Performance Analysis of a Generic GMPLS Switching Architecture under ON–OFF Traffic Sources*

Ling-Chih Kao and Zsehong Tsai

Department of Electrical Engineering, National Taiwan University, Taipei, TAIWAN
d6942004@yahoo.com.tw, ztsai@cc.ee.ntu.edu.tw

Abstract. This paper proposes a queueing model including the control plane and the switching buffer mechanism of a GMPLS switch for evaluating the performance of a GMPLS switching architecture. With the proposed model, one can select appropriate parameters for the label-setup policy and the label-release policy to match the traffic load and network environment. Key performance metrics, including the throughput, the label-setup rate, and the fast path bandwidth utilization can be obtained via the analytical results. Numerical results and simulations are used to verify the accuracy of our proposed queueing model. The trade-off among these performance metrics can be observed as well.

1 Introduction

In recent years, it has been a trend to provide a wide range of data services over the same backbone network via newly adopted technologies such as Multi-Protocol Label Switching (MPLS) [1] and Multi-Protocol Lambda Switching (MPλS) [2] to overcome the scalability and complexity issues. Many other layer-3 switching technologies [3] have also been specially designed to solve the dilemma of routing table scalability and overload of routing processing. But these proposed technologies may not be compatible with one another. In order to integrate these techniques, the Internet Engineering Task Force (IETF) proposes MPLS. The basic concept of MPLS is that packet forwarding is based on a fixed short length label instead of longest matching search, which can shorten packet transit time. There are two most popular approaches for connection setup in MPLS. The traffic-driven method is to trigger label-setup according to traffic demand, while the topology-driven system is based on routing information. In addition, by way of Constraint-based Routed Label Distribution Protocol (CR-LDP) [4] or Resource ReSerVation Protocol (RSVP) [5], it is possible to include QoS mechanism in MPLS. When it is necessary to combine traffic engineering aspect of MPLS and bandwidth provision capability of DWDM, MPλS [2] is found to play a major role. Meanwhile, considering that there are many different underlying data-link and physical layer technologies, Generalized Multi-Protocol Label Switching (GMPLS) [6,7] is thus suggested to extend MPLS to encompass time-division, wavelength (lambdas) and spatial switching.

In order to control different switching operations under GMPLS, the label defined for various switches is required to be of different formats [8], and related signaling and

* This work was supported by National Science Council, R.O.C., under Grant NSC 90-2213-E-002-076, and by Ministry of Education, R.O.C., under Grant 89E-FA06-2-4-7.

I. Chong (Ed.): ICOIN 2002, LNCS 2343, pp. 407–418, 2002.

routing protocols also need modifying [6,9]. However, the key operations of the control plane of these various switching protocol suites are found to be similar. Although basic functions of the GMPLS control plane have been discussed or defined in the literature, operation procedures for efficient resource control have still being defined and their impact on performance have still being investigated. Some papers [10]–[12] proposed performance queueing models for MPLS or GMPLS, but most detailed operations of the GMPLS control plane are not well covered. At the same time, it is often found that a sophisticated queueing model which can evaluate the performance of the GMPLS switching network is not easy to build. Therefore, a model embracing detailed operations of the GMPLS control plane is thus strongly needed.

In this paper, we develop a queueing model to characterize the behavior of most operations of a GMPLS switch. The aggregation of IP streams can save usage of labels (or lambdas) and thus alleviate the processing load of the GMPLS controller. The label-setup policy is based on the accumulated packets in the default path buffer. The label-release policy is controlled by an adjustable label-release timer. Efficient resource allocation mechanism is thus achieved by fine tuning the flexible label-setup policy and the adjustable label-release timer. Although our queueing model is traffic-driven oriented, the behavior of the topology-driven system can be approximately obtained via extreme case of this traffic-driven model. Some key performance measures, such as the throughput, the label-setup rate, and the path bandwidth utilization, can all be derived in the proposed model.

The remainder of the paper is organized in the following. In Section II, the queueing model for a GMPLS switch is described. In Section III, the analysis procedure is proposed. In Section IV, three performance measures are derived. Numerical experiments and simulation results are discussed in Section V. Conclusions are drawn in Section VI.

