# New Preemption Policies for DiffServ-Aware Traffic Engineering to Minimize Rerouting in MPLS Networks

Jaudelice C. de Oliveira, *Member, IEEE*, Caterina Scoglio, *Associate Member, IEEE*, Ian F. Akyildiz, *Fellow, IEEE*, and George Uhl

***Abstract*—The preemption policy currently in use in MPLS-enabled commercial routers selects LSPs for preemption based only on their priority and holding time. This can lead to waste of resources and excessive number of rerouting decisions. In this paper, a new preemption policy is proposed and complemented with an adaptive scheme that aims to minimize rerouting. The new policy combines the three main preemption optimization criteria: number of LSPs to be preempted, priority of the LSPs, and preempted bandwidth. Weights can be configured to stress the desired criteria. The new policy is complemented by an adaptive scheme that selects lower priority LSPs that can afford to have their rate reduced. The selected LSPs will fairly reduce their rate in order to accommodate the new high-priority LSP setup request. Performance comparisons of a nonpreemptive approach, a policy currently in use by commercial routers, and our policies are also investigated.**

***Index Terms*—DiffServ, MPLS networks, preemption, Traffic Engineering (TE).**

## I. Introduction

THE bandwidth reservation and management problem is one of the most actively studied open issues in several areas of communication networks. The objective is to maximize the network resources utilization while minimizing the number of connections that would be denied access to the network due to insufficient resource availability. Load balancing is another important issue. It is undesirable that portions of the network become overutilized and congested, while alternate feasible paths remain underutilized. These issues are addressed by Traffic Engineering (TE) [1].

Existing TE strategies do not allow different bandwidth constraints for different classes of traffic to be considered in constraint based routing decisions. Only a single bandwidth constraint is considered for all classes, which may not satisfy the needs of individual classes. Where fine-grained optimization of resources is sought, it is a must to perform TE at a per-class rather than a per-aggregate level in order to improve network performance and efficiency [2].

The Multiprotocol Label Switching (MPLS) technology is a suitable way to provide TE [1]. However, MPLS by itself cannot provide service differentiation, which brings up the need to complement it with another technology capable of providing such feature: the Differentiated Services (DiffServ). By mapping the traffic from a given DiffServ class of service on a separate MPLS Label Switched Path (LSP), DiffServ-aware MPLS networks can meet engineering constraints which are specific to the given class on both shortest and nonshortest path. This TE strategy is called DiffServ-aware Traffic Engineering (DS-TE) [2].

In [1], issues and requirements for Traffic Engineering in an MPLS network are highlighted. In order to address both traffic-oriented and resource-oriented performance objectives, the authors point out the need for *priority* and *preemption* parameters as TE attributes of traffic trunks. A *traffic trunk* is defined as an aggregate of traffic flows belonging to the same class which are placed inside an LSP [2]. In this context, *preemption* is the act of selecting an LSP which will be removed from a given path in order to give room to another LSP with a higher *priority*. More specifically, the *preemption attributes* determine whether an LSP with a certain *setup preemption priority* can preempt another LSP with a lower *holding preemption priority* from a given path, when there is a competition for available resources. The preempted LSP may then be rerouted.

Preemption can be used to assure that high-priority LSPs can be always routed through relatively favorable paths within a differentiated services environment. In the same context, preemption can be used to implement various prioritized access policies as well as restoration policies following fault events [1]. Preemption policies have also been recently proposed in other contexts. In [3], the authors developed a framework to implement preemption policies in non-Markovian Stochastic Petri Nets (SPNs). In a computing system context, preemption has been applied in cache-related events. In [4], a technique to bound cache-related preemption delay was proposed. Finally, in the wireless mobile networks framework, preemption has been applied to handoff schemes [5].

Although not a mandatory attribute in the traditional IP world, preemption becomes indeed a more attractive strategy in a differentiated services scenario [6], [7]. Moreover, in the

Manuscript received December 4, 2002; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor R. Govindan. This work was supported in part by NASA Goddard and Swales Aerospace under contract number S11201 (NAS5-01090). A preliminary version of this paper appeared in the Proceedings of IEEE INFOCOM 2002, New York, NY, June 23-27, 2002.

J. C. de Oliveira is with the Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA 19104-2875 USA (e-mail: jau@ece.drexel.edu).

C. Scoglio and I. F. Akyildiz are with the Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: caterina@ece. gatech.edu; ian@ee.gatech.edu).

G. Uhl is with the Swales Aerospace and NASA Goddard Space Flight Center, Beltsville, MD 20705 USA (e-mail: uhl@rattler.gsfc.nasa.gov).

Digital Object Identifier 10.1109/TNET.2004.833156

emerging optical network architectures, preemption policies can be used to reduce restoration time for high priority traffic trunks under fault conditions [1]. Nevertheless, in the DS-TE approach, whose issues and requirements are discussed in [2], the preemption policy is again considered an important piece on the bandwidth reservation and management puzzle, but no preemption strategy is defined.

In this paper, a new preemption policy is proposed and complemented with an adaptive scheme that aims to minimize rerouting. The preemption policy (V-PREPT) is versatile, simple, and robust, combining the three main preemption optimization criteria: number of LSPs to be preempted, priority of LSPs to be preempted, and amount of bandwidth to be preempted. Using V-PREPT, a service provider can balance the objective function that will be optimized in order to stress the desired criteria. V-PREPT is complemented by an adaptive scheme, called Adapt-V-PREPT. The new adaptive policy selects lower priority LSPs that can afford to have their rate reduced. The selected LSPs will fairly reduce their rate in order to accommodate the new high-priority LSP setup request. Heuristics for both simple preemption policy and adaptive preemption scheme are derived and their accuracies are analyzed. Performance comparisons among a nonpreemptive approach, V-PREPT, Adapt-V-PREPT, and a policy purely based on priority and holding time are also provided.

The rest of this paper is organized as follows. In Section II, we introduce the preemption problem. A mathematical formulation, a simple heuristic, and simulation results for the proposed new policy, V-PREPT, are discussed in Section III. In Section IV, we propose Adapt-V-PREPT. A mathematical formulation for its optimization problem and heuristic are included in this section, as well as example results. Performance evaluation of V-PREPT, Adapt-V-PREPT, and a nonpreemptive approach are discussed in Section V. Section V also includes the time complexity analysis of the proposed policies. Finally, the paper is concluded in Section VI.

## II. MPLS Technology and Preemption Problem Formulation

The basic idea behind MPLS is to attach a short fixed-length label to packets at the ingress router of the MPLS domain. These edge routers are called *Label Edge Routers* (LERs), while routers which are capable of forwarding both MPLS and IP packets are called *Label Switching Routers* (LSRs). The packets are then routed based on the assigned label rather than the original packet header. The label assignments are based on the concept of *Forwarding Equivalent Class* (FEC). According to this concept packets belonging to the same FEC are assigned the same label and generally traverse through the same path across the MPLS network. An FEC may consist of packets that have common ingress and egress nodes, or same service class and same ingress/egress nodes, etc. A path traversed by an FEC is called a *Label Switched Path* (LSP). The *Label Distribution Protocol* (LDP) and an extension to the *Resource Reservation Protocol* (RSVP) are used to establish, maintain (refresh), and teardown LSPs [8]. More details on MPLS and DiffServ can be found in [8] and [9].

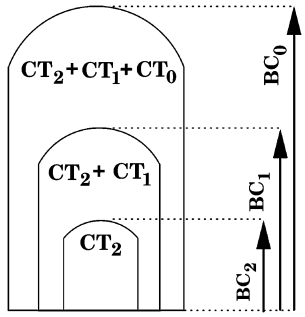


Fig. 1. Russian Doll model with three active CTs.

In this section we present the preemption problem formulation in a DS-TE context. The fundamental requirement for DS-TE is to be able to enforce different bandwidth constraints for different sets of traffic classes. In [2], the definition of Class Type (CT), previously formulated in [10], is refined into the following: the set of traffic trunks crossing a link in which a specific set of bandwidth constraints is enforced.

DS-TE may support up to eight CTs: $\mathrm{CT}_c$, $c = 0, \ldots, 7$. By definition, each CT is assigned either a Bandwidth Constraint (BC), or a set of BCs. Therefore, DS-TE must support up to eight BCs: $\mathrm{BC}_b$, $b = 0, \ldots, 7$. However, the network administrator does not need to always deploy the maximum number of CTs, but only the ones actually in use.

The *Russian Doll Model* (RDM) [11] is under discussion in the IETF Traffic Engineering Working Group for standardization in the requirements for DiffServ MPLS TE draft ([2], to become RFC). Other models have been proposed, such the the *Maximum Allocation Model* (MAM) [12], and *Maximum Allocation with Reservation* (MAR) [13]. In [14], the author compares the three models and concludes that RDM is a better match to DS-TE objectives and recommends the selection of RDM as the default model for DS-TE.

The Russian Doll Model may be defined as follows [11]:

- Maximum number of BCs is equal to maximum number of $\mathrm{CTs} = 8$;
- All LSPs from $\mathrm{CT}_c$ must use no more than $\mathrm{BC}_b$ (with $b \leq c \leq 7$, and $\mathrm{BC}_b \leq \mathrm{BC}_{b-1}$, for $b = 1, \ldots, 7$), i.e.,:

— All LSPs from $\mathrm{CT}_7$ use no more than $\mathrm{BC}_7$;
— All LSPs from $\mathrm{CT}_6$ and $\mathrm{CT}_7$ use no more than $\mathrm{BC}_6$;
— All LSPs from $\mathrm{CT}_5$, $\mathrm{CT}_6$, and $\mathrm{CT}_7$ use no more than $\mathrm{BC}_5$;
— :
— All LSPs from $\mathrm{CT}_0$, $\mathrm{CT}_1$, $\mathrm{CT}_2$, $\mathrm{CT}_3$, $\mathrm{CT}_4$, $\mathrm{CT}_5$, $\mathrm{CT}_6$, and $\mathrm{CT}_7$ use no more than $\mathrm{BC}_0$.

To illustrate the model, assume only three CTs are activated in a link and the following BCs are configured: $\mathrm{BC}_0 = 100$, $\mathrm{BC}_1 = 80$, and $\mathrm{BC}_2 = 50$. Fig. 1 shows the model in a pictorial manner (nesting dolls). $\mathrm{CT}_0$ could be representing the best-effort traffic, while $\mathrm{CT}_1$ the nonreal-time traffic, and $\mathrm{CT}_2$ the real-time traffic. Following the model, $\mathrm{CT}_0$ could use up to 100% of the link capacity given that no $\mathrm{CT}_1$ or $\mathrm{CT}_2$ traffic would be present in that link. Once $\mathrm{CT}_1$ comes into play, it would be able to occupy up to 80% of the link, and $\mathrm{CT}_0$ would be reduced to 20%. Whenever $\mathrm{CT}_2$ traffic would also be routed in that link, $\mathrm{CT}_2$ would then be able to use up to 50% by itself,

TABLE I
TE-CLASSES AND RESERVED BANDWIDTHS FOR EXAMPLE

| TE-Class | TE-Class Attributes | Reserved BW |
|---|---|---|
| TE-Class 0 | $\mathrm{CT}_1, p = h = 0$ | $B = 25$ |
| TE-Class 1 | $\mathrm{CT}_0, p = h = 1$ | $B = 20$ |
| TE-Class 2 | $\mathrm{CT}_1, p = h = 2$ | $B = 25$ |
| TE-Class 3 | $\mathrm{CT}_0, p = h = 3$ | $B = 20$ |

$\mathrm{CT}_1$ would be able to use up to 30% by itself, while $\mathrm{CT}_0$ could use up to 20% alone.

