs find k mutually link-disjoint routes.

Whenever a new traffic flow belonging to delay-sensitive class requests explicitly routed GLSP, the virtual lightpath will be configured and the network state will be updated. Unless it is possible to support the GLSP with the desired QoS, the request can be blocked or renegotiated and attempted again. Alternatively, the network manager can preempt the minimum number of lower priority flows that will allow the GLSP to be set up. The procedure for the k^{th} request, which does not account for preemption, proceeds as follows:

Step 0: Obtain the parameters for the k^{th} setup request
Step 1: *if* a transmitter or a receiver is not available at I_k or E_k
    then go to Step 3;
Step 2: *if* there is any GLSP to minimize Eq. 13
    then go to Step 4;
    else if the request is negotiable
        then relax ε_k and go to Step 2;
        else go to Step 3;
Step 3: Block this GLSP request and go to Step 5;
Step 4: Provision resources for the GLSP
    and update network state;
Step 5: End;

To satisfy the requirements of diverse routing, rerouting and restoration as well as TE, explicit routing is necessary for constructing lightpaths. The route on which a new lightpath is to be established is specified by an object contained in the GLSP setup message. This route is typically be chosen by the ingress O-LSR, but it could be determined by a higher level network management system. The route may be specified either as a series of routers/OXCs, or in terms of the specific links used. Therefore, the above mechanism performs the calculation of primary and restoration lightpath routes on-line as the individual requests arrive. These lightpaths could be computed all at once by doing an offline calculation that accounts for all the pending requests.

Because the network loading varies over time, the consideration of the optimal route selection could require the reconfiguration of established lightpath routes, as described in [3]. Although frequent lightpath reroutings may not be acceptable, a limited number of lightpath reroutings could improve the network state, supporting the requested QoS of the future traffic while maintaining the QoS of the traffic that the network is already supporting. In the initial configuration stage where there is no configured virtual topology, the appropriate virtual lightpath could be found by repeating the above procedure. This procedure is applied to the traffic that has the most stringent delay limit among all initial traffic demands of delay-sensitive service class at all egress O-LSRs being done so to the traffic with the next highest QoS requirement, in turn. Like setting up usual GLSP, the virtual topology would be configured for the traffic demand of delay-sensitive service class at each egress OXC by minimizing the objective function of delay. By finding the values of all the elements of the set $\{\lambda_{ij}^{IE}\}$, we can obtain a full set of routing assignments for all the traffic in the optical network.

Failure of LSPs due to link failures is detected via GMPLS signaling protocol (e.g. CR-LDP or RSVP-TE) information by the edge routers. They can request a rerouting of the LSPs after the link-state database has been updated by routing protocols or by other means. (An alternative, not studied in the paper, is to setup a disjoint path backup GLSP so that failures can be accommodated by changing the FEC to GLSP mapping at the ingress routers.)

Let $\mathcal{S}$ be the set of Shared Risk Link Groups (SRLGs) associated with elements of the network. A single link or node may belong to more than one SRLG. Define S_{ij} to be the set of SRLG identifiers associated with the link (i, j) and define S_n to the the set of all SRG identifiers associated with node n. Then the full set of SRG identifiers for the network is just the union of all these sets: $\mathcal{S} = \left(\bigcup_{n\in\mathcal{N}} S_n\right) \bigcup \left(\bigcup_{\ell\in\mathcal{L}} S_\ell\right)$. In order to set up a M:N protection group with QoS constraints that generally routes M backup paths that are node disjoint from N primary paths, we can proceed in the following way. Let $\mathcal{K} = \{k_i\}_{i=1}^{K}$ be the set of requests for working paths in the protection group. For the first working path k_1, we can use the algorithm that we have described above since there is no SRLG/SRG constraint. The SRLG/SRG identifiers associated with the working path are accumulated in the set $S(k_1)$. In order to compute a backup path for this working path, which will have the same source and destination nodes I_{k_1} and E_{k_1} and the same delay and bandwidth QoS requirements ε_{k_1} and B_{k_1}, we can do several things. One possibility is to execute the Shortest Path First (SPF) algorithm described in [14], with some modifications. Like Eq. 11, let $\tau_n = \sum_{I,E} \frac{\rho_n^{IE}}{(1-\rho_n^{IE})\lambda_n^{IE}}$ be the average layer 3 processing delay at node n due to all the existing traffic flows that are passing through the node. Whenever a flow is added, modified, or deleted, $\{\tau_n\}_{n\in\mathcal{N}}$ should be updated. Recall that we defined D_{ij} to be the propagation delay on the link from Node i to Node j. Define B_{ij} to be the available bandwidth on the link from i to j.