2 Queueing Model

In this section, a queueing model characterizing the behavior of an aggregated IP stream passing through a GMPLS switch is proposed. The number of labels is assumed to be enough for all incoming flows. The bandwidth allocated to each label (or a flow) is fixed. Therefore, we can focus on investigating the steady-state performance of a GMPLS switch without label contentions. For simplicity, we focus on the case that only one single flow is included in this queueing model. The results can then be easily extended to the general case. Regarding the traffic source, an aggregated stream (equivalently a flow) is assumed to be consisted of N homogeneous IPPs (Interrupted Poisson Process), where each IPP has an exponentially distributed *on* (*off*) duration with mean equals $1/\alpha$ $(1/\beta)$ and λ is the arrival rate in *on* state. Note that this traffic source model includes four parameters to match most Markovian traffic patterns[1]

The queueing model for a GMPLS switch, GMPLS queueing model, is shown in Fig. 1. The solid lines in Fig. 1 denote the paths that packets go through and the dotted lines are the signaling paths. There are three major functional blocks in this model: the GMPLS controller, the default route module, and the fast route module. The functions

[1] Self-similar process is not considered in this paper.

of the control plane are included in the GMPLS controller. The default route module stands for the IP-layer (layer-3) and data-link layer (layer-2) on the default path. The fast route data-link is represented by the fast route module. In this GMPLS architecture, the *label* is used as a generic term. When GMPLS is used to control TDM such as SONET, time slots are labels. Each frequency (or λ) corresponds to a label when FDM such as WDM is taken as the underlying switching technology. When the switching mechanism is space-division multiplexing based, labels are referred to as ports. Six queueing nodes are included in this model: *Default_Route, Fast_Route, Label_Pool, Fast_Route_Setup, Label_Release, and Label_Release_Timer.* Traffic served by traditional routing protocol will be served by the *Default_Route* node, whose buffer stores the packets which cannot be processed in time by the IP processor on the default path. Meanwhile, the *Fast_Route* node serves packets whose stream has been assigned a label. The fast path buffer stores the packets which can not be processed in time by the fast route. The *Label_Pool* stores the labels which represent the availability of the fast path and the fast path is available if there is a label in the *Label_Pool.* The *Fast_Route_Setup* node represents the time required to set up an LSP (Label Switched Path) for an aggregated stream. The *Label_Release* node represents the time required to release a label. The *Label_Release_Timer* node represents a label-release timer. This timer indicates the maximum length of idle period of an aggregated stream before its label is released for other use. Once an aggregated stream is granted a label, it is served with its own *Fast_Route* node and uses its own label-release mechanism. As a result, this model is used to examine the protocol efficiency instead of label competitions. The assumed details of label operations over this queueing model are described as follows.

When an aggregated IP stream arrives, two possible operations may occur. In the first case, incoming traffic has been assigned a fast path. Then its packets will be directly sent to the *Fast_Route* node. In the second case, incoming traffic has not been assigned a fast path. All the packets are continuously served by the *Default_Route* node (via the *default path*) during the label-setup operations under this situation. If the accumulated packets in the buffer of the *Default_Route* node have not reached the triggering threshold (m), it is served by the *Default_Route* node through traditional IP routing protocol. However, if the accumulated packets in the buffer of the *Default_Route* node reach the triggering threshold, the flow classifier & label_setup_policy manager will trigger the default route module to send a setup request packet through the *Fast_Route_Setup* node to its downstream LSR (Label Switch Router) until the egress LSR for negotiating an appropriate LSP according to the current network resources. The GMPLS controller will set up a path called the *fast path* for this stream and assign it a label.

The label manager maintains an activity timer to control the label-release operation of the flow. The label is released only if the activity timer indicating that the maximum allowed inactive duration has been reached. Incoming packets will be blocked if the accumulated packets in the *Default_Route* node exceed the buffer size of the *Default_Route* node but the stream has not been assigned a fast path, or if the accumulated packets in the *Fast_Route* node exceed the buffer size of the *Fast_Route* node when the stream has been assigned a fast path.