Two preemption attributes were defined in [1]: *setup preemption priority*, $s$, and *holding preemption priority*, $h$. These parameters may be configured as having the same value or different values and must work across Class Types, i.e., if LSP1 contends for resources with LSP2, LSP1 may preempt LSP2 if LSP1 has a higher setup preemption priority (lower numerical value) than LSP2's holding preemption priority, regardless of their CTs.

In [2], the TE-Class concept is defined. A TE-Class is composed of a unique pair of information—a Class Type, $\mathrm{CT}_c$, and the preemption priority assigned to that Class Type, $p$, which can be used as the setup preemption priority ($s = p$), as the holding preemption priority ($h = p$), or both ($s = h = p$):

$$\text{TE-Class } i = \langle \mathrm{CT}_c, p \rangle$$

where $0 \le i \le 7$, $0 \le c \le 7$, $0 \le p \le 7$.

By definition there may be more than one TE-Class using the same CT, as long as each TE-Class uses a different preemption priority. Also, there may be more than one TE-Class with the same preemption priority, provided that each TE-Class uses a different CT. The network administrator may define the TE Classes in order to support preemption across CTs, to avoid preemption within a certain CT, or to avoid preemption completely, when so desired. To ensure coherent operation, the same TE Classes must be configured in every Label Switched Router (LSR) in the DS-TE domain.

As a consequence of this per-TE-Class treatment, the Interior Gateway Protocol (IGP) needs to advertise separate TE information for each TE-Class, which consists of the *Unreserved Bandwidth* (UB) information [15]. The UB information will be used by the routers, checking against the Russian Doll parameters, to decide whether to preempt an LSP.

Following the example in [15] on how to compute $\mathrm{UB}_i$, (UB [TE-Class $i$]), we assume that the Russian Doll bandwidth constraint model is in use. We define $B(\mathrm{CT}_c, h)$ as the total bandwidth reserved for all LSPs belonging to $\mathrm{CT}_c$ and that have a *holding preemption priority* of $h$. The unreserved bandwidth (UB) for each TE-Class $i = \langle \mathrm{CT}_c, p \rangle$ can be calculated using the following formula:

$$\mathrm{UB}_i = \min[\mathrm{BC}_b - \sum B(\mathrm{CT}_c, h) \text{ for } h \le p \text{ and } b \le c \le 7, \ldots, \mathrm{BC}_0 - \sum B(\mathrm{CT}_c, h) \text{ for } h \le p \text{ and } 0 \le c \le 7].$$

For example, suppose a link with 100 Mb/s and only four active TE Classes, as shown in Table I. Also, suppose a Russian Doll bandwidth constraint model with $\mathrm{BC}_0 = 100$ and $\mathrm{BC}_1 = 50$. Using the above-described formula we calculate

$$\begin{aligned}
\mathrm{UB}_0 &= \min[\mathrm{BC}_1 - B(\mathrm{CT}_1{,}0), \mathrm{BC}_0 - B(\mathrm{CT}_1, 0)] \\
&= \min[50 - 25{,}100 - 25] = 25 \\
\mathrm{UB}_1 &= \min[\mathrm{BC}_0 - \{B(\mathrm{CT}_1{,}0) + B(\mathrm{CT}_0, 1)\}] \\
&= \min[100 - \{25 + 20\}] = 55 \\
\mathrm{UB}_2 &= \min[\mathrm{BC}_1 - \{B(\mathrm{CT}_1, 0) + B(\mathrm{CT}_1, 2)\}, \mathrm{BC}_0 \\
&\qquad - \{B(\mathrm{CT}_1, 0) + B(\mathrm{CT}_0, 1) + B(\mathrm{CT}_1, 2)\}] \\
&= \min[50 - \{25 + 25\}, 100 - \{25 + 20 + 25\}] \\
&= 0 \\
\mathrm{UB}_3 &= \min[\mathrm{BC}_0 - \{B(\mathrm{CT}_1, 0) + B(\mathrm{CT}_0, 1) \\
&\qquad + B(\mathrm{CT}_1, 2) + B(\mathrm{CT}_0, 3)\}] \\
&= \min[100 - \{25 + 20 + 25 + 20\}] = 10.
\end{aligned}$$

Note that a new LSP setup request from TE-Class 0 could be accepted, if it requires less than $\mathrm{UB}_0$, preempting bandwidth from TE-Class 1, or TE-Class 2, or TE-Class 3, or from any combination of them. A new LSP setup request belonging to TE-Class 1 could be accepted, if it requires less than $\mathrm{UB}_1$, preempting bandwidth from LSPs from TE-Class 2, or TE-Class 3, or from both. A new LSP setup request belonging to TE-Class 2 would be rejected since the whole $\mathrm{BC}_1$ is already in use. A new LSP setup request from TE-Class 3 could only be accepted if it requires less than $\mathrm{UB}_3$, since LSPs from TE-Class 3 cannot preempt any other LSPs.

It is important to mention that preemption can be reduced if an alternative shortest-path route (e.g., second or third shortest-path) can be considered. Even in that case, preemption may be needed as such path options may also be congested. In a fixed shortest-path routing approach, preemption would happen more frequently.

In the case in which preemption will occur, a preemption policy should be activated to find the preemptable LSPs with lower preemption priorities. Now an interesting question arises: which LSPs should be preempted? Running preemption experiments using CISCO routers (7204VXR and 7505, OS version 12.2.1), we could conclude that the preempted LSPs were always the ones with the lowest priority, even when the bandwidth allocated was much larger than the one required for the new LSP. This policy would result in high bandwidth wastage for cases in which rerouting is not allowed. An LSP with a large bandwidth share might be preempted to give room to a higher priority LSP that requires a much lower bandwidth.

A new LSP setup request has two important parameters: bandwidth and preemption priority. In order to minimize wastage, the set of LSPs to be preempted can be selected by optimizing an objective function that represents these two parameters, and the number of LSPs to be preempted. More specifically, the objective function could be any or a combination of the following [6], [7], [16].

1) Preempt the connections that have the least priority (preemption priority). The QoS of high priority traffic would be better satisfied.
2) Preempt the least number of LSPs. The number of LSPs that need to be rerouted would be lower.

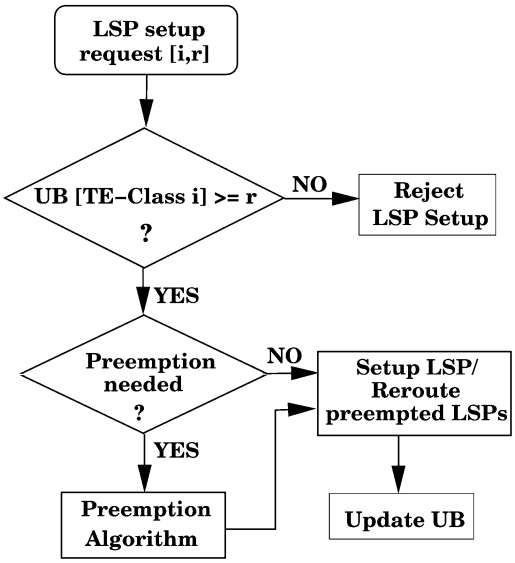


Fig. 2. Flowchart for LSP setup procedure.

3) Preempt the least amount of bandwidth that still satisfies the request. Resource utilization would be improved.

After the preemption selection phase is finished, the selected LSPs must be torn down (and possibly rerouted), releasing the reserved bandwidth. The new LSP is established, using the currently available bandwidth. The UB information is then updated. Fig. 2 shows a flowchart that summarizes how each LSP setup request is treated in a preemption enabled scenario.

In [16], the authors propose connection preemption policies that optimize the discussed criteria in a given order of importance: number of connections, bandwidth, and priority; and bandwidth, priority, and number of connections. In [17], a scheduling heuristic that takes into account the bandwidth and priority of bandwidth allocation requests is proposed. The scheduling heuristic can preempt, or degrade in a continuous manner, already scheduled requests. A request is characterized by a certain amount of bandwidth, a start time, an end time, a priority level, and a utility function chosen between a step function, a concave function or a linear function. The novelty in our approach is to propose an objective function that can be adjusted by the service provider in order to stress the desired criteria. No particular criteria order is enforced. Moreover, our preemption policy is complemented by an adaptive rate scheme. The resulting policy, reduces the number of preempted LSPs by adjusting the rate of selected low-priority LSPs that can afford to have their rate reduced in order to accommodate a higher priority request. This approach minimizes service disruption, and rerouting decisions. To the best of our knowledge, such a comprehensive solution for the preemption problem has not been investigated before.

## III. V-PREPT: A Versatile Preemption Policy

In this section, a mathematical formulation for V-PREPT is presented, a simple heuristic is proposed, and simulations results are shown to compare both approaches. Considerations about how to implement V-PREPT to preempt resources on a path rather than on a link are discussed next.

### A. Preempting Resources on a Path

It is important to note that once a request for an LSP setup arrives, the routers on the path to be taken by the new LSP need to check for bandwidth availability in all links that compose the path. For the links in which the available bandwidth is not enough, the preemption policy needs to be activated in order to guarantee the end-to-end bandwidth reservation for the new LSP. This is a decentralized approach, in which every node on the path would be responsible to run the preemption algorithm and determine which LSPs would be preempted in order to fit the new request. A decentralized approach may sometimes not lead to an optimal solution.

In another approach, a "manager entity" runs the preemption policy and determines the best LSPs to be preempted in order to free the required bandwidth in all the links that compose the path. A unique LSP may be already set in between several nodes on that path, and the preemption of that LSP would free the required bandwidth in many links that compose the path.

Both centralized and decentralized approaches have their advantages and drawbacks. A centralized approach is more precise, but requires that the whole network state be stored and updated accordingly, which raises scalability issues. In a network where LSPs are mostly static, an off-line decision can be made to reroute LSPs and the centralized approach could be appropriate. However, in a dynamic network in which LSPs are setup and torn down in a frequent manner, the correctness of the stored network state could be questionable. In this scenario, the decentralized approach would bring more benefits, even when resulting in a nonoptimal solution. A distributed approach is also easier to be implemented due to the distributed nature of the current Internet protocols.