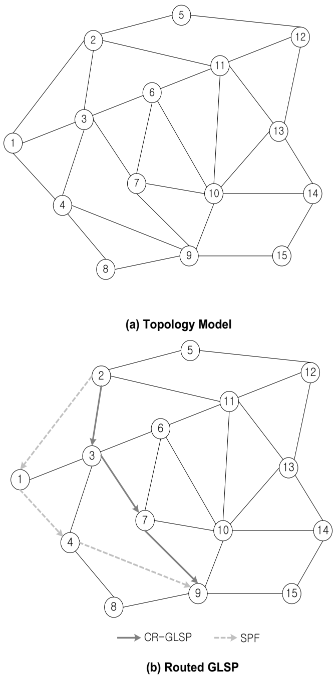

Fig. 3. Network Model

Step 1: If $B_{ij} < B_{k_1}$, set $D_{ij} = \infty$.
Step 2: If $S_{ij} \in S(k_1)$, set $D_{ij} = \infty$.
Step 3: (Backup path initialization.) Set $P = \{I_{k_1}\}, m = I_{k_1}$. Let $f(P) = \tau_{I_{k_1}}$ be the total delay on the backup path.
Step 4: Determine the node $n^\star$ with the lowest delay cost with respect to node m, which satisfies $\min_{n\in\mathcal{N}, n\notin P}[D_{m,n} + \tau_n]$.
Step 5: Add $n^\star$ to P. $f(P) = f(P) + D_{m,n^\star} + \tau_{n^\star}$.
Step 6: If $f(P) > \varepsilon_{k_1}$, no backup path satisfying the desired QoS requirements can be found. If $n^\star = E_{k_1}$, the algorithm has executed successfully.
Step 7: Set $m = n^\star$ and go to Step 4.

If it is not possible to find a backup path with the desired QoS characteristics, the algorithm could be run again with relaxed values for the maximum delay or minimum bandwidth, although this could result in a degradation of performance in the event of a failure on the primary path, even if the flow is successfully switched over to the backup path.

IV. Performance Evaluation from Simulation

Some simulation experiments have been conducted in order to study the performance of the proposed CR-GLSP routing algorithm. In this section, we present the simulation results of the proposed algorithm and compare it to the well-known SPF algorithm using a set of simulations. We have run our algorithm several times using GMPLS Lightwave Agile Switching Simulator (GLASS) which was developed by NIST to evaluate control algorithms in integrated MPLS/optical networks [15], and calculated average blocking probability and average end-to-end delay. That is, the performance is analyzed in terms of the number of rejected CR-GLSP requests and delay.

In the first place, we describe a simulation environment used to validate our algorithm in providing service requested by