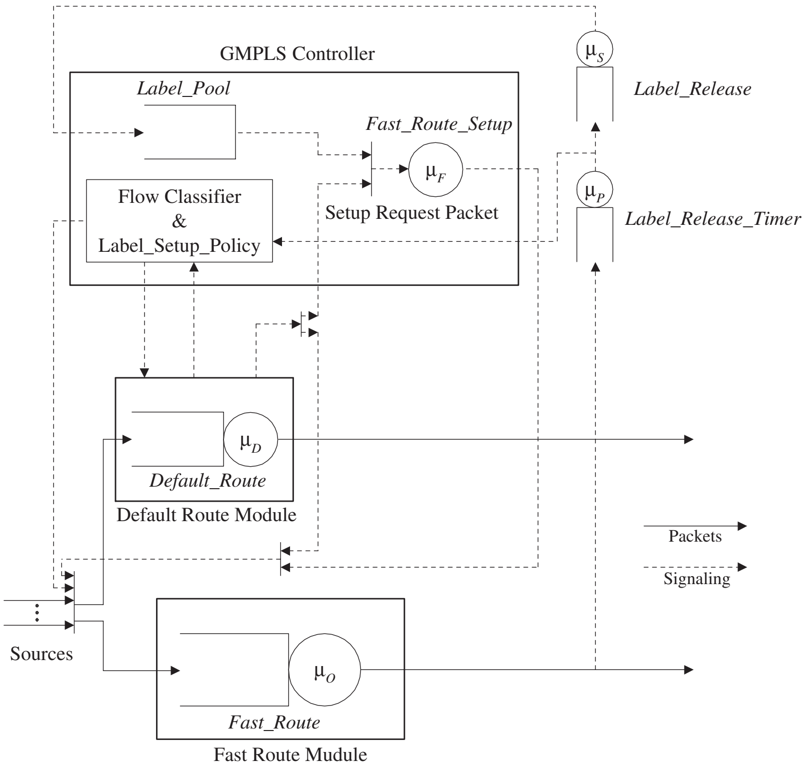

Fig. 1. GMPLS queueing model.

3 Steady-State Analysis

We here propose a procedure to calculate steady-state distribution of a GMPLS switch model as shown in Fig. 1. We adopt the following notations:

$1/\mu$: packet length (bit per packet).
C_D: default path capacity (bps).
C_O: fast path capacity (bps).
$\mu_O = \mu C_O$: service rate of the *Fast_Route* node.
$T_{rel} = \frac{1}{\mu_P}$: the average sojourn time of the *Label_Release_Timer* node, where μ_P is the service rate of the *Label_Release_Timer* node.
μ_S: service rate of the *Label_Release* node.
μ_F: service rate of the *Fast_Route_Setup* node.
T_{LSP}: label-setup latency.
$\mu_D = \mu C_D$: service rate of the *Default_Route* node.
n: buffer size of *Default_Route* node.
t: buffer size of *Fast_Route* node.
n_I: the number of packets in the *Default_Route* node.

n_S: the number of packets in the *Fast_Route* node.
n_T: the number of labels in the *Label_Pool* ($n_T = 1$, if the fast path is available; $n_T = 0$, otherwise).
n_O: the number of IPPs in *on* state.
π_T: the state of the *Label_Release_Timer* node ($\pi_T = 0$, if the *Label_Release_Timer* node is idle; $\pi_T = 1$, otherwise).
π_R: the state of the *Label_Release* node ($\pi_R = 0$, if the *Label_Release* node is idle; $\pi_R = 1$, otherwise).
m: the triggering threshold which represents the minimum number of accumulated packets that will trigger label-setup operation.

In order to avoid the state size explosion problem, we employ the technique of state-aggregation approximation. The aggregated system state of the GMPLS queueing model is defined as the number of IPPs in *on* state, and we use π_k to denote the steady-state probability, where k (ranging from $0 \sim N$) is the aggregated state. The aggregated transition diagram is shown in Fig. 2. We then employ the state vector (a, b, c, d, e, f) to represent the internal state of the aggregated state k of the GMPLS queueing model when $n_I = a, n_S = b, n_O = c, \pi_T = d$, $\pi_R = e$, and $n_T = f$. The behavior of the GMPLS queueing model can then be predicted by a Markov chain.

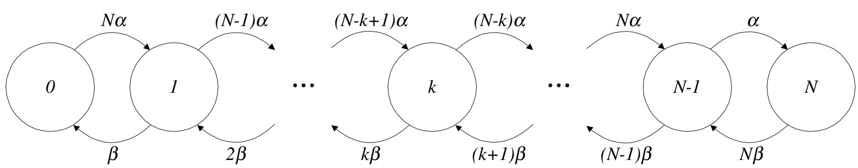

Fig. 2. Aggregated state-transition diagram of the GMPLS queueing model.

