Using Topology Aggregation for Efficient Shared Segment Protection Solutions in Multi-Domain Networks

Dieu-Linh Truong and Brigitte Jaumard, *Member, IEEE*

***Abstract*— The dynamic routing problem for *Overlapping Segment Shared Protection* (OSSP) in multi-domain networks has not received a lot of interest so far as it is more complex than in single-domain networks. Difficulties lie in the lack of complete and global knowledge about network topologies and bandwidth allocation whereas this knowledge is easily available in single-domain networks. We propose a two-step routing approach for the OSSP based on a topology aggregation scheme and link cost estimation: an inter-domain step and an intra-domain step. We propose two different heuristics, GROS and DYPOS for the inter-domain step, and a "Blocking-go-back" strategy in order to reduce the blocking rate in the intra-domain step. We compare the performance of the two heuristics against an optimal single-domain approach. We show that both heuristics lead to resource efficient solutions that are not far from the optimal ones. Moreover, both heuristics require relatively small computational efforts and are scalable for multi-domain networks.**

***Index Terms*— Multi-domain network, protection, routing.**

I. Introduction

In segment protection, an end-to-end working path is divided into segments, each of which is protected by a unique backup segment. Only one backup segment is activated upon a single link or node failure, the other working segments, which are not impaired by the failure, remain used. As a result, segment protection offers a faster recovery than path protection. In the classical segment protection schemes, working segments are non-overlapping. Segment end nodes are then not protected because the failures of those nodes impair both working and backup segments. Overlapping Segment Protection, firstly proposed in [23] and [10], overcomes this weakness thanks to the overlapping between working segments (see Fig. 1) while still inheriting the fast recovery property of segment protection.

For achieving backup bandwidth efficiency, shared protection has been proposed for link, path and segment protections [22]. In segment protection, in order to guarantee 100% recovery under a single link or node failure, two backup segments can share some bandwidth if and only if their working segments are link and node-disjoint. We call this *segment sharing condition*. Fig. 2 gives an illustration. In case (a), the working segment from v_1 to v_2 with requested bandwidth d_1 and the working segment from v_5 to v_6 with requested bandwidth d_2 are link and node disjoint. Therefore their backup segment can share bandwidth over the common link (v_4, v_3) and the total bandwidth used by the two backup segments on this link is $\max\{d_1, d_2\}$. In case (b), the two working segments share node v_7, therefore their backup segments must reserve separate backup bandwidth. The total backup bandwidth for both backup segments on link (v_4, v_3) is $d_1 + d_2$ which is greater than in case (a).

With the shared protection feature, Overlapping Segment

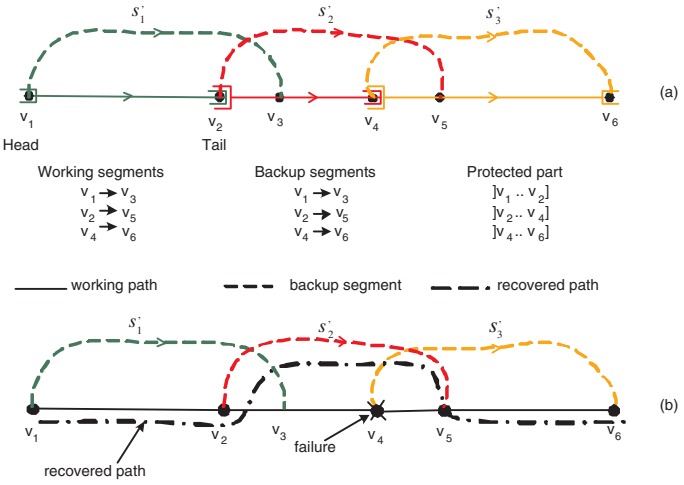

Fig. 1. Example of Overlapping Segment Protection when v_4 fails. The protected part $]v_2..v_4]$ contains all links and nodes between v_2 exclusively and v_4 inclusively, thus v_4 is recovered by segment s'_2.