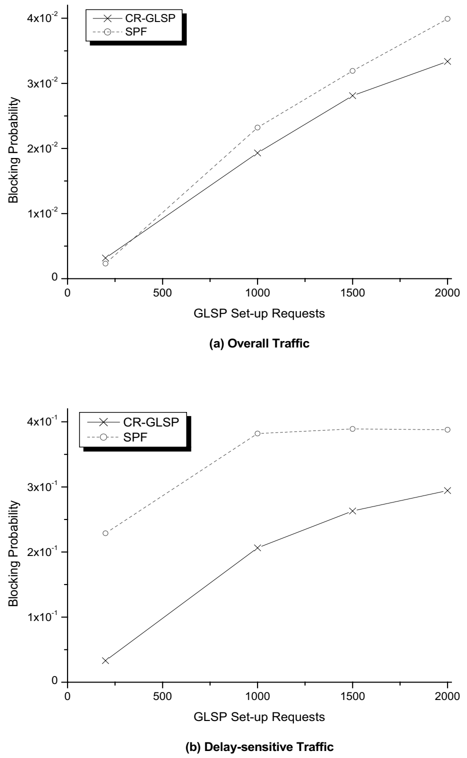

Fig. 4. Blocking Probability for Model 1

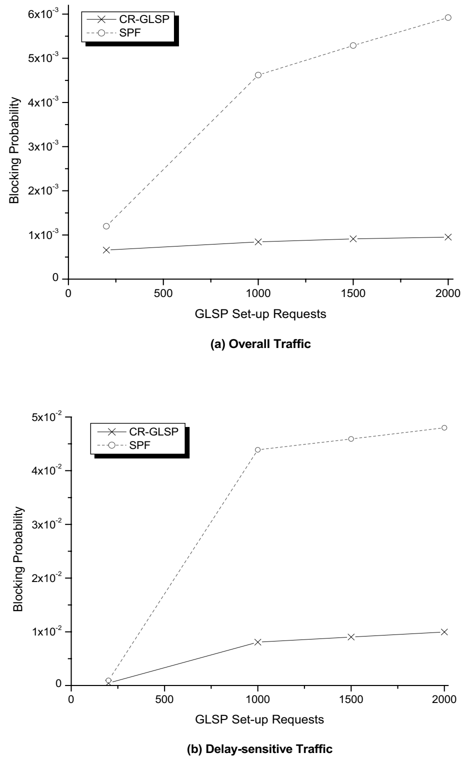

Fig. 5. Blocking Probability for Model 2

TABLE I

DELAY PERFORMANCE (UNIT: MSEC: $1/\lambda_r$; GLSP REQUESTS)

Model 1		
$1/\lambda_r$	CR-GLSP algorithm	SPF algorithm
200	0.266	0.424
1000	0.292	0.477
1500	0.321	0.499
2000	0.343	0.521
Model 2		
$1/\lambda_r$	CR-GLSP algorithm	SPF algorithm
200	0.215	0.371
1000	0.228	0.417
1500	0.257	0.421
2000	0.275	0.429

delay-sensitive class. The simulation tests were carried out on a model of the network shown in Fig. 3(a), which has also been used in [16]. In this topology model, a maximum of 4 wavelengths are available for use in each link and the capacity is 10 Gbps. The propagation delay D_{ij} is assumed to be 0.05 msec for all links (i, j), and the maximum tolerable delay limit of every GLSP request from delay-sensitive traffic is 0.5 msec. For this topology, we consider in this paper two network models: one model where the values of the x_j are assumed to be 1 for all the edge nodes and the other one where the network layer processing overhead is placed on the nodes 5,8,9,12 and 14. Every GLSP request arrives at all edge nodes, according to a Poisson process with mean arrival rate, λ_r of 200, 1000, 1500 and, 2000 in each experiment, respectively. Like in [17], the arrival rates of traffic for the different source and destination pair were chosen from a uniformly distributed random variable in the interval [1, λ^{IE}]. The value of λ^{IE} was selected as 2.5 Gbps in this simulation experiment. 10% of total traffic flows belongs to delay-sensitive service class. We assume that the time between GLSP set-up and teardown is exponentially distributed with a mean of 1 min. When a GLSP set-up request arrives at an ingress node, the destination is chosen randomly among all the edge nodes except the ingress with the set-up request. All the simulations were run till 5000 GLSP set-up requests were generated and each simulation was repeated 10 times to get the average blocking rate and the average delay measured during the simulation.

The average blocking performance versus the GLSP set-up requests is shown in Fig. 4 where graph (a) and (b) is the performance of overall traffic and that of delay sensitive traffic, respectively. Fig. 5 also shows the average blocking performance from the other set of simulations where the location of intermediate O-LSRs is changed, i.e. the layer 3 processing is at nodes 5, 8, 9, 12 and 14. The average blocking rate increases at the increasing of the GLSP set-up requests. Moreover, the gain due to the consideration of network layer processing, is evident from these graphs. Therefore, we find that the proposed CR-GLSP algorithm performed better with respect to the blocking rate, which means that the CR-GLSP algorithm utilizes network resources more efficiently. This improvement is realized due to the fact that although CR-GLSP algorithm would not be able to find the available bandwidth over the path with the minimum delay, it tries to utilize the path with the next minimum delay keeping the tolerable delay.

Without performance degradation in blocking rate, we obtain the better performance in terms of average end-to-end delay. The average delay associated with both algorithms is presented in Table.I measured during the overall simulation time for the two topology models where model 1 performs layer 3 processing at all the edge nodes and the model 2 does so at only the 5 nodes listed above. From Table. I, it can be observed that the average delay increases at the increasing of the GLSP set-up requests since the more GLSPs are requested to be established, the more the traffic flows into the network. Despite the increasing of the average delay, the average delay of the CR-GLSP algorithm is still smaller than that of SPF algorithm. This is due to the fact that the CR-GLSP algorithm incorporates network layer processing delay into routing decision. It resulted from this incorporation that the CR-GLSP algorithm chose different path from that in SPF algorithm between source node 2 and destination node 9. As can be seen in Fig. 3(b), the SPF algorithm routed the GLSP over the route $2 \to 1 \to 4 \to 9$ while the CR-GLSP algorithm selected the route $2 \to 3 \to 7 \to 9$.