Since the current Internet routing protocols are essentially a distributed approach, we chose to use a decentralized LSP preemption policy. The parameters required by our policies are currently available for protocols such as OSPF or are easy to be determined.

### B. Mathematical Formulation

We formulate our preemption policy, V-PREPT, with an integer optimization approach. Consider a request for a new LSP setup with bandwidth $b$ and setup preemption priority $p$. When preemption is needed, due to lack of available resources, the preemptable LSPs will be chosen among the ones with lower holding preemption priority (higher numerical value) in order to fit $r = b - A_{\mathrm{bw}}(l)$. The constant $r$ represents the actual bandwidth that needs to be preempted (the requested, $b$, minus the available bandwidth on link $l : A_{\mathrm{bw}}(l)$).

Without loss of generality, we assume that bandwidth is available in bandwidth modules, which implies that variables such as $r$ and $b$ are integers.

Define $\mathcal{L}$ as the set of active LSPs having a holding preemption prioritylower (numerically higher) than $p$. We denote the cardinality of $\mathcal{L}$ by $L$. $b(l)$ is the bandwidth reserved by LSP $l \in \mathcal{L}$, expressed in bandwidth modules and $p(l)$ is the holding preemption priority of LSP $l$.

In order to represent a cost for each preemption priority, we define an associated cost $y(l)$ inversely related to the holding

| | |
|---|---|
| GIVEN | $\mathcal{L}, \mathbf{b}, \mathbf{y}, r, \alpha, \beta, \gamma$. |
| FIND | $\mathbf{z}$ ($L$ binary integer variables) |
| MINIMIZING | $F(\mathbf{z}) = \alpha(\mathbf{z} \cdot \mathbf{y^T}) + \beta(\mathbf{z} \cdot \mathbf{1^T}) + \gamma(\mathbf{z} \cdot \mathbf{b^T})$ |
| SUBJECT TO | $\mathbf{z} \cdot \mathbf{b^T} \geq \mathbf{r}$ |

Fig. 3. V-PREPTs optimization formulation.

preemption priority $p(l)$. For simplicity, we choose a linear relation $y(l) = 8 - p(l)$. We define $\mathbf{y}$ as a cost vector with $L$ components, $y(l)$. We also define $\mathbf{b}$ as a reserved bandwidth vector with dimension $L$, and components $b(l)$.

The vector $\mathbf{z}$ is the optimization variable. $\mathbf{z}$ is composed of $L$ binary variables, each defined as follows:

$$z(l) = \begin{cases} 1, & \text{if LSP } l \text{ is preempted} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

For example, assume there exist three LSPs, $\mathcal{L} = (l_1, l_2, l_3)$, with reserved bandwidth of 3 Mb/s, 2 Mb/s, and 1 Mb/s, respectively. Consider that $p(l_1) = 3$, $p(l_2) = 6$, $p(l_3) = 7$. Consequently, $y(l_1) = 5$, $y(l_2) = 2$, $y(l_3) = 1$, $\mathbf{y} = [5, 2, 1]$, and $\mathbf{b} = [3, 2, 1]$. $\mathbf{z} = [0, 1, 1]$ means that LSPs $l_2$ and $l_3$ are chosen to be preempted.

Concerning the objective function, as reported in the Section III-A, three main objectives can be reached in the selection of preempted LSPs:

- minimize the priority of preempted LSPs,
- minimize the number of preempted LSPs,
- minimize the preempted bandwidth.

To have the widest choice on the overall objective that each service provider needs to achieve, we define the following objective function $F$, which for simplicity is chosen as a weighted sum of the above-mentioned criteria:

$$F(\mathbf{z}) = \alpha(\mathbf{z} \cdot \mathbf{y^T}) + \beta(\mathbf{z} \cdot \mathbf{1^T}) + \gamma(\mathbf{z} \cdot \mathbf{b^T}) \tag{2}$$

where the term $\mathbf{z} \cdot \mathbf{y^T}$ represents the preemption priority of preempted LSPs, $\mathbf{z} \cdot \mathbf{1^T}$ represents the number of preempted LSPs ($\mathbf{1}$ is a unit vector with adequate dimension), and $\mathbf{z} \cdot \mathbf{b^T}$ represents the total preempted capacity. Coefficients $\alpha$, $\beta$, and $\gamma$ are suitable weights that can be configured in order to stress the importance of each component in $F$.

The following constraint ensures that the preempted LSP's release enough bandwidth to satisfy the new request:

$$\mathbf{z} \cdot \mathbf{b^T} \geq \mathbf{r}. \tag{3}$$

Fig. 3 contains a summary of the proposed integer program for our preemption policy, named V-PREPT.

### C. Heuristic

The choice of LSPs to be preempted is known to be an NP-complete problem [18]. For networks of small and medium size, or for a small number of LSPs, the online use of an optimization tool is a fast and accurate way to implement V-PREPT. However, for large networks and large number of LSPs, a simple heuristic that could approximate the optimal result would be preferable.

```
Algorithm V-PREPT (L, b, y, r, α, β, γ)
preempt = 0;
if Σ_∀i b(i) >= r, {
   Calculate H(i) = αy(i) + β(1/b(l)) + γ(b(i) − r)^2;
   Sort H, b, in increasing order of H(i);
   If H has repeated values, sort those by increasing b;
   while preempt < r {
      if b(i) < r {
         if H does not have repeated values {
            preempt = preempt + b(i); z(i) = 1; i = i + 1;
         else
            if b(i) >= r for any repeated H(i)
               preempt = b(i); % preempt only LSP i
               z(i) = 1; ∀z(n)_{n≠i} = 0; break;
            elseif preempt + b(i) > r for any repeated H(i)
               preempt = preempt + b(i);
               z(i) = 1; break;
            else {
               j = larger i in repeated H(i); k = j + 1;
               while (preempt >= r) or (j ≥ i) {
                  preempt = preempt + b(j);
                  z(j) = 1; j = j − 1;}
               i = k;}}
      else
         preempt = b(i);
         z(i) = 1; ∀z(n)_{n≠i} = 0; break;}}
else
   reject LSP setup request
}
return(z)
```

Fig. 4. Heuristic for V-PREPT.

In order to simplify the online choice of LSPs to be preempted, we propose the following equation, used in V-PREPT's heuristic (Fig. 4):

$$H(l) = \alpha y(l) + \beta\left(\frac{1}{b(l)}\right) + \gamma(b(l) - r)^2. \tag{4}$$

In this equation, $\alpha y(l)$ represents the cost of preempting LSP $l$, $\beta(1/b(l))$ represents the choice of a minimum number of LSPs to be preempted in order to fit the request $r$, and $\gamma(b(l) - r)^2$ penalizes a choice of an LSP to be preempted that would result in high bandwidth wastage.

In V-PREPT's heuristic, $H$ is calculated for each LSP. The LSPs to be preempted are chosen as the ones with smaller $H$ that add enough bandwidth to accommodate $r$. The respective components in the vector $\mathbf{z}$ are made equal to one for the selected LSPs.

In case $H$ contained repeated values, the sequence of choice follows the bandwidth $b$ reserved for each of the regarded LSPs, in increasing order. For each LSP with repeated $H$, we test whether the bandwidth $b$ assigned to that LSP only is enough to satisfy $r$. If there is no such LSP, we test whether the bandwidth of each of those LSPs, added to the previously preempted LSPs' bandwidth is enough to satisfy $r$. If that is not true for any LSP in that repeated $H$ value sequence, we preempt the LSP that has the larger amount of bandwidth in the sequence, and keep preempting in decreasing order of $b$ until $r$ is satisfied or the sequence is finished. If the sequence is finished and $r$ is not satisfied, we again select LSPs to be preempted based on an increasing order of $H$. More details on the algorithm to implement V-PREPT's heuristic are shown in Fig. 4.

TABLE II
BANDWIDTH AND COST INFORMATION FOR SIMULATIONS

| LSP | $l_1$ | $l_2$ | $l_3$ | $l_4$ | $l_5$ | $l_6$ | $l_7$ | $l_8$ |
|---|---|---|---|---|---|---|---|---|
| Bandwidth ($b$) | 20 | 10 | 60 | 25 | 20 | 1 | 75 | 45 |
| Priority ($p$) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 5 |
| Cost ($y$) | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 3 |

| LSP | $l_9$ | $l_{10}$ | $l_{11}$ | $l_{12}$ | $l_{13}$ | $l_{14}$ | $l_{15}$ | $l_{16}$ |
|---|---|---|---|---|---|---|---|---|
| Bandwidth ($b$) | 100 | 5 | 40 | 85 | 50 | 20 | 70 | 25 |
| Priority ($p$) | 3 | 6 | 4 | 5 | 2 | 3 | 4 | 7 |
| Cost ($y$) | 5 | 2 | 4 | 3 | 6 | 5 | 4 | 1 |

The output of the algorithm is $\mathbf{z}$, which contains the information about which LSPs are to be preempted and the variable *preempt* contains the amount of bandwidth preempted. In a much larger network, our heuristic would still be very simple to compute when compared to the optimization problem described in (2) and (3).

### D. Example

Consider a network composed of 16 LSPs with reserved bandwidth $b$ in Mb/s, preemption holding priority $p$, and cost $y$, as shown in Table II. In this example, eight TE-Classes are active.

Suppose the network operator decides to configure $\beta = 0$, indicating that the number of LSPs preempted is not important (rerouting is allowed and not expensive: small topology), $\alpha = 1$ and $\gamma = 1$, indicating that preemption priority and preempted bandwidth are more important.

A request for an LSP establishment arrives with $r = 155$ Mb/s and $p = 0$ (highest possible priority, which implies that all LSPs with $p > 0$ in Table II will be considered when running the algorithms). From (2) and (3), we formulate the following optimization problem:

Minimize $F(\mathbf{z}) = (\mathbf{z} \cdot \mathbf{y^T}) + (\mathbf{z} \cdot \mathbf{b^T})$,

subject to $\mathbf{z} \cdot \mathbf{b^T} \geq \mathbf{155}$, with $\mathbf{y}$ and $\mathbf{b}$ defined as in Table II.

Using an optimization tool to solve the above optimization problem, one will find that LSPs $l_8$, $l_{12}$, and $l_{16}$ are selected for preemption.

Suppose the network operator decides that it is more appropriate to configure $\alpha = 1$, $\beta = 1$, and $\gamma = 0$, because in this network rerouting is now cheaper, LSP priority is again very important, but bandwidth is not a critical issue. The optimization problem now becomes:

Minimize $F(\mathbf{z}) = (\mathbf{z} \cdot \mathbf{y^T}) + (\mathbf{z} \cdot \mathbf{1^T})$,

subject to $\mathbf{z} \cdot \mathbf{b^T} \geq \mathbf{155}$, in which case, LSPs $l_7$ and $l_{12}$ are selected for preemption.