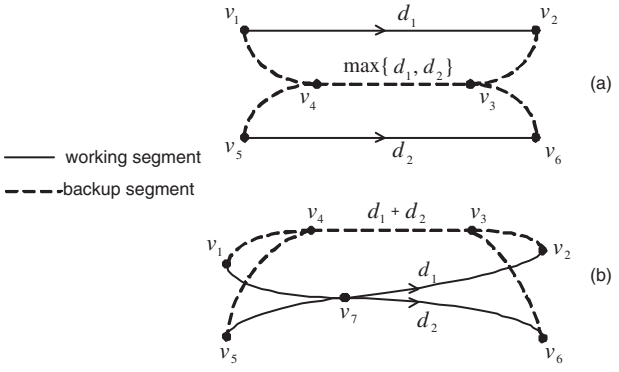

Fig. 2. Examples of backup bandwidth sharable (a) and non-sharable (b) cases.

Manuscript received November 3, 2006; revised August 20, 2007.

D. L. Truong is with the Department of Computer Science and Operations Research, Université de Montréal, Montréal, QC, Canada (e-mail: linh@crt.umontreal.ca).

B. Jaumard is with the Concordia Institute for Information Systems Engineering, Concordia University, Montréal, QC, Canada (e-mail: bjaumard@ciise.concordia.ca).

Digital Object Identifier 10.1109/JSAC-OCN.2007.028606.

Protection becomes Overlapping Segment Shared Protection (OSSP). This paper aims to investigate the OSSP routing problem in multi-domain networks because of its characteristics: node protection, fast recovery and bandwidth saving.

Shared protection under static traffic has received a lot of interest. Many efficient solutions have been proposed, especially the well-known p-cycle. It was initially introduced in [9] and further developed for segment protection in [25], [26]. However, network traffic changes unpredictably and dynamically today, static traffic is no longer an appropriate assumption unless in the network design or planning contexts. For this reason, we are focusing only on dynamic traffic where a new incoming request needs to be routed without assuming any forecast about the upcoming requests. The objective of the routing is to minimize the bandwidth capacity used by both working and backup segments of the incoming request.

A multi-domain network is an interconnection of several single-domain networks [7] (Fig. 3a). For the *scalability requirement*, only the aggregated routing information can be exchanged among domains [24] by an Exterior Gateway Protocol (EGP) such as Border Gateway Protocol (BGP). Consequently, a given node is neither aware of the global multi-domain network topology nor of the detailed bandwidth allocation on each network link. However, the complete routing information is still available within each domain thanks to more frequent routing information exchanges performed by an Interior Gateway Protocol (IGP) such as the link state routing protocols Open Shortest Path First (OSPF), Intermediate System to Intermediate System (IS-IS) etc..

Most studies on OSSP remain within the single-domain network context. An optimal solution has been proposed in [12] but it requires a huge computational effort even for small networks. Several heuristics with smaller computational efforts have been proposed such as the work in [23], SLSP-O in [11], CDR in [10], PROMISE in [31] or recursive shared segment protection [8]. The first study ignores the sharing possibility during the routing. The other ones as well as the optimal solution scheme in [12] are restricted to single domain networks as they assume that the global and detailed network information is available at any given internal node.

Some solutions have also been proposed for multi-domain networks with drawbacks. In [21], the working path is divided into non-overlapping segments at domain border nodes which are then not protected. In [6], the authors decide to recover those border nodes in case of failure by using an end-to-end restoration. However, in comparison with protection, restoration offers a slower and uncertain recovery leading to recovery quality degradation. In [18], a simple multi-domain network without transit domain is assumed: domains do not directly connect to each other but to a central backbone domain. Connections from one domain to another one are established through some links of the backbone domain. In practice, neighboring domains connect to each other without a backbone domain and a connection between distant domains goes often through one or more transit domains. This makes the routing problem more complex.