In this model, the service time is assumed to be exponentially distributed in all nodes. At the same time, silence interval, burst length and IP packet size are also assumed to be exponentially distributed. According to the above definitions, the global balance equations in state k of aggregated state-transition diagram are listed as follows.

$$P_{n,t,k,0,0,0} = (1 - \mu_O - \mu_D)P_{n,t,k,0,0,0} + k\lambda P_{n,t-1,k,0,0,0} \quad (1)$$

$$\begin{aligned} P_{n-i,t,k,0,0,0} &= (1 - \mu_O - \mu_D)P_{n-i,t,k,0,0,0} + \mu_D P_{n-i+1,t,k,0,0,0} \\ &+ k\lambda P_{n-i,t-1,k,0,0,0} \ , 1 \leq i \leq n-1 \end{aligned} \quad (2)$$

$$P_{0,t,k,0,0,0} = (1 - \mu_O)P_{0,t,k,0,0,0} + \mu_D P_{1,t,k,0,0,0} + k\lambda P_{0,t-1,k,0,0,0} \quad (3)$$

$$\begin{aligned} P_{n-i,t-j,k,0,0,0} &= (1 - \mu_O - \mu_D - k\lambda)P_{n-i,t-j,k,0,0,0} + \mu_D P_{n-i+1,t-j,k,0,0,0} \\ &+ k\lambda P_{n-i,t-j-1,k,0,0,0} + \mu_O P_{n-i,t-j+1,k,0,0,0} \ , \\ & 1 \leq i \leq n-1, \ 1 \leq j \leq t-2 \end{aligned} \quad (4)$$

$$\begin{aligned} P_{n,t-j,k,0,0,0} &= (1 - \mu_O - \mu_D - k\lambda)P_{n,t-j,k,0,0,0} + \mu_O P_{n,t-j+1,k,0,0,0} \\ &+ k\lambda P_{n,t-j-1,k,0,0,0} \ , 1 \leq j \leq t-2 \end{aligned} \quad (5)$$

$$\begin{aligned} P_{0,t-j,k,0,0,0} &= (1 - \mu_O - k\lambda)P_{0,t-j,k,0,0,0} + \mu_O P_{0,t-j+1,k,0,0,0} \\ &+ k\lambda P_{0,t-j-1,k,0,0,0} + \mu_D P_{1,t-j,k,0,0,0} \ , 1 \leq j \leq t-2 \end{aligned} \quad (6)$$

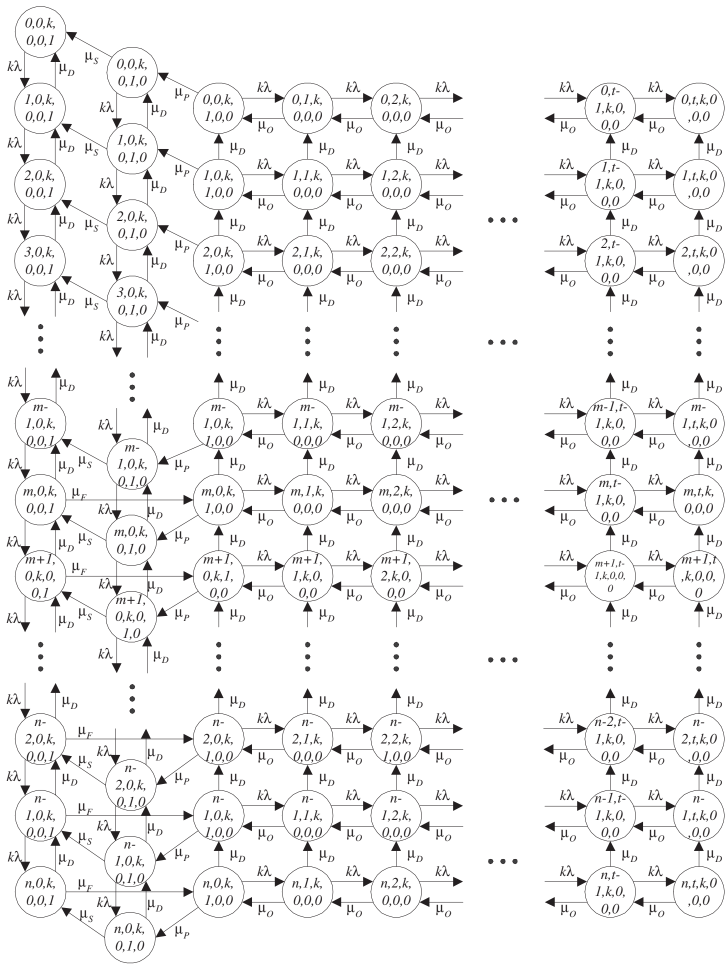

Fig. 3. Detailed state-transition diagram in aggregated state k.