To take into account the number of LSPs preempted, the preemption priority, and the amount of bandwidth preempted, the network operator may set $\alpha = \beta = \gamma = 1$. In that case, LSPs $l_{12}$ and $l_{15}$ are selected.

From the above example we can observe that when the number of LSPs preempted was not an issue, three LSPs adding exactly the requested bandwidth, and with the lowest priority were selected. When a possible waste of bandwidth was not an issue, two LSPs were selected, adding more bandwidth than requested, but with lower preemption priority. Considering the three factors as crucial, two LSPs are preempted, and in this case adding exactly 155 Mb/s with the lowest possible preemption priorities.

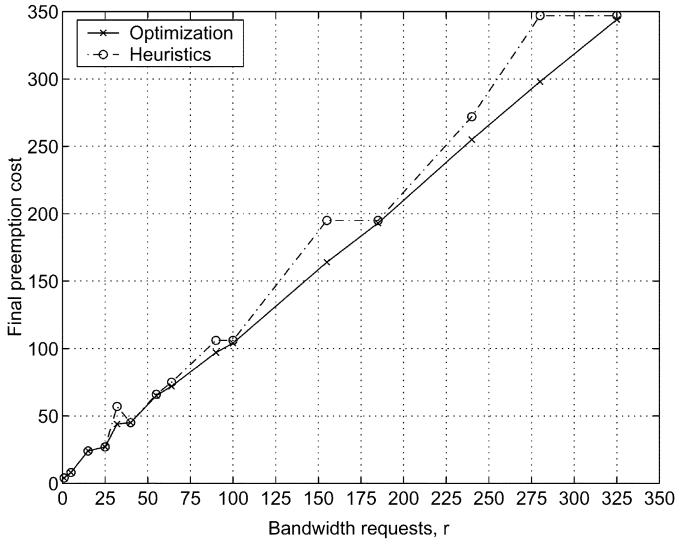


Fig. 5. Comparison between V-PREPT's optimization formulation and heuristic.

If a balance amongst the three objectives is sought, the coefficients $\alpha$, $\beta$, and $\gamma$ need to be configured in a proper manner. In our example, $\alpha$ is multiplying a term that could be any value between 1 and $\sum \mathbf{y}$ (1 and 60), $\beta$ is multiplying a number between 1 and $L$ (total number of LSPs in the link: 1 and 16), and $\gamma$ is multiplying a number between $\min b$ and $\sum b$ (1 and 651). It is very likely that neither the number multiplied by $\alpha$ nor the number multiplied by $\beta$ will be large when compared to the number multiplied by $\gamma$, which will be of the order of $r$. Depending on the value of $r$, the $\gamma$ factor in the objective function can be quite large when compared to the other two terms. As an example, assume a request arrives for an LSP requesting $r = 90$. If only priority is selected as the most important criteria ($\alpha = 1, \beta = \gamma = 0$), LSPs $l_7$ and $l_{16}$ would be selected for preemption. When number of preempted LSPs would be the criteria of consideration ($\beta = 1, \alpha = \gamma = 0$), LSP $l_9$ would be selected, releasing 100 Mb/s. If bandwidth is the only important criteria, LSPs $l_5$ and $l_{15}$ could be selected, adding exactly 90 Mb/s. Following our previous analysis, the coefficients could be selected as follows, when a balance is sought: $\alpha = 1$, $\beta = 1$ and $\gamma = 0.01$. In that case, two LSPs would be selected for preemption, LSPs $l_7$ and $l_{16}$, adding 100 Mb/s, but both with the least priority. We analyzed the sensitivity of the objective function to the coefficients, and determined that, in this case, the same LSPs were selected for preemption when $\alpha \geq 0.35$, $\beta \leq 3$, and $\gamma \leq 0.3$.

Using the same data as in Table II, and with $\alpha = \beta = \gamma = 1$, we varied the value of the request $r$ and compared the results found by V-PREPT's optimization formulation (Fig. 3) and heuristic (Fig. 4), regarding the final cost achieved, calculated by (2). Fig. 5 shows the result of these tests.

Figs. 6–8 show results for V-PREPT's heuristic and optimization problem when only the preemption priority, only the number of LSPs preempted, or only the amount of preempted bandwidth is important, respectively.

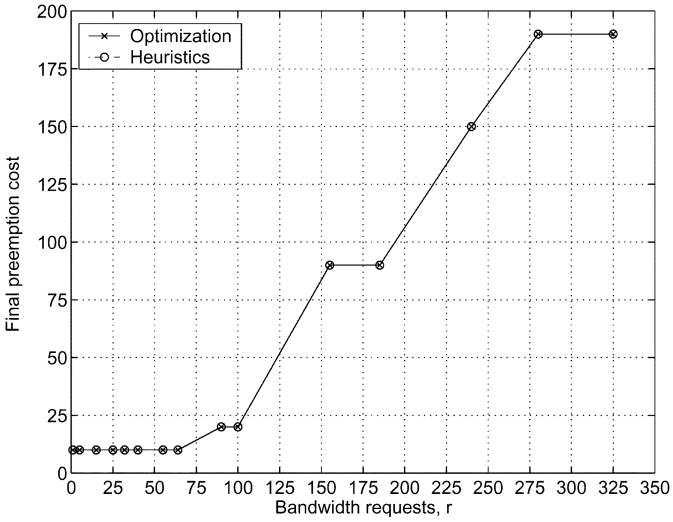


Fig. 6. V-PREPT's optimization and heuristic when $\alpha = 10, \beta = 0, \gamma = 0$.

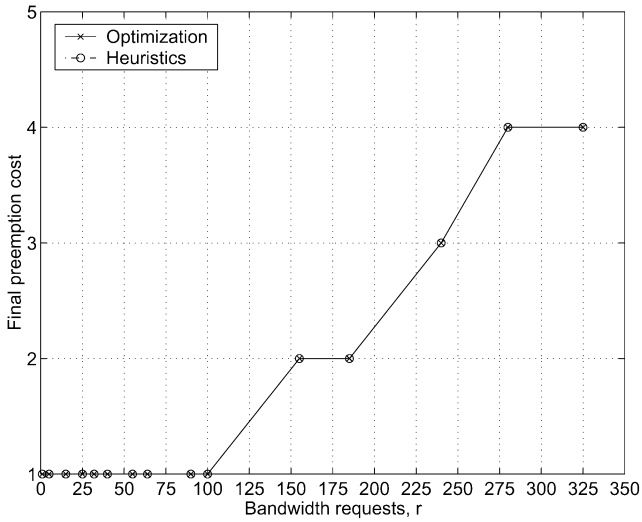


Fig. 7. V-PREPT's optimization and heuristic when $\alpha = 0, \beta = 1, \gamma = 0$.

Figs. 6 and 7 show the perfect accuracy of V-PREPT's heuristic for the considered cases. The results in Fig. 5 and in Fig. 8 show that the heuristic finds a similar solution for most of the cases, and that when $r$ increases the heuristic leads to a slightly higher cost—a price paid because the heuristic follows a much simpler approach. When comparing the bandwidth request $r$ to the already setup bandwidth reservations, we observe that when $r$ is comparable to one or two LSP reservations (which is a more likely scenario) the heuristic always finds a similar cost solution. The zig-zag effect on the graphic is due to the preemption of LSPs that add more bandwidth than the request $r$, which increases the value found by the cost function. When the next request $r$ is considered, and the new selected LSPs add exactly or about the same value as the new $r$, the cost function is reduced, therefore the zig-zag occurs.

Fig. 9 shows V-PREPT's optimal and heuristic results for the preemption cost when 200 LSPs share a link in which preemption needs to be performed. The parameters $\alpha, \beta$, and $\gamma$ were set to unit values. The LSPs' bandwidth varied from 1 Mb/s to 100 Mb/s. The results (for several values of bandwidth request $r$) corroborate the previous conclusions about the heuristic's accuracy.

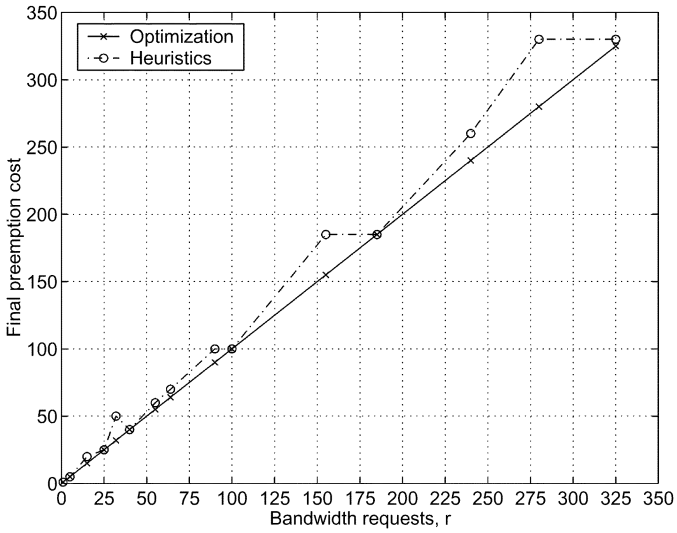


Fig. 8. V-PREPT's optimization and heuristic when $\alpha = 0, \beta = 0, \gamma = 1$.

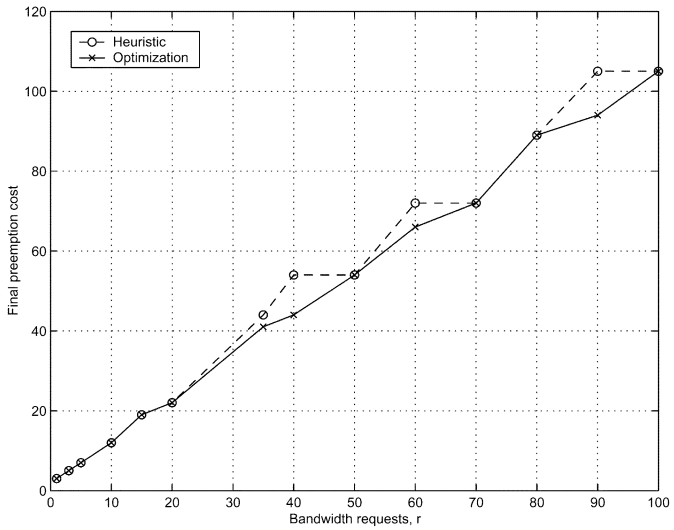


Fig. 9. V-PREPT's results for a link with 200 LSPs, $\alpha = \beta = \gamma = 1$.