In this study, we focus on general multi-domain networks with transit domains. We develop a two-step heuristic solution. The multi-domain network is first topologically aggregated

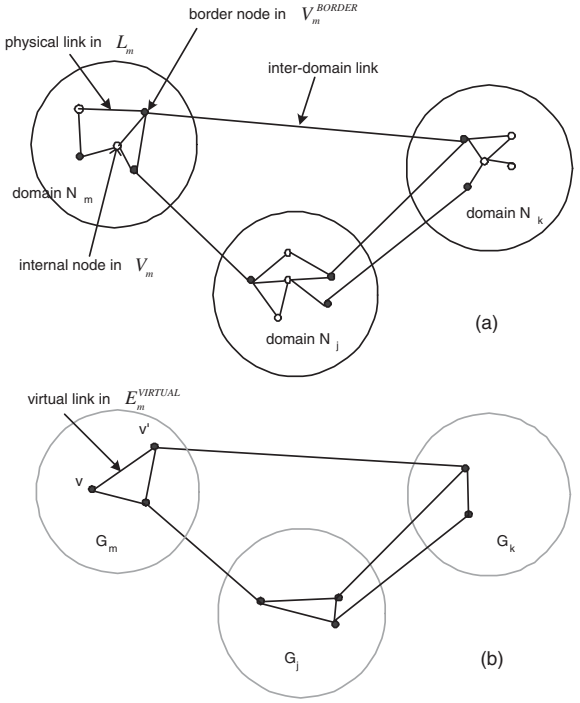

Fig. 3. A multi-domain network (a) and its *inter-domain network* (b) obtained from Topology Aggregation.

to become a compact network called *inter-domain network*, where a rough routing is sketched out. Then detailed routings are performed inside each original domain network. The use of an aggregated topology at the first step eliminates the need for global and detailed information requirements and thus preserves the scalability. The first routing step can be solved by using a greedy or dynamic programming algorithm (to be presented in sections IV-B and IV-C) or any single-domain routing solution.

In the proposed solution, the working and backup segment lengths are also restricted. It is known that the failure recovery time consists mainly of the failure notification time and the backup segment activation time. The first one is proportional to the working segment length and the second one is proportional to the backup segment length [8]. Therefore, the restrictions on the working and backup segment lengths will guarantee a fast recovery. Except for SLSP-O, in most published OSSP solutions, this restriction is not considered, leading usually to solutions with single segment patterns. The segment protection solutions degenerate thus to path protection solutions. The bandwidth cost may decrease but the recovery time increases. In some experiments of PROMISE in [32], the length of a backup segment is bounded by a function of its working segment length. However, this bound is not tight as the working segment length is not bounded. On the contrary, we restrict both working and backup segment lengths in our study. The single-domain solution in [8] and SHALL in [16] restrict the working and backup segment lengths. However, SHALL uses Suurballe and Tarjan′s algorithm [27], which does not require overlapping between segments, and thus cannot offer the node protection capability. Similarly, the work in [8] does not require backup segment overlapping either.

We consider networks with bandwidth guaranteed connec-

tions such as SONET/SDH, MPLS-TE, ATM or DWDM networks. In the case of DWDM networks, each network node is assumed to be equipped with Multiservice Provisioning Platform (MSPP, see i.e., [19]) with bandwidth grooming and wavelength conversion abilities. The wavelength continuity constraint and wavelength assignment problem are thus relaxed. Without bandwidth grooming, the proposed solution is still applicable on DWDM network as long as one wavelength is considered as a bandwidth unit.

This paper is organized as follows. Notations and fundamental concepts are introduced in the next section. Section III presents link costs which will be used in the routing algorithms proposed in Section IV. Section V outlines the signaling processes that coordinate the routing, the connection setup as well as the information update. Computational results are discussed in Section VI. Section VII concludes the paper.