$$P_{n,1,k,0,0,0} = (1 - \mu_O - \mu_D - k\lambda)P_{n,1,k,0,0,0} + \mu_O P_{n,2,k,0,0,0} + k\lambda P_{n,0,k,1,0,0} \tag{7}$$

$$P_{n-i,1,k,0,0,0} = (1 - \mu_O - \mu_D - k\lambda)P_{n-i,1,k,0,0,0} + \mu_O P_{n-i,2,k,0,0,0} + k\lambda P_{n-i,0,k,1,0,0} + \mu_D P_{n-i+1,1,k,0,0,0}\ , 1 \le i \le n-1 \tag{8}$$

$$P_{0,1,k,0,0,0} = (1 - \mu_O - k\lambda)P_{0,1,k,0,0,0} + \mu_O P_{0,2,k,0,0,0} + k\lambda P_{0,0,k,1,0,0} + \mu_D P_{1,1,k,0,0,0} \tag{9}$$

$$P_{n,0,k,1,0,0} = (1 - \mu_D - \mu_P - k\lambda)P_{n,0,k,1,0,0} + \mu_O P_{n,1,k,0,0,0} + \mu_F P_{n,0,k,0,0,1} \tag{10}$$

$$P_{n-i,0,k,1,0,0} = (1 - \mu_D - \mu_P - k\lambda)P_{n-i,0,k,1,0,0} + \mu_O P_{n-i,1,k,0,0,0} + \mu_D P_{n-i+1,0,k,1,0,0} + \mu_F P_{n-i,0,k,0,0,1}\ , 1 \le i \le n-m \tag{11}$$

$$P_{n-i,0,k,1,0,0} = (1-\mu_D-\mu_P-k\lambda)P_{n-i,0,k,1,0,0} + \mu_O P_{n-i,1,k,0,0,0} + \mu_D P_{n-i+1,0,k,1,0,0}\ ,n-m+1 \le i \le n-1 \tag{12}$$

$$P_{0,0,k,1,0,0} = (1-\mu_P-k\lambda)P_{0,0,k,1,0,0} + \mu_O P_{0,1,k,0,0,0} + \mu_D P_{1,0,k,1,0,0} \tag{13}$$

$$P_{n,0,k,0,1,0} = (1-\mu_S-\mu_D)P_{n,0,k,0,1,0} + \mu_P P_{n,0,k,1,0,0} + k\lambda P_{n-1,0,k,0,1,0} \tag{14}$$

$$P_{n-i,0,k,0,1,0} = (1-\mu_S-\mu_D-k\lambda)P_{n-i,0,k,0,1,0} + \mu_P P_{n-i,0,k,1,0,0} + k\lambda P_{n-i-1,0,k,0,1,0} + \mu_D P_{n-i+1,0,k,0,1,0}\ ,1 \le i \le n-1 \tag{15}$$

$$P_{0,0,k,0,1,0} = (1-\mu_S-k\lambda)P_{0,0,k,0,1,0} + \mu_P P_{0,0,k,1,0,0} + \mu_D P_{1,0,k,0,1,0} \tag{16}$$

$$P_{n,0,k,0,0,1} = (1-\mu_F-\mu_D)P_{n,0,k,0,0,1} + \mu_S P_{n,0,k,0,1,0} + k\lambda P_{n-1,0,k,0,0,1} \tag{17}$$

$$P_{n-i,0,k,0,0,1} = (1-\mu_F-k\lambda-\mu_D)P_{n-i,0,k,0,0,1} + \mu_S P_{n-i,0,k,0,1,0} + k\lambda P_{n-i-1,0,k,0,0,1} + \mu_D P_{n-i+1,0,k,0,0,1}\ ,1 \le i \le n-m \tag{18}$$

$$P_{n-i,0,k,0,0,1} = (1-\mu_D-k\lambda)P_{n-i,0,k,0,0,1} + \mu_S P_{n-i,0,k,0,1,0} + k\lambda P_{n-i-1,0,k,0,0,1} + \mu_D P_{n-i+1,0,k,0,0,1}\ , n-m+1 \le i \le n-1 \tag{19}$$

$$P_{0,0,k,0,0,1} = (1-k\lambda)P_{0,0,k,0,0,1} + \mu_S P_{0,0,k,0,1,0} + \mu_D P_{1,0,k,0,0,1} \tag{20}$$

where $P_{a,b,c,d,e,f}$ is the steady-state probability of the state vector (a,b,c,d,e,f). The detailed state-transition diagram corresponding to equations (1)–(20) is shown in Fig. 3.