## IV. Adapt-V-PREPT: V-PREPT With Adaptive Rate Scheme

In this section we complement V-PREPT with an adaptive rate scheme. In Section III, when a set of LSPs was chosen to be preempted, those LSPs were torn down and could be rerouted, which implied extra signaling and routing decisions. In order to avoid or minimize rerouting, we propose to reduce the number of preempted LSPs by selecting a few low-priority LSPs that would have their rate reduced by a certain maximum percentage in order to accommodate the new request. After an LSP is selected for rate reduction, there will be signaling to inform the originating LSR of the rate reduction. However, after the rate reduction is made in the originating LSR, the same RSVP signaling previously used to refresh the LSP will now be used to announce the new rate to every LSR in the LSP route. No additional signaling would be needed, and therefore less signaling effort is necessary overall when compared to tearing down, rerouting and setting up a new LSP. In the future, whenever there exists available bandwidth in the network,

the lowered-rate LSPs would fairly increase their rate to the original reserved bandwidth.

Some applications such as nonreal-time video or data transfer can afford to have their transmission rate reduced, and would be the most likely to be assigned to such TE Classes. By reducing the rate in a fair fashion, the LSPs would not be torn down, there would not be service disruption, extra setup and tear down signaling, or rerouting decisions. In DiffServ, traffic aggregates assigned to the Assured Forward Per-Hop Behavior (AF PHB) would be the natural candidates for rate reduction. Whereas Expedited Forward Per-Hop Behavior (EF PHB) supports services with "hard" bandwidth and jitter guarantees, the AF PHB allows for more flexible and dynamic sharing of network resources, supporting the "soft" bandwidth and loss guarantees appropriated for bursty traffic [9].

Next, we present a mathematical formulation for the new adaptive policy, called Adapt-V-PREPT, followed by a simple heuristic that approximates the results provided by the optimization problem. Simulation results are shown to stress the accuracy of the proposed heuristic. Comparison with V-PREPT's simulation results of Section III are also included, taking into account costs and rewards of both approaches. We again chose a decentralized approach to solve the rate reduction problem on a path (link by link). The policy is individually run in every router that composes the selected path, starting from the destination and going toward the origin. Each decision is locally taken by the respective router, avoiding race conditions.

### A. Mathematical Formulation

Similarly to V-PREPT, again we formulate the preemption policy as an integer optimization problem.

We assume that bandwidth is available in bandwidth modules, and define $\mathcal{L}'$ (cardinality $L'$) as a set of active LSPs with holding preemption priority lower than the setup preemption priority of the new LSP, and that can afford to have their rate reduced. Therefore, $\mathcal{L}' \subset L$. The parameters $r$, $b$, and $p$ have the same context as in Section III.

We define $M$ as the total number of bandwidth modules allocated to LSPs that can be preempted or have their rate reduced:

$$M = \sum_{l=1}^{L'} b(l). \tag{5}$$

We also define vector $\mathbf{v^l}$ with $M$ components, representing the bandwidth modules reserved by an active LSP $l \in \mathcal{L}'$: $\mathbf{v^l} = (v_1^l, v_2^l, \ldots, v_M^l)$, where

$$v_m^l = \begin{cases} 1, & \text{if module } m \text{ belongs to } l \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

$\mathbf{V}$ is a matrix, $\mathcal{L}' \times M$, composed of vectors $\mathbf{v^l}$. We define the vector $\mathbf{B}$ as $\mathbf{B} = \mathbf{V} \cdot \mathbf{b}$.

We again define $\mathbf{y}$ as a priority vector, now with $M$ components, where each component $y(m)$ is the priority $y$ of the bandwidth module $m$. Every bandwidth module of an LSP has the same cost value, which implies that $\mathbf{y}$ is composed of a series of repeated values (as many as the number of modules in each LSP). Vectors $\mathbf{x}$ and $\mathbf{z}$ are the variables to be optimized, and are defined as follows.

Vector $\mathbf{x}$ is composed of $M$ binary variables:

$$x(m) = \begin{cases} 1, & \text{if module } m \text{ is preempted} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

A binary component $x(m) = 1$ means that the $m$th bandwidth module is preempted in order to reduce that LSP's rate and make room to satisfy the request of $r$ bandwidth modules.

Vector $\mathbf{z}$ is composed of $L'$ binary variables, and follows the same definition as in Section III, equation (1).

Note that the optimization variables are binary and their total number is $L' + M$.

For example, assume there exists three LSPs that can afford to have their rate reduced, $\mathcal{L}' = (l_1, l_2, l_3)$, with reserved bandwidth of 3 Mb/s, 2 Mb/s, and 1 Mb/s, respectively. Assume bandwidth module of 1 Mb/s. The size of the set of bandwidth modules can be calculated with (5): $M = 6$. Each LSP can be represented by the following bandwidth module vectors (6).

$\mathbf{v^{l_1}} = (1,1,1,0,0,0) \rightarrow$ modules 1, 2, and 3 belong to $l_1$.

$\mathbf{v^{l_2}} = (0,0,0,1,1,0) \rightarrow$ modules 4 and 5 belong to $l_2$.

$\mathbf{v^{l_3}} = (0,0,0,0,0,1) \rightarrow$ module 6 belongs to $l_3$.

Let us assume that $p(l_1) = 3$, $p(l_2) = 6$, $p(l_3) = 7$. Consequently, $y(l_1) = 5$, $y(l_2) = 2$, $y(l_3) = 1$, and $\mathbf{y} = (5,5,5,2,2,1)$.

We define the following new objective function $\mathcal{F}$:

$$\mathcal{F}(\mathbf{x},\mathbf{z}) = \alpha(\mathbf{x}\cdot\mathbf{y^T}) + \beta(\mathbf{z}\cdot\mathbf{1^T}) + \gamma(\mathbf{x}\cdot\mathbf{1^T}) + \mathbf{x}\cdot\left(\frac{1}{\mathbf{B}}\right)^{\mathbf{T}} \tag{8}$$

where $\mathbf{x}\cdot\mathbf{y^T}$ represents the priority of preempted bandwidth modules, $\mathbf{z}\cdot\mathbf{1^T}$ represents the number of preempted LSPs, $\mathbf{x}\cdot\mathbf{1^T}$ represents the total preempted capacity, and $\mathbf{x}\cdot(\mathbf{1/B})^{\mathbf{T}}$ represents the bandwidth module cost per LSP, proportional to the number of modules reserved by the LSP. Coefficients $\alpha$, $\beta$ and $\gamma$ are used for the same purpose as in Section III, equation (2): in order to stress the importance of each component in $\mathcal{F}$.

As for constraints, we must make sure that the bandwidth requirement is met, that all the bandwidth modules from an LSP are made available when that LSP is preempted, that the respective modules for the LSPs that will reduce their rate are also preempted, and that the preempted rate will not be more than $\Delta\%$ of the reserved bandwidth for that LSP.

We represent these constraints as follows, remarking that the *greater than* and *less than* signs are considered in a row-by-row relation between the matrices:

$$\begin{aligned} \mathbf{x}\cdot\mathbf{1^T} &\geq \mathbf{r} && (1 \text{ constraint}) \\ \mathbf{x^T} - (\mathbf{z}\cdot\mathbf{V})^{\mathbf{T}} &\geq 0 && (M \text{ constraints}) \\ \mathbf{z^T} - \mathbf{V}\cdot\mathbf{x^T} &> -(b^T - 1) && (L' \text{ constraints}) \\ \mathbf{V}\cdot\mathbf{x^T} &\leq \frac{\Delta}{100}\left(\text{diag}(\mathbf{b^T}\cdot\mathbf{z}) + \mathbf{b^T}\right) && (L' \text{constraints}) \end{aligned} \tag{9}$$

where diag means the diagonal values of the matrix displayed in a column vector.

The first constraint implies that the number of preempted capacity modules should be equal or greater than the request. The

$$
\begin{array}{ll}
\text{GIVEN} & \mathcal{L}', \mathbf{b}, \mathbf{y}, r, \alpha, \beta, \gamma, \Delta. \\
\text{FIND} & \mathbf{x}, \mathbf{z} \\
\text{MINIMIZING} & \mathcal{F}(\mathbf{x}, \mathbf{z}) = \alpha(\mathbf{x} \cdot \mathbf{y^T}) + \beta(\mathbf{z} \cdot \mathbf{1^T}) + \gamma(\mathbf{x} \cdot \mathbf{1^T}) \\
 & \qquad +\mathbf{x}\cdot(1/\mathbf{B})^{\mathbf{T}} \\
\text{SUBJECT TO} & \mathbf{x} \cdot \mathbf{1^T} \geq \mathbf{r} \\
 & \mathbf{x^T} - (\mathbf{z} \cdot \mathbf{V})^{\mathbf{T}} \geq \mathbf{0} \\
 & \mathbf{z^T} - \mathbf{V} \cdot \mathbf{x^T} > -(\mathbf{b^T} - \mathbf{1}) \\
 & \mathbf{V} \cdot \mathbf{x^T} \leq \frac{\Delta}{100}(diag(\mathbf{b^T} \cdot \mathbf{z}) + \mathbf{b^T})
\end{array}
$$

Fig. 10. Adapt-V-PREPT's optimization formulation.

remaining constraints imply that when an LSP is preempted, all the capacity modules belonging to it should be preempted, that the respective modules for the LSPs that will reduce their rate are also preempted, and that the preempted rate is no more than $\Delta\%$ of the actual reserved bandwidth for that LSP. The total number of constraints is $2L' + M + 1$.

Fig. 10 contains a summary of Adapt-V-PREPT's integer program.

### *B. Adapt-V-PREPT's Heuristic*

As discussed before, the choice of LSPs to be preempted or to have their rate reduced is an NP-complete problem. In order to simplify and expedite the online choice of LSPs for preemption, we propose to use a simple heuristic shown to be as accurate as the optimization formulation illustrated in Fig. 10.

When using Adapt-V-PREPT, a new LSP setup request is treated differently. First, we test whether there is enough bandwidth among the preemptable LSPs in order to fit the new request $r$. If $\sum_{i=1}^{\mathcal{L}} b(i) \geq r$, we proceed. If that is not true, the LSP setup request is rejected. Suppose there is enough bandwidth. Now we test whether there are LSPs that can afford to reduce their rate. If not, we run V-PREPT (Fig. 4) and choose the LSPs that will be preempted and rerouted. If $\mathcal{L}' \neq \emptyset$, we test whether the bandwidth occupied by these LSPs is enough to fit $r$. If yes, we run Adapt-V-PREPT (Fig. 12), which will be explained in detail in the following, and choose the LSPs that will reduce their rate by a maximum of $\Delta\%$ or that will be completely preempted in order to accommodate $r$. If the bandwidth allocated to the LSPs that can reduce their rate is not enough to fit $r$, we execute V-PREPT to choose one LSP to be preempted and test again if the remaining required bandwidth, $r - \ preempt$ can be made free by reducing the rate of the LSPs in $\mathcal{L}'$ ( *preempt* accumulates the total preempted bandwidth amount). If the available bandwidth is still not enough, we again run V-PREPT, this time preempting two LSPs. Another test is made to see whether the remaining bandwidth can be accommodated by reducing the rate of $\mathcal{L}'$ elements and so on. Fig. 11 illustrates the new LSP setup procedure.