II. Fundamental concepts and Notations

The multi-domain network is represented by a graph $\mathcal{N} = (V, L)$ composed of M connected single-domain networks $\mathcal{N}_m = (V_m, L_m)$, $m = 1, .., M$ where V, V_m are sets of nodes and L, L_m are sets of links. Each single-domain network contains border nodes which connect with the border nodes of other domains through inter-domain links (see Fig. 3a). The set of border nodes of $\mathcal{N}_m$ is V_m^{BORDER}. The set of inter-domain links of the multi-domain network is $L^{\text{INTER}} \subset L$. Thus:

$$V = \bigcup_{m=1..M} V_m,$$

$$L = \left(\bigcup_{m=1..M} L_m \right) \cup L^{\text{INTER}}.$$

A full mesh topology aggregation (TA) will be applied to each domain network. The TA on domain $\mathcal{N}_m$ results in an aggregated graph $G_m = (V_m^{\text{BORDER}}, E_m^{\text{VIRTUAL}})$ containing only border nodes of $\mathcal{N}_m$ and a set of virtual links connecting all pairs of border nodes $E_m^{\text{VIRTUAL}} = \{(v_1, v_2) : v_1, v_2 \in V_m^{\text{BORDER}}\}$. A virtual link $(v_1, v_2) \in G_m$ represents the set of intra domain paths (called intra-paths) inside $\mathcal{N}_m$ from v_1 to v_2. The multi-domain network is transformed into the compact network $G = (V^{\text{BORDER}}, E)$, called *inter-domain network* (see illustration on Fig. 3b), where

$$V^{\text{BORDER}} = \bigcup_{m=1..M} V_m^{\text{BORDER}},$$

$$E = \left(\bigcup_{m=1..M} E_m^{\text{VIRTUAL}} \right) \cup L^{\text{INTER}}.$$

We will denote by e an edge of G, e can then be a virtual link or an inter-domain link. Let $\mathcal{P}_e$ be the set of intra-paths represented by e if e is a virtual link and $\mathcal{P}_e = \{e\}$ if e is an inter-domain link. Edge e will be associated with some link-states containing aggregated routing information obtained from its intra-paths. Such aggregated information can be exchanged between border nodes without impairing the *scalability requirement*. The *inter-domain network* can be then viewed as a single-domain network.

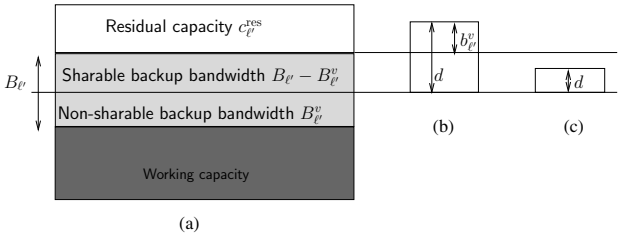

Fig. 4. (a) bandwidth structure on a physical link ℓ'; (b) and (c) are two examples of the required additional backup bandwidth on link ℓ'.

Let us consider a new incoming request with bandwidth d from node v_s to node v_d to be routed without flow bifurcation. We have to find an end-to-end working path p made of $|I|$ segments $\{p_i, i \in I\}$, and a set of backup segments $\{p'_i, i \in I\}$ such that their total bandwidth capacity is minimized. The backup segment p'_i protects the working segment p_i. The working path consumes bandwidth d on each of its links without any sharing. The following additional notations will be used in the routing algorithms.

A. Notations used for the original multi-domain network

- c_ℓ^{res} total residual bandwidth capacity on physical link $\ell \in L$.
- a_ℓ bandwidth to be used by the working path p of the new incoming request on physical link $\ell \in L$ if p contains ℓ. In this notation as well as in the other ones related to the new incoming request, the index p (or sometimes p'_i) is omitted to relieve the notations.
- $B_{\ell'}$ total backup bandwidth already reserved by backup segments on physical link $\ell' \in L$ before the routing of the new incoming request.
- $B_{\ell'}^v$ fraction of backup bandwidth on physical link $\ell' \in L$ that is reserved by the backup segments whose working segments go through node $v \in V$. Of course, $B_{\ell'}^v \leq B_{\ell'}$. This backup bandwidth cannot be shared with the backup segments of the new incoming request that protect v otherwise it would violate the *segment sharing condition*.
- $B_{\max}^v = \max_{\ell' \in E} B_{\ell'}^v$ and $B_{\max}^q = \max_{v \in q} B_{\max}^v$ the maximum backup bandwidths reserved on a network link in order to protect the working segments going through node v and through sub-path q respectively.
- $b_{\ell'}^v$ additional backup bandwidth (with respect to the existing backup bandwidth $B_{\ell'}$) that a backup segment of the new incoming request needs to reserve on a physical link ℓ' in order to ensure the survival of all already protected working segments as well as of its own working segment when node v fails.
- $b_{\ell'}^q$ additional backup bandwidth (with respect to the existing backup bandwidth $B_{\ell'}$) that a backup segment of the new incoming request needs to reserve on a physical link ℓ' in order to ensure the survival of all already protected working segments as well as its own working segment when a node or a link of the sub-path q fails.