4 Performance Measures

One key performance metric is the throughput. We define T_d and T_f as the average throughput at the *Default_Route* node and *Fast_Route* node respectively. $T_{total} = T_d + T_f$ is the total throughput. Their formulas are given by

$$T_d = \mu_D \{ \sum_{k=0}^{N} \sum_{i=1}^{n} \sum_{j=1}^{t} P_{i,j,k,0,0,0}\pi_k + \sum_{k=0}^{N} \sum_{i=1}^{n} (P_{i,0,k,1,0,0} + P_{i,0,k,0,1,0} + P_{i,0,k,0,0,1})\pi_k \} \tag{21}$$

$$T_f = \mu_O \sum_{k=0}^{N} \sum_{i=0}^{n} \sum_{j=1}^{t} P_{i,j,k,0,0,0}\pi_k \tag{22}$$

Since the label-setup rate is proportional to the required label processing load, it is included as another key metric. The label-setup rate S_R is defined as the average number of label-setup operations in the *Fast_Route_Setup* node per unit time and given by

$$S_R = \mu_F \sum_{k=0}^{N} \sum_{i=m}^{n} P_{i,0,k,0,0,1}\pi_k \tag{23}$$

Regarding the path bandwidth utilization we focus on the prediction of the ratio of wasted bandwidth on the fast path. For the fast path, the time periods considered to be "reserved" by an aggregated stream include the packet transmission time by the *Fast_Route* node (with time ratio B_f), the idle period waiting for label-release timeout (with time ratio B_t) and the time required to release a label (with time ratio B_r). However, only the period that the packets are transmitted by the *Fast_Route* node is considered effectively utilized. Hence, the fast path bandwidth utilization U_F is given by $U_F = \frac{B_f}{B_f+B_t+B_r}$, where $B_f = \sum_{k=0}^{N} \sum_{i=0}^{n} \sum_{j=1}^{t} P_{i,j,k,0,0,0}\pi_k$, $B_t = \sum_{k=0}^{N} \sum_{i=0}^{n} P_{i,0,k,1,0,0}\pi_k$, and $B_r = \sum_{k=0}^{N} \sum_{i=0}^{n} P_{i,0,k,0,1,0}\pi_k$.

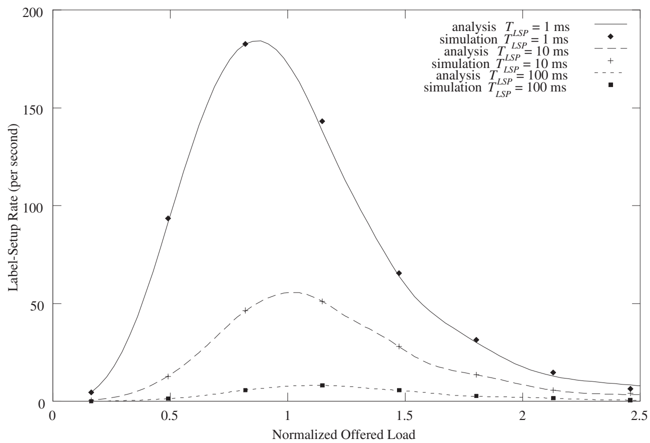

Fig. 4. Label-setup rate as a function of normalized offered load with $T_{rel} = 1$ ms and $m = 3$ under different T_{LSP}.

5 Numerical Examples

In this section, we demonstrate the applicability of the queueing model and discuss analytical and simulation results of the proposed generic GMPLS switch. We also illustrate the trade-off among key system parameters. Throughout this section, we set the number of IPPs (N) to 5, the average silence interval ($1/\alpha$) to 0.2 sec, the average burst length ($1/\beta$) to 0.8 sec, the average IP packet size to 512 bytes, the fast path capacity to 150 Mbps, the default path capacity to 100 Mbps, the average label-release latency ($\frac{1}{\mu_S}$) to 0.2 ms, the buffer size of *Default_Route* node (n) to 50 packets, and the buffer size of *Fast_Route* node (t) to 30 packets. The normalized offered load is defined as $N\lambda\left(\frac{\alpha}{\alpha+\beta}\right)/\mu_D$.

From Fig.s 4 and 5, one can observe that the longer the label-setup latency (T_{LSP}), the lower the label-setup rate. In other words, when it takes time to set up an LSP due to long T_{LSP}, more traffic will go through the default path. Hence, a switch with very large T_{LSP} should not be considered a topology-driven system because almost all traffic still goes through its default path. When traffic load becomes large, we also notice the increase of LSP lifetime. As a result, the label-setup rate decreases as traffic increases.