We propose the following equation, used in Adapt-V-PREPT's heuristic algorithm (Fig. 12):

$$\mathcal{H}(l) = \alpha y(l) + \beta + \gamma + \frac{1}{b(l)}. \tag{10}$$

In this equation, $\alpha y(l)$ represents the cost of preempting an LSP, $\beta$ represents the choice of a minimum number of LSPs for preemption or rate reduction, $\gamma$ represents the amount of bandwidth to be preempted, and $1/b(l)$ represents an additional cost by bandwidth module. This cost is calculated as the inverse of the amount of bandwidth reserved for the considered LSP. In this way, an LSP with more bandwidth modules will be more likely to be preempted than one with just a few number of modules.

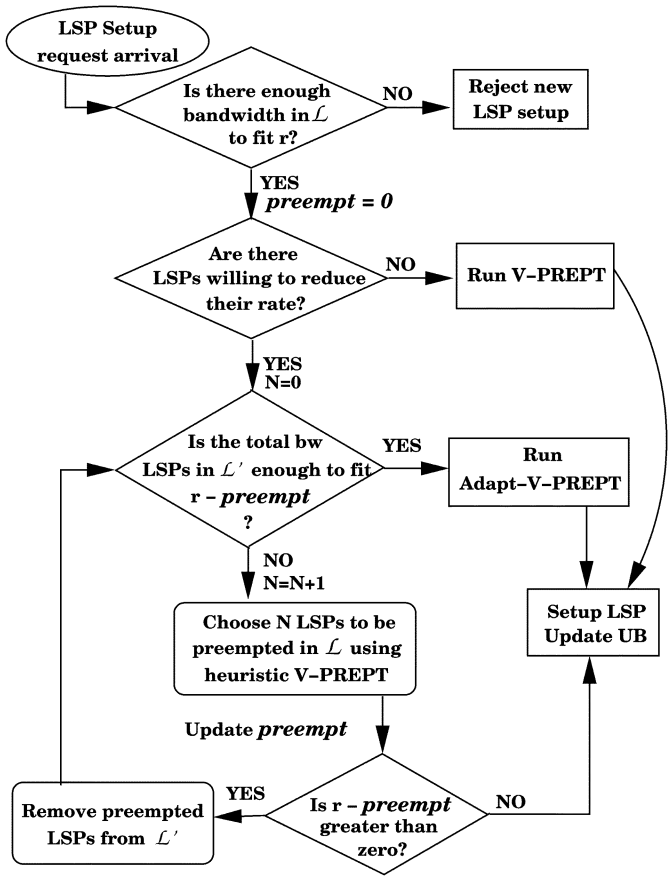


Fig. 11. Flowchart for LSP setup procedure with adaptive preemption policy.

**Alg. Adapt-V-PREPT** ($\mathcal{L}', \mathbf{b}, \mathbf{y}, r, \alpha, \beta, \gamma, \Delta$, *preempt*)
*rate_reduce = 0*;
generate vectors $\mathbf{x}$ and $\mathbf{x}$_index;
% $\mathbf{x}$_index contains pointers to where each LSP bandwidth
% allocation starts in vector $\mathbf{x}$.
*limit* = round_down($\Delta/100 * \mathbf{b}'$); % maximum rate reduction
calculate $\mathcal{H}(i) = \alpha y(i) + \beta + \gamma + 1/b(i)$;
sort $\mathcal{H}$, $\mathbf{b}$, in increasing order of $\mathcal{H}(i)$;
if $\mathcal{H}$ has repeated values sort those by increasing $\mathbf{b}$;
*r*_aux = *r-preempt*;
while *rate_reduce* + *preempt* < *r* {
  if $b(i)$ <= *r*_aux {
    *preempt* = *preempt* + $b(i)$;
    *r*_aux = *r*_aux - $b(i)$; $z(i) = 1$;
  else {
    index = $\mathbf{x}$_index(i);
    while bandwidth reduced for $i <$ *limit*$(i)$ and
    *rate_reduce* + *preempt* < *r* {
      x(index) = 1;
      *rate_reduce* = *rate_reduce* + 1;
      index = index + 1;
      *r*_aux = *r*_aux - *rate_reduce*;}}}
}
return($\mathbf{x}, \mathbf{z}$)

Fig. 12. Heuristic for Adapt-V-PREPT.

Our heuristic for Adapt-V-PREPT uses $\mathcal{H}$ to determine the choice of bandwidth modules that will be preempted (sometimes resulting in a whole LSP being preempted). $\mathcal{H}$ is calculated for each LSP in $\mathcal{L}'$ and it does not depend on the request value $r$. Again, $\mathcal{H}$ is sorted in increasing order with repeated values being ordered by increasing associated bandwidth $b$.

Fig. 12 illustrates the heuristic algorithm. The following input data is given: the set $\mathcal{L}'$; the request $r$; the parameters $\alpha$, $\beta$, $\gamma$; the vectors $\mathbf{b}$ and $\mathbf{y}$ containing bandwidth and cost information for each LSP in $\mathcal{L}'$, respectively; the maximum reduced rate per LSP in percentage $\Delta$; and the amount of bandwidth already preempted in a previously run V-PREPT algorithm, *preempt*. The algorithm calculates $\mathcal{H}$ and the preemptable bandwidth (number of modules) for each LSP, $limit(l) = \text{round\_down}(\Delta/100 * b(l))$: no more than $\Delta\%$ of $b(l)$.

Following the sorted $\mathbf{b}$ vector, and while the total preempted amount is not larger or equal to $r$, we test whether $b(i)$ is less than $r - preempt$. If that is true, the whole LSP $i$ will be preempted, $z(i)$ will be set to 1, and all the respective $x$ modules will be also made equal to 1. If not, we preempt module by module of LSP $i$ until we reach either the amount requested or the maximum value: $limit(i)$. The vector $\mathbf{x}$ is always updated with the respective preempted modules being set to 1. After that, if the requested amount is still not reached, we choose a new LSP from the sorted $\mathcal{H}$ and repeat the described process.

The output of the algorithm is $\mathbf{z}$ and $\mathbf{x}$, which contain information about which LSPs are to be preempted or have their rate reduced and by how much, respectively. Adapt-V-PREPT's heuristic is very accurate, as shown in the example discussed next.

### C. Example

Consider the same network proposed in Section III-D. Now suppose that LSPs $l_1$, $l_2$, $l_6$, $l_{10}$, and $l_{13}$ are not available for rate reducing, which means that $\mathcal{L}' = [l_3, l_4, l_5, l_7, l_8, l_9, l_{11}, l_{12}, l_{14}, l_{15}, l_{16}]$. The vectors $\mathbf{b}$, $\mathbf{p}$, and $\mathbf{y}$ are now composed of the bandwidth, holding preemption priority, and cost assignments for these LSPs only.

We run several simulations varying the value of $r$ in order to compare the cost of rate reduction and preemption, and with $\alpha = \beta = \gamma = 1$, which means that the network operator considered all the criteria with the same importance. Moreover, the parameter $\Delta$ was configured as $\Delta = 50$, indicating that each LSP in $\mathcal{L}'$ is willing to have its rate reduced by a maximum 50% of the bandwidth already reserved for it. Fig. 13 shows a chart that illustrates the results obtained for several different requests $r$. The $x$ labels indicate that the rate of that LSP was reduced, while the $z$ labels indicate that the whole LSP was preempted to satisfy the request $r$. Note that rate reduction never overcomes the 50% total bandwidth limit on each LSP.

Fig. 14 shows the accuracy of our heuristic for Adapt-V-PREPT. The heuristic obtains the same results found by the optimization formulation. Several other cases were run and the results found by the optimization and heuristic always matched.

To illustrate the case in which $r$ is larger than total bandwidth reserved by LSPs in $\mathcal{L}'$, suppose a request for $r = 600\,\text{Mb/s}$ arrives. We observe that $\sum_{i=1}^{11} b(i) = 565$, which is less than $r$. In this case, following the flowchart in Fig. 11, we run V-PREPT

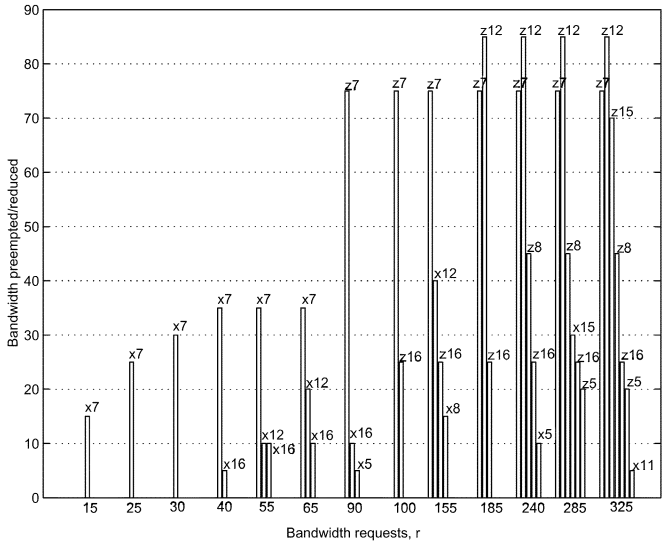


Fig. 13. Rate reduction and preemption for several values of $r$.

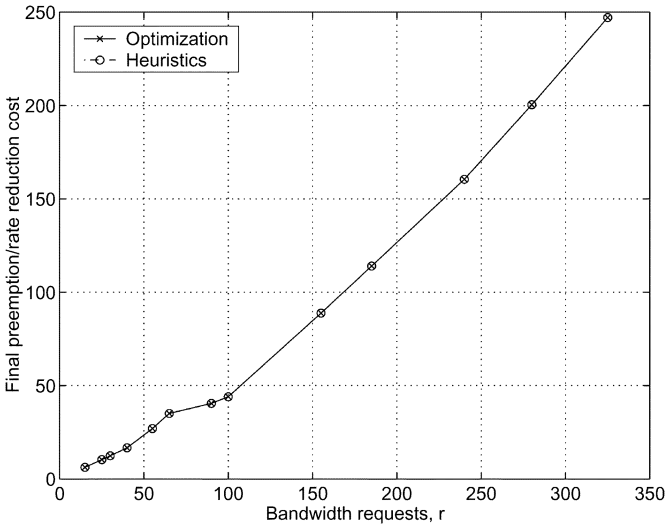


Fig. 14. Optimization and heuristic for Adapt-V-PREPT.

and keep selecting LSPs to be preempted until $r - preempt$ is less than the bandwidth of the remaining LSPs in $\mathcal{L}'$. This means that LSPs $l_3$, $l_7$, $l_9$, $l_{12}$, $l_{13}$, and $l_{15}$ are preempted, resulting in $preempt = 440\,\text{Mb/s}$, using V-PREPT's heuristic (Fig. 4). The remaining bandwidth, $r - preempt = 160$, is now suitable for Adapt-V-PREPT (Fig. 12). Running this heuristic, we realize that LSPs $l_4$, $l_5$, $l_8$, $l_{11}$, and $l_{16}$ are preempted completely, making $preempt = 595$, and LSP $l_{14}$ reduces its rate by 5 Mb/s, which results in a total of exactly 600 Mb/s available bandwidth for the new LSP setup request.