Fig. 4a illustrates the bandwidth structure on physical link ℓ'. Clearly, the sharable backup bandwidth on ℓ' for protecting

a node v is $(B_{\ell'} - B^v_{\ell'})$. This bandwidth is profitable for the new incoming request in order to protect v. There are two cases: i) the sharable backup bandwidth is not greater than the requested bandwidth d, we need an additional backup bandwidth $b^v_{\ell'} = d - (B_{\ell'} - B^v_{\ell'})$ on ℓ' (Fig. 4b); ii) the sharable backup bandwidth is greater than d, no additional backup bandwidth is necessary and $b^v_{\ell'} = 0$ (Fig. 4c). In other words:

$$b^v_{\ell'} = \max\{0, B^v_{\ell'} + d - B_{\ell'}\}. \tag{1}$$

Readers are encouraged to read [14] and [30] for detailed and similar computations in the case of link protection.

Observe that with OSSP, for a given node, the same backup segments must be activated when either this node fails or all its adjacent links fail simultaneously. The solution that protects a node is sufficient to protect every adjacent link of the node. We deduce the following result:

Property 1: The backup bandwidth required on a link (ℓ') by one backup segment of a connection in order to protect a working segment (q) is equal to the largest backup bandwidth needed on this link in order to protect a node of the working segment:

$$b^q_{\ell'} = \max_{v \in q} b^v_{\ell'}. \tag{2}$$

B. Notations used for the inter-domain network

- $\pi, \pi_i, \pi'_i (i \in I)$ representations of the working path p, working and backup segments p_i, p'_i of the new incoming request in the *inter-domain network* G.
- $q \mapsto e$ indicates that the intra-path $q \in \mathcal{P}_e$ is a part, represented by $e \in G$, of the working or backup segments of the new incoming request.
- α_e total working bandwidth that the working path p of the new incoming request consumes along its subpath $q \mapsto e \in E$. Thus, $\alpha_e = \sum_{\ell \in q} a_\ell$.
- $\beta^e_{e'}$ (resp. $\beta^{\pi_i}_{e'}$) total additional backup bandwidth that a backup segment of the new incoming request needs to reserve along $q' \mapsto e' \in \pi'_i$ to protect $q \mapsto e \in \pi_i$ (resp. to protect p_i). Thus, $\beta^e_{e'} = \sum_{\ell' \in q'} b^q_{\ell'}$ and $\beta^{\pi_i}_{e'} = \sum_{\ell' \in q'} b^{p_i}_{\ell'}$.
- $\overline{B}_{e'} = \begin{cases} B_{\ell'} & \text{if } e' = \ell' \in L^{\text{INTER}} \\ \max_{\ell' \in L_m} B_{\ell'} & \text{if } e' \in E^{\text{VIRTUAL}}_m \end{cases}$. If e is a virtual link, $\overline{B}_{e'}$ is the maximum backup bandwidth reserved on a physical link of the domain to whom e belongs. If e is an inter-domain link, it is the existing backup bandwidth on e.
- γ^{res}_e maximal bandwidth that can be routed over any intra-path $q \in \mathcal{P}_e$ of $e \in E$. $\gamma^{\text{res}}_e = \max_{q \in \mathcal{P}_e} \min_{\ell \in q} c^{\text{res}}_\ell$.
- $\|e\|$ length of the shortest intra-path represented by e. It is also called the estimated length of e.