Therefore, it is known that our formulation in CR-GLSP is applicable to explicit routing of backup GLSP as well as working GLSP in large IP over WDM networks with QoS constraints.

V. Conclusion

The success of next generation IP over WDM networking depends on the ability to offer and support QoS to customers. It is clear that network layer processing cannot be neglected for designing TE algorithms as well as for this. In this paper, we formulated a optimization problem for constraint-based routing to support the QoS requested by delay-sensitive traffic in optical Internet. The CR-GLSP algorithm described in this paper is based on the idea that traffic flows encounter the delay resulting from layer 3 processing at subnetwork edge nodes. The algorithm uses the current state of the network to determine the delay associated with each possible path, and then chooses the path with the minimum total delay. The simulation results showed that the proposed algorithm improves the delay performance as well as the probability of successful GLSP establishment in comparison to SPF algorithm which do not take the network layer processing into account. In the simulation test, GLASS tool was used as a simulation tool that was developed by NIST for optical Internet. The presented algorithm can also be used for establishing backup path with delay constraint and we described how to apply to protection.

The future study is network survivability which is also treated as an important issue recently. We are designing a several efficient restoration algorithms that ensure its survivability in optical Internet while we sustain the delay QoS. At that time, it is worth noting multiple service categories whose level of restoration is different.

References

[1] E. Rosen et al., "Multiprotocol Label Switching Architecture", *R*FC 3031, Jan. 2001.

[2] A. Banerjee et al., "Generalized Multiprotocol Label Switching: An Overview of Routing and Management Enhancements", *IEEE Commun. Mag.*, Jan. 2001,

[3] G. Apostolopoulos et al., "QoS Routing Mechanisms and OSPF Extensions", RFC 2676, Aug. 1999.

[4] E. Crawley et al., "A Framework for QoS-based Routing in the Internet", RFC 2386, Aug. 1998.

[5] N. Golmie, T. D. Ndousse, and D. Su, "A Differentiated Optical Services Model for WDM Networks", *I*EEE Commun. Mag., vol.38, no.2, Feb. 2000.

[6] H. Harai, F. Kubota, and H. Nakazato, "Design of Reconfigurable lightpaths in IP over WDM Networks", *I*EICE Transactions on Communications, vol.E83-B, no.10, Oct. 2000.

[7] D. Awduche et al., "Requirements for Traffic Engineering over MPLS", *R*FC 2702, Sep. 1999.

[8] Data Connection Computer Technology, "MPLS Traffic Engineering: A Choice of Signaling Protocols", *http://www.dataconnection.com/download/crldprsvp.pdf*

[9] P.Ashwood et al., "Generalized MPLS Signaling - RSVP-TE Extensions," *W*ork on Progress, draft-ietf-mpls-generalized-rsvp-te-04.txt, Jul. 2001.

[10] P. Ashwood, et al., "Generalized MPLS Signaling - CR-LDP Extensions", *W*ork on Progress, draft-ietf-mpls-generalized-cr-ldp-04.txt, Jul. 2001.

[11] M. Garey, and D. Johnson, *C*omputers and Intractability, W.H. Freeman and Company, 1979.

[12] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath Communications: An Approach to High Bandwidth Optical WANs", *I*EEE Trans. on Commun., vol.40, no.7, Jul. 1992.

[13] R. Ramaswami and K. N. Sivarajan, *O*ptical Networks: A practical Perspective, Morgan Kaufmann, 1998.

[14] Z. Wang and J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications", *I*EEE JSAC, vol.14, no.7, Sep. 1996.

[15] NIST, *h*ttp://www.antd.nist.gov/proj/asia/glass/

[16] K. Kar, M. Kodialam, and T.V. Lakshman, "Minimum Interference Routing of Bandwidth Guaranteed Tunnels with MPLS Traffic Engineering Applications", *I*EEE JSAC, vol.18, no.12, Dec. 2000.

[17] I. Widjaja and A.I. Elwalid, "Study of GMPLS Lightpath Setup over Lambda-Router Networks", *I*CC'02, May. 2002.