## V. Performance Evaluation

In order to highlight the benefits of a preemption enabled scenario, we grouped the priority levels into three categories: low ($p = 5, 6, 7$); medium ($p = 2, 3, 4$); and high ($p = 0, 1$). We perform a simulation in which LSP setup requests were generated randomly in time (average 1.5 per second), with random bandwidth request $b$ (varying from 1 to 10 Mb/s), random priority level (0–7), and exponentially distributed

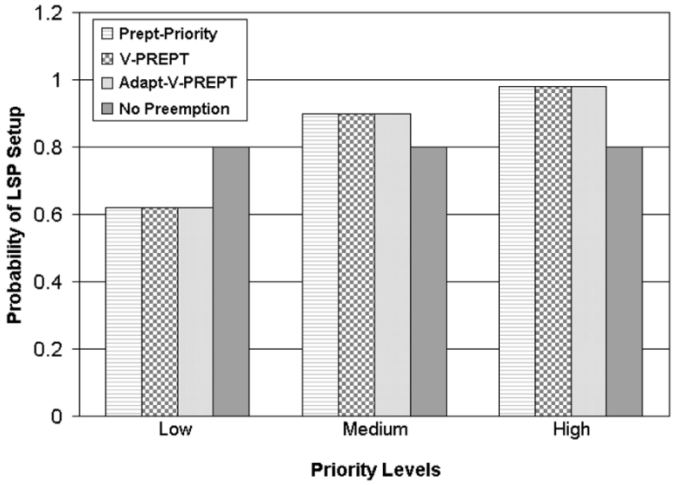


Fig. 15. LSP setup probability for preemptive and nonpreemptive scenarios.

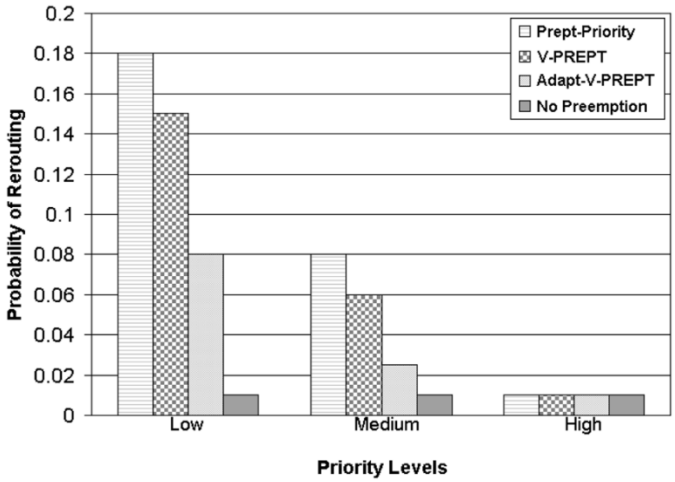


Fig. 16. LSP rerouting probability for preemptive and nonpreemptive scenarios.

holding time with average 500 seconds. The total capacity per link was 155 Mb/s. The Abilene network topology was considered. We observed the probability of a successful LSP setup for low, medium, and high level priorities for preemption-enabled and nonpreemptive approaches as well as the probability of rerouting. Prept-Priority is a preemption policy in which LSPs are selected first based only on their priority (LSPs with lower priority are selected), and then on their "age" (holding time). It selects the LSPs with lower priority and then the one that has been established for the longest time ("older"). This policy is similar to the one currently in use in CISCO routers.

Fig. 15 illustrates the results obtained for each priority category. For the nonpreemptive approach, the probability of LSP setup does not change with priority level. However, in a preemptive approach, low priority LSPs will be rerouted, while high priority LSPs will be more stable, always using the best path and only being rerouted in case of link failure. LSP setup probability is the same for V-PREPT and Adapt-V-PREPT heuristics due to the fact that in the worst case Adapt-V-PREPT will also preempt the whole LSP.

For a nonpreemptive approach rerouting will only happen when link failure occurs, and we denote a probability of 0.01 for that event (a common value used for failure estimation in current networks). Fig. 16 shows the rerouting probability. For lower priority and medium priority LSPs, the rerouting probability is higher. For high priority traffic, the probability is almost the same as for nonpreemptive approach, since this kind of traffic will not be preempted by other traffics. The rerouting probability for Adapt-V-PREPT's heuristic is smaller for medium and low priority traffic, and would depend on the request $r$, since the LSPs would have their rate reduced to fit the new request, which implies less preemption and consequently less rerouting.

In Fig. 17, Prept-Priority and our preemption policies, V-PREPT and Adapt-V-PREPT, are compared by calculating a cost for each solution and for each request $r$, using the same cost definition: $\text{Cost} = \sum \mathbf{b} \cdot y + \sum \mathbf{z} + b + \sum \mathbf{z}(b - r)$, where $b$ is the total bandwidth preempted by the new LSP, and $\mathbf{b}$, $\mathbf{y}$, and $\mathbf{z}$ are the same vectors defined earlier in this paper. This cost function gives more importance to the priority of the preempted LSPs ($\sum \mathbf{b} \cdot y$), but it also includes the number of

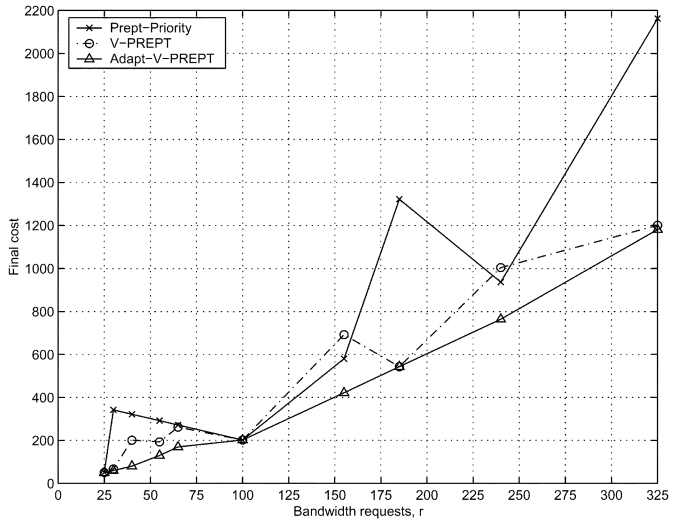


Fig. 17. Cost for Prept-priority, V-PREPT, and Adapt-V-PREPT.

preempted LSPs ($\sum \mathbf{z}$), the preempted bandwidth ($b$), and a penalization regarding bandwidth wastage and number of LSPs preempted ($\sum \mathbf{z}(b - r)$). The results for the three policies coincide when the chosen LSPs to be preempted are exactly the same. The final cost achieved by the preemption policy complemented by the adaptive rate scheme (Adapt-V-PREPT) is significantly smaller than the one obtained by the preemption policy (V-PREPT) by itself and than the one obtained by Prept-Priority. Moreover, signaling costs are reduced due to the fact that rerouting is performed less frequently.

The nonmonotonicity of Prept-Priority's curve in Fig. 17 is due to the fact that sometimes the selected lowest priority LSP has a bandwidth much higher than the requested bandwidth $r$, resulting in a high cost. In other situations, the selected LSP had just enough bandwidth, resulting in less bandwidth wastage and therefore smaller cost. It is important to note that even though Prept-Priority results in a slightly lower cost for two values of $r$ in Fig. 17, that is due to the fact that our chosen cost function gives more importance to the priority of the preempted LSPs: $\sum \mathbf{b} \cdot y$ is a heavy component in the cost function. The choice of an LSP with bandwidth much larger than the request $r$ but with

TABLE III
NUMBER OF PREEMPTED LSPS AND BANDWIDTH WASTAGE FOR EACH POLICY

| r | Number of LSPs | | | Bandwidth Wastage | | |
|---|---|---|---|---|---|---|
| | Priority | V-PREPT | Adaptive | Priority | V-PREPT | Adaptive |
| 15 | 1 | 1 | 0 | 10 | 5 | 0 |
| 25 | 1 | 1 | 0 | 0 | 0 | 0 |
| 30 | 2 | 2 | 0 | 70 | 0 | 0 |
| 40 | 2 | 1 | 0 | 60 | 0 | 0 |
| 55 | 2 | 3 | 0 | 45 | 0 | 0 |
| 65 | 2 | 2 | 0 | 35 | 0 | 0 |
| 90 | 2 | 2 | 1 | 10 | 0 | 0 |
| 100 | 2 | 2 | 2 | 0 | 0 | 0 |
| 155 | 6 | 2 | 2 | 16 | 0 | 0 |
| 185 | 7 | 3 | 3 | 71 | 0 | 0 |
| 240 | 7 | 4 | 4 | 16 | 0 | 0 |
| 280 | 8 | 5 | 5 | 1 | 0 | 0 |
| 325 | 10 | 6 | 6 | 66 | 0 | 0 |

TABLE IV
PRIORITY OF PREEMPTED/REDUCED LSPS FOR EACH POLICY

| r | Priority of LSPs | | |
|---|---|---|---|
| | Prept-Priority | V-PREPT | Adapt-V-PREPT |
| 15 | 7 | 5 | 7 |
| 25 | 7 | 7 | 7 |
| 30 | 7,7 | 7,6 | 7 |
| 40 | 7,7 | 4 | 7,7 |
| 55 | 7,7 | 7,6,4 | 7,7,5 |
| 65 | 7,7 | 7,4 | 7,7,5 |
| 90 | 7,7 | 6,5 | 7,7,5 |
| 100 | 7,7 | 7,7 | 7,7 |
| 155 | 7,7,6,6,5,5 | 5,4 | 7,7,5,5 |
| 185 | 7,7,6,6,5,5,5 | 7,7,5 | 7,7,5 |
| 240 | 7,7,6,6,5,5,5 | 7,7,4,3 | 7,7,5,5,5 |
| 280 | 7,7,6,6,5,5,5,4 | 7,6,5,5,4 | 7,7,5,5,5,4 |
| 325 | 7,7,6,6,5,5,5,4,4,4 | 7,7,5,5,4,4 | 7,7,5,5,5,4,4 |

a very low priority will result in a lower cost than the selection of an LSP with the exact amount of bandwidth requested but not so low preemption priority. The two tables that follow give a better general overview of the advantages of using V-PREPT and Adapt-V-PREPT.