The parameters a and b with different indexes denote the working and backup costs of physical links. Similarly, α and β denote the working and backup costs of the virtual and inter-domain links.

III. Costs of virtual and physical links

In this section, the costs of virtual and physical links are presented. We will see later in Section IV that these costs are essential parameters for the proposed routing algorithms.

A. Estimations of the costs of virtual links

The exact values of the costs $\alpha_e, \beta^e_{e'}, \beta^{\pi_i}_{e'}$ of virtual link e or $e' \in E$ depend on intra-path q on p or $p'_i, i \in I$ that e or e' represents, e.g., $q \mapsto e$ or $q \mapsto e'$. However, q, p and $p'_i, i \in I$ are unknown until the routing completion, so that the exact computation of these values is impossible before the routing has been set. Moreover, these costs are associated with the *inter-domain network* where physical link information is inaccessible. Therefore, we will use approximations to define these costs as functions that will be virtual link dependent but physical link independent.

The working cost of $e \in E$ is defined as the smallest total bandwidth that the working path p consumes along e. The choice of taking the smallest total bandwidth but not the average or other estimations is due to the objective of minimizing the bandwidth cost. The intra-path of $\mathcal{P}_e$ with minimum total bandwidth will be the best intra-path that p should go through. Thus:

$$\alpha_e = \begin{cases} \|e\| \times d & \text{if } d \le \gamma^{\text{res}}_e, e \in E^{\text{VIRTUAL}} \\ d & \text{if } d \le \gamma^{\text{res}}_e, e \in L^{\text{INTER}} \\ \infty & \text{otherwise.} \end{cases} \tag{3}$$

The approximation of the backup cost $\beta^{\pi_i}_{e'}$ is more complex. Let us begin with $b^v_{\ell'}$ defined in (1). In order to eliminate the dependency of $b^v_{\ell'}$ on the detailed information $B^v_{\ell'}$, $B^v_{\ell'}$ is overestimated by: $B^v_{\max}$. Recall that $b^v_{\ell'}$ cannot be greater than the requested bandwidth. We get the following overestimation:

$$b^v_{\ell'} = \min\{\max\{0, B^v_{\max} + d - B_{\ell'}\}, d\}. \tag{4}$$

Using (4), it can be shown that the backup cost of a virtual or inter-domain link for protecting a working segment cannot be smaller than the cost for protecting a virtual/inter-domain link of the segment. Thus:

$$\beta^{\pi_i}_{e'} = \max_{e \in \pi_i} \beta^e_{e'}. \tag{5}$$

The cost $\beta^e_{e'}$ is also approximated in its turn. Since $\beta^e_{e'} = \sum_{\ell' \in q'} b^q_{\ell'}$, it is lower bounded by the minimum backup bandwidth that should be reserved along e':

$$\beta^e_{e'} \ge \min_{q \in \mathcal{P}_e, q' \in \mathcal{P}_{e'}} \sum_{\ell' \in q'} b^q_{\ell'}, \tag{6}$$

where

$$b^q_{\ell'} \ge \min\{\max\{0, B^q_{\max} + d - B_{\ell'}\}, d\}$$

as $b^q_{\ell'} = \max_{v \in q} b^v_{\ell'}$ and $B^q_{\max} = \max_{v \in q} B^v_{\max}$.