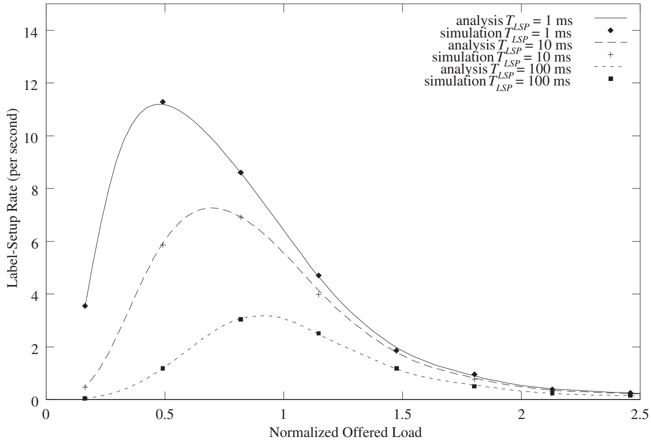

Fig. 5. Label-setup rate as a function of normalized offered load with $T_{rel} = 50$ ms and $m = 3$ under different T_{LSP}.

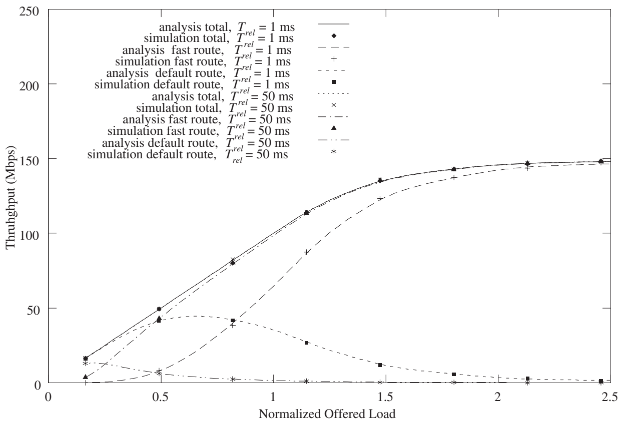

Fig. 6. Throughput as a function of normalized offered load with $T_{LSP} = 1$ ms and $m = 3$ under different T_{rel}.

Although the total throughput is almost the same under different T_{LSP} and label release timer (T_{rel}), the difference exists in the behavior of default path and fast path. With our model, one can determine how much traffic is served by the fast path. We plot the throughput as a function of normalized offered load with $m = 3$ under different T_{LSP} and T_{rel} in Fig. 6 and Fig. 7. From these two figures, one can find that the default path throughput will increase with the increase of traffic load if total traffic load is light. When most traffic starts to be switched to the fast path, the default path throughput decreases. Additionally, one can observe that most traffic will go through the fast path with larger T_{rel}, even under different T_{LSP}. Another phenomenon is that most traffic

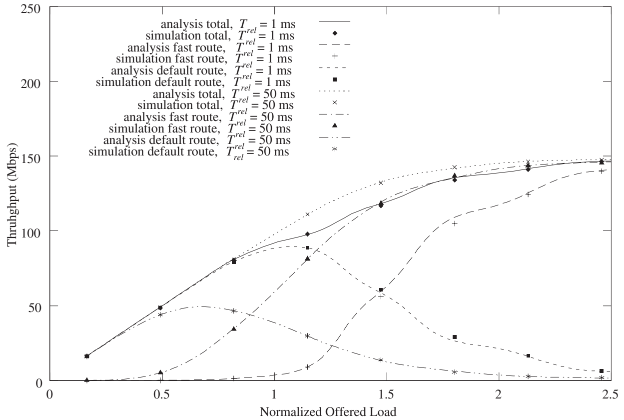

Fig. 7. Throughput as a function of normalized offered load with $T_{LSP} = 100$ ms and $m = 3$ under different T_{rel}.

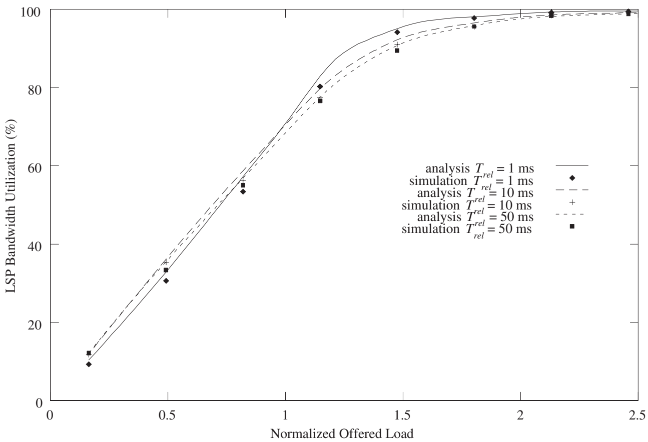

Fig. 8. Fast path bandwidth utilization as a function of normalized offered load with $T_{LSP} = 10$ ms and $m = 3$ under different T_{rel}.

goes through the default path with small T_{rel} and large T_{LSP} in the range of small to medium traffic condition.