Table III shows the number of preempted LSPs and the bandwidth wastage, for each policy (Prept-Priority values are under the label *Priority* and Adapt-V-PREPT values are under the label *Adaptive*). Prept-Priority leads to the highest bandwidth wastage and, in many cases, the highest number of preempted LSPs. The adaptive policy always preempts the exact requested bandwidth, and some times it does not preempt any LSP at all.

The priority of the LSPs preempted or selected for rate reduction are shown in Table IV. Note that for Adapt-V-Prept we also show the priority of LSPs that were not preempted, but only had their rate reduced. Another important information to keep in mind is that LSPs $l_1$, $l_2$, $l_6$, $l_{10}$, and $l_{13}$ were not available for rate reduction. Some of these LSPs have lower priority than others selected by the policy, but could not be chosen for rate-reduction due to the nature of the traffic carried by them.

The results shown in Tables III and IV as well as in Fig. 17 are related to the simulations performed with $\alpha = \beta = \gamma = 1$ for each policy.

The running time of the heuristic is $O(L \times h)$, where $L$ is the number of LSPs in a single link and $h$ is the number of hops in the path where the new LSP will be setup. We ran the heuristic on a link with 2000 LSPs and the decision on which LSP to preempt in that single link was taken in less than 30 ms (using a Pentium III PC, 1 GHz, 128 MB). With 200 LSPs, the running time was 2.8 ms.

## VI. CONCLUSION

In this paper, new preemption policies that aim to minimize rerouting caused by preemption were proposed. V-PREPT is a versatile preemption policy that combines the three main optimization criteria: number of LSPs to be preempted, priority of LSPs to be preempted, and amount of bandwidth to be preempted. Adapt-V-PREPT is an adaptive preemption policy that selects low-priority LSPs that can afford to reduce their rate by a maximum percentage in order to free bandwidth to accommodate a new high-priority LSP. Heuristics for both V-PREPT and Adapt-V-PREPT were derived and their accuracy is demonstrated by simulation and experimental results. Performance comparisons of a nonpreemptive approach, our preemption policies, V-PREPT and Adapt-V-PREPT, and a policy similar to the currently in use by commercial routers show the advantages of using our policies in a differentiated services environment. The proposed adaptive rate policy performs much better than the standalone preemption policy. Further studies regarding cascading effect (preemption caused by preempted LSPs) have been investigated by the authors and are reported in [19].

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their thoughtful and contructive comments. They also thank L. Chen for his help in prototyping V-PREPT and Adapt-V-PREPT for testbed experiments. The authors would also like to acknowledge the helpful and insightful conversations with J.-P. Vasseur and F. Le Faucheur, Cisco Systems.

## REFERENCES

[1] D. O. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for traffic engineering over MPLS," IETF, RFC 2702, Sept. 1999.

[2] F. L. Faucheur and W. Lai, "Requirements for support of differentiated services-aware MPLS traffic engineering," IETF, RFC 3564, July 2003.

[3] A. Bobbio, A. Puliafito, and M. Tekel, "A modeling framework to implement preemption policies in non-Markovian SPNs," *IEEE Trans. Software Eng.*, vol. 26, pp. 36–54, Jan. 2000.

[4] C.-G. Lee, K. Lee, J. Hahn, Y.-M. Seo, S. L. Min, R. Ha, S. Hong, C. Y. Park, M. Lee, and C. S. Kim, "Bounding cache-related preemption delay for real-time systems," *IEEE Trans. Software Eng.*, vol. 27, pp. 805–826, Sept. 2001.

[5] J. Wang, Q.-A. Zeng, and D. P. Agrawal, "Performance analysis of preemptive handoff scheme for integrated wireless mobile networks," in *Proc. IEEE GLOBECOM*, San Antonio, TX, 2001, pp. 3277–3281.

[6] S. Poretsky, "Connection precedence and preemption in military asynchronous transfer mode (ATM) networks," in *Proc. MILCOM'98*, 1998, pp. 86–90.

[7] S. Poretsky and T. Gannon, "An algorithm for connection precedence and preemption in asynchronous transfer mode (ATM) networks," in *Proc. IEEE ICC'98*, 1998, pp. 299–303.

[8] D. O. Awduche and B. Jabbari, "Internet traffic engineering using multiprotocol label switching (MPLS)," *Comput. Netw.*, vol. 40, pp. 111–129, 2002.

[9] G. Armitage, *Quality of Service in IP Networks*. New York: MacMillan, 2000.

[10] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and principles of Internet traffic engineering," IETF, RFC 3272, May 2002.

[11] F. Le Faucheur, "Russian dolls bandwidth constraints model for Diff-Serv-aware MPLS traffic engineering," IETF, Internet Draft, work in progress, June 2003.
[12] W. S. Lai, "Bandwidth constraint models for DiffServ-aware MPLS traffic engineering: Performance evaluation," IETF, Internet Draft, work in progress, June 2003.
[13] J. Ash, "Max allocation with reservation bandwidth constraint model for MPLS/DiffServ TE and performance comparisons," IETF, Internet Draft, work in progress, Mar. 2003.
[14] F. Le Faucheur, "Considerations on bandwidth constraint models for DS-TE," IETF, Internet Draft, work in progress, June 2002.
[15] ——, "Protocol extensions for support of DiffServ-aware MPLS traffic engineering," IETF, Internet Draft, work in progress, June 2003.
[16] M. Peyravian and A. D. Kshemkalyani, "Decentralized network connection preemption algorithms," *Comput. Netw. ISDN Syst.*, vol. 30, no. 11, pp. 1029–1043, June 1998.
[17] P. Dharwadkar, H. J. Siegel, and E. K. P. Chong, "A heuristic for dynamic bandwidth allocation with preemption and degradation for prioritized requests," in *Proc. 21st Int. Conf. Distributed Computing Systems*, Phoenix, AZ, 2001, pp. 547–556.
[18] J. A. Garay and I. S. Gopal, "Call preemption in communication networks," *Proc. IEEE INFOCOM*, pp. 1043–1050, 1992.
[19] J. C. de Oliveira, L. Chen, C. Scoglio, and I. F. Akyildiz, "LSP preemption policies for DiffServ-aware MPLS traffic engineering," IETF, Internet Draft, work in progress, Mar. 2003.

**Jaudelice C. de Oliveira** (S'98–M'03) was born in Fortaleza, Ceara, Brazil. She received the B.S.E.E. degree from Universidade Federal do Ceara (UFC), Ceara, Brazil, in December 1995, the M.S.E.E. degree from Universidade Estadual de Campinas (UNICAMP), Sao Paulo, Brazil, in February 1998, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, in May 2003.

She joined Drexel University, Philadelphia, PA, in July 2003, as an Assistant Professor. Her research interests include the development of new protocols and policies to support fine grained quality of service provisioning in the future Internet.

Dr. de Oliveira has been a member of the ACM since 1998.

**Caterina Scoglio** (M'90–A'03) was born in Catania, Italy. She received the Dr. Ing. degree (*summa cum laude*) in electronics engineering from the University of Rome "La Sapienza", Italy, in May 1987. She received a post-graduate degree in "mathematical theory and methods for system analysis and control" from the University of Rome "La Sapienza", in November 1988.

From June 1987 to June 2000, she was with Fondazione Ugo Bordoni, Rome, Italy, where she was a Research Scientist at the TLC Network Department—Network Planning Group. From November 1991, to August 1992, she was a Visiting Researcher at the College of Computing, Georgia Institute of Technology, Atlanta. Since September 2000, she has been with the Broadband and Wireless Networking Laboratory, Georgia Institute of Technology, as a Research Engineer. She is leading the IP QoS group in their research on QoS issues in next-generation Internet and DiffServ/MPLS networks. Her research interests include optimal design and management of multiservice networks.

**Ian F. Akyildiz** (M'86–SM'89–F'96) received the B.S., M.S., and Ph.D. degrees in computer engineering from the University of Erlangen-Nuernberg, Germany, in 1978, 1981, and 1984, respectively.

Currently, he is the Ken Byers Distinguished Chair Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, and Director of the Broadband and Wireless Networking Laboratory. He has held visiting professorships at the Universidad Tecnica Federico Santa Maria, Chile, Universite Pierre et Marie Curie (Paris VI), Ecole Nationale Superieure Telecommunications in Paris, France, Universidad Politecnico de Cataluna in Barcelona, Spain, and Universidad Illes Baleares, Palma de Mallorca, Spain. He has published over 200 technical papers in journals and conference proceedings. His current research interests are in Sensor Networks, InterPlaNetary Internet, Wireless Networks, Satellite Networks and the Next Generation Internet.

He is an Editor-in-Chief of *Computer Networks* (Elsevier) and of the newly launched *AdHoc Networks Journal*, and an Editor for ACM/Kluwer *Journal of Wireless Networks*. He is a past editor for IEEE/ACM TRANSACTIONS ON NETWORKING (1996–2001), Kluwer *Journal of Cluster Computing* (1997–2001), ACM/Springer *Journal for Multimedia Systems* (1995–2001), and IEEE TRANSACTIONS ON COMPUTERS (1992–1996). He has guest edited more than ten special issues for various journals in the last decade. He was the technical program chair of the 9th IEEE Computer Communications Workshop in 1994, ACM/IEEE MOBICOM'96 (Mobile Computing and Networking) Conference, IEEE INFOCOM'98 (Computer Networking Conference), and IEEE ICC'2003 (International Conference on Communications). He served as the General Chair for the premier conference in wireless networking, ACM/IEEE MOBICOM'2002, Atlanta, September 2002. He is the co-founder and co-General Chair of the newly established ACM SenSys'03 Conference on Sensor Systems, which will take place in Los Angeles, CA, in November 2003.

Dr. Akyildiz has been a Fellow of the ACM since 1996. He received the Don Federico Santa Maria Medal for his services to the Universidad of Federico Santa Maria, Chile, in 1986. He served as a National Lecturer for ACM from 1989 to 1998 and received the ACM Outstanding Distinguished Lecturer Award for 1994. He received the 1997 IEEE Leonard G. Abraham Prize Award (IEEE Communications Society) for his paper entitled "Multimedia Group Synchronization Protocols for Integrated Services Architectures" published in the IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS (JSAC) in January 1996. He received the 2002 IEEE Harry M. Goode Memorial award (IEEE Computer Society) with the citation "for significant and pioneering contributions to advanced architectures and protocols for wireless and satellite networking." He received the 2003 IEEE Best Tutorial Award (IEEE Communicaton Society) for his paper entitled "A Survey on Sensor Networks" published in *IEEE Communication Magazine* in August 2002. He also received the 2003 ACM Sigmobile Outstanding Contribution Award with the citation "for pioneering contributions in the area of mobility and resource management for wireless communication networks."

**George Uhl** is the Lead Engineer at NASA's Earth Science Data and Information System (ESDIS) Network Prototyping Laboratory, NASA Goddard Space Flight Center, Beltsville, MD. He directs network research and protoyping activities for ESDIS. His current areas of research include network quality of service and end-to-end performance improvement.