Thus:

$$\beta^e_{e'} \ge \min_{q \in \mathcal{P}_e, q' \in \mathcal{P}_{e'}} \sum_{\ell' \in q'} \min\{\max\{0, B^q_{\max} + d - B_{\ell'}\}, d\}.$$

Since $\overline{B}_{e'} \ge B_{\ell'}$, for all $\ell' \in q \mapsto e$ then:

$$\beta^e_{e'} \ge \min_{q \in \mathcal{P}_e} \|e'\| \times \min\{\max\{0, B^q_{\max} + d - \overline{B}_{e'}\}, d\}. \tag{7}$$

Let v_1, v_2 be the border end nodes of e and $B^e_{\max} = \max\{B^{v_1}_{\max}, B^{v_2}_{\max}\}$. Clearly, $B^e_{\max} \leq B^q_{\max}$. Thus we have:

$$\beta^e_{e'} \geq \|e'\| \times \min\{\max\{0, B^e_{\max} + d - \overline{B}_{e'}\}, d\}. \quad (8)$$

Let us underestimate $\beta^e_{e'}$ by the right-hand side of (8) which is indeed the lower bound of the backup bandwidth that should be reserved along e' for p'. Taking into account the link capacity, we define:

$$\beta^e_{e'} = \begin{cases} 0 & \text{if } B^e_{\max} + d \leq \overline{B}_{e'} \\ \|e'\| \times (B^e_{\max} + d - \overline{B}_{e'}) & \text{if } B^e_{\max} + d > \overline{B}_{e'} > B^e_{\max} \\ & \text{and } \gamma^{\text{res}}_{e'} \geq B^e_{\max} + d - \overline{B}_{e'} \\ \|e'\| \times d & \text{if } B^e_{\max} \geq \overline{B}_{e'} \text{ and } \gamma^{\text{res}}_{e'} \geq d \\ \infty & \text{otherwise.} \end{cases} \quad (9)$$

In summary, the working and backup costs of a virtual or inter-domain link are represented by functions of the virtual link dependent parameters: $\|e\|$, γ^{res}_e, $\overline{B}_{e'}$, $B^e_{\max}$. These parameters define the link-states of e. Border nodes exchange among themselves these link-states in order to get a common view of the compact *inter-domain network*.

B. Costs of physical links

The working cost a_ℓ of physical link ℓ is exactly defined by:

$$a_\ell = \begin{cases} d & \text{if } d \leq c^{\text{res}}_\ell \\ \infty & \text{otherwise.} \end{cases} \quad (10)$$

Using (4) and the definitions of $b^q_{\ell'}$ and $B^q_{\max}$, it is easy to deduce that: $b^{p_i}_{\ell'} = \min\{\max\{0, B^{p_i}_{\max} + d - B_{\ell'}\}, d\}$, i.e.,:

$$b^{p_i}_{\ell'} = \begin{cases} 0 & \text{if } B^{p_i}_{\max} + d - B_{\ell'} \leq 0 \\ B^{p_i}_{\max} + d - B_{\ell'} & \text{if } B^{p_i}_{\max} + d > B_{\ell'} > B^{p_i}_{\max}, \\ & \quad c^{\text{res}}_{\ell'} \geq B^{p_i}_{\max} + d - B_{\ell'} \\ d & \text{if } B^{p_i}_{\max} \geq B_{\ell'}, c^{\text{res}}_{\ell'} \geq d \\ \infty & \text{otherwise.} \end{cases} \quad (11)$$

IV. Routing solutions

A. Outline of the solution

In this study, the objective of the routing is to minimize the total bandwidth consumed by p and $p'_i, i \in I$ of the new incoming request. It can be expressed as follows:

$$\min \sum_{\ell \in p} a_\ell + \sum_{p'_i, i \in I} \sum_{\ell' \in p'_i} b^{p_i}_{\ell'}. \quad (12)$$

In the *inter-domain network*, it is equivalent to:

$$\min \sum_{e \in \pi} \alpha_e + \sum_{\pi'_i, i \in I} \sum_{e' \in \pi'_i} \beta^{\pi_i}_{e'}. \quad (13)$$

In multi-domain networks, paths tend to be long. In order to guarantee a fast recovery, we require that each working and backup segments are not longer than thresholds l^{W} and l^{B} respectively. This requirement is afterward referred as segment length constraints.