The ratio of wasted bandwidth can be predicted by the fast path bandwidth utilization. From Fig. 8, one can know that the fast path bandwidth utilization for smaller T_{rel} is always higher than that for larger T_{rel} under arbitrary traffic load as long as T_{rel} is set sufficiently long. However, one can observe that the fast path bandwidth utilization with small T_{rel} (such as 1 ms) is lower than that with larger T_{rel} (such as 10 ms and 50 ms) when traffic load is light or medium. The reason is that the time ratio waiting for label-release timeout (B_t) under smaller value of T_{rel} will increase under such load. When traffic is heavy, this phenomenon will diminish because B_t becomes small.

From the above results, one can know that when T_{rel} is small, the system behavior is traffic-driven oriented. However, in the case that T_{rel} is extremely large, the system behavior approaches a topology-driven GMPLS switch.

6 Conclusions

The queueing model for a generic GMPLS switching architecture is proposed. On the basis of the approximated analysis and simulation results, one can effectively fine tune the resource utilization level or label processing load. Furthermore, the trade-off between the fast path bandwidth utilization and the label-setup rate can be observed. Hence, an appropriate value of label release timer T_{rel} can be carefully selected to meet the requirement of both. For a network with large round-trip time and sufficient resources in the fast path, if one uses a small value of T_{rel}, most traffic will go through the default path instead of the fast path. Therefore, choosing large value of T_{rel} is preferred. For a network with a small round-trip delay and insufficient resources in the fast path, it is adequate to use the system with a small value of T_{rel}.

Our study shows that the best performance of a GMPLS switch can be achieved only when its control plane parameters are appropriately tuned. In the future, we will investigate a mechanism to reduce the out-of-sequence problem due to dynamic path changes in GMPLS.

References

1. E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," *RFC 3031,* Jan. 2001.
2. D. Awduche, Y. Rekhter, J. Darke, and R. Colton, "Multi-Protocol Lambda Switching: Combining MPLS Traffic Engineering Control with Optical Crossconnects," *IETF Internet draft-awduche-mpls-te-optical-02.txt,* July 2000.
3. C. Y. Metz, *IP Switching: Protocols and Architextures,* MacGraw-Hill, 1999.
4. O. Aboul-Magd, et al., "Constraint-Based LSP Setup using LDP," *IETF Internet draft-ietf-mpls-cr-ldp-05.txt,* Feb. 2001.
5. D. O. Awduche, et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels," *IETF Internet draft-ietf-mpls-rsvp-lsp-tunnel-09.txt,* Aug. 2001.
6. P. Ashwood-Smith, et al., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture," *IETF Internet draft-ietf-ccamp-gmpls-architecture-00.txt,* June 2001.
7. A. Banerjee, et al., "Generalized Multiprotocol Label Switching: An Overview of Routing and Management Enhancements," *IEEE Commun., Mag.,* pp. 2–8, Jan. 2001.
8. J. Sadler, et al., "Generalized Switch Management Protocol (gsmp)," *IETF Internet draft-sadler-gsmp-tdm-labels-00.txt,* Feb. 2001.
9. P. Ashwood-Smith, et al., "Generalized MPLS–Signaling Functional Description," *IETF Internet draft-ietf-mpls-generalized-signaling-05.txt,* July 2001.
10. S. Nakazawa, K. Kawahara, S. Yamaguchi, and Y. OIE, "Performance Comparasion with Layer 3 Switches in Case of Flow-And Topology-Driven Connection Setup," *IEEE GLOBECOM'99,* pp. 79–86, Rio de Janeiro, Brazil.

11. L.-C. Kao and Z. Tsai, "Performance Analysis of Flow-Based Label Switching: the Single IP Flow Model," *IEICE Trans. Commun.,* vol. E83-B, no. 7, pp. 1417–1425, July 2000.
12. L.-C. Kao and Z. Tsai, "Steady-State Performance Analysis of MPLS Label Switching," *IEICE Trans. Commun.* vol. E84-B, no.8, pp. 2279–2291, Aug. 2001.