We propose a two-step routing as follows:

- *Inter-domain step*: We first optimize (13) in the *inter-domain network* where virtual and inter-domain links are assigned costs α_e and $\beta^{\pi_i}_e$. The constraints on working and backup segment lengths are also taken into account. This leads to segments π_i and $\pi'_i, i \in I$ as paths of virtual/inter-domain links. If no solution is found, the routing fails. Otherwise, the *intra-domain step* will follow.
  In fact, (13) is an OSSP single-domain routing problem. All OSSP single-domain routing algorithms cited in this paper can be used to solve (13) as long as they are applied on the *inter-domain network* with the proposed virtual link costs and if the segment length constraints are integrated. Two solution schemes, GROS and DYPOS, are proposed in the next two paragraphs IV-B, IV-C.
- *Intra-domain step*: The segment pairs $(\pi_i, \pi'_i), i \in I$ are considered one after the other. For each pair, the virtual links of the working segment are mapped first to the intra-paths with the least working costs:

$$\min_{q \in \mathcal{P}_e} \sum_{\ell \in q} a_\ell (= \alpha_e). \quad (14)$$

  The selected intra-path for the virtual link e is indeed the Shortest Path (SP) in terms of physical working costs a_ℓ between the end nodes of the virtual link. Once the complete working segment p_i is obtained, the virtual links of π'_i will be mapped similarly into the SP but in terms of $b^{p_i}_{\ell'}$:

$$\min_{q' \in \mathcal{P}_{e'}} \sum_{\ell' \in q'} b^{p_i}_{\ell'} (= \beta^{\pi_i}_{e'}). \quad (15)$$

  Note that the nodes along p_i are excluded in this mapping in order to guarantee the disjointness between the working and backup segments of a given pair.
  Each mapping relates to only one domain and can be solved using Dijsktra's SP algorithm within the domain while respecting the *scalability requirement*.

B. GROS: A greedy solution

The first routing solution for the *inter-domain step* is a greedy heuristic denoted by GROS (GReedy Overlapping Short segment shared protection). For each new incoming request, the GROS heuristic works as follows.

1) Working path π is the shortest path in the *inter-domain network* between the source and the destination in terms of working costs α_e.
2) The working path is greedily divided into segments. The first segment π_1 originates from the source node of the working path. The tail node of each segment is chosen so that the segment is as long as possible with a total estimated length that does not exceed l^{W}. However, if no such tail node is found, the node that is closest to the head node will be designated as tail node. From the tail node, we go back toward the head node with the smallest number of hops until reaching a new node with nodal degree larger than 2. This last node will be the head of the next segment. The process continues until the destination node is reached.
3) For each previously identified working segment, a backup segment is computed as the shortest path in terms of backup costs $\beta^{\pi_i}_{e'}$ between the segment end

nodes. The total estimated length of the segment must not be larger than l^{B}. The shortest path with additive constraint algorithm A*Prune (or A*Dijkstra) [15] is used for computing each backup segment.

If the algorithm does not find a solution at a given step, the routing fails. It is necessary to note that, in the GROS heuristic, the working segment length sometimes exceeds threshold l^{W}. In other words, the working segment length constraint is soft.

GROS differs from CDR in [12]. In CDR, a set of segment end nodes are predefined for each pair of source and destination before the working path is identified. From these segment end nodes, the working and backup segments are computed. In GROS, we determine only the segment end nodes once the working path is routed in the *inter-domain network*.

C. DYPOS: A Dynamic programming solution

The second routing solution for the *inter-domain step* is called DYPOS (DYnamic Programming Overlapping Short segment shared protection). It is inspired from the PROMISE dynamic programing solution (PRO-D) [32] for single-domain networks. The difference is in the integration of the working and backup segment length constraints.

Let us first briefly recall PRO-D. In PRO-D, the working path is the shortest path between the source and the destination. The backup segment is computed as follows. Assume that the nodes along the working path are numbered from 0 to T. Let $i \rightarrow j$ denote the working segment from node i to node j. Let D_m be the "best known" solution to protect the part of the working path from node m to node T, T is excluded. D_m divides possibly that part into multiple overlapping segments and protects each of them by one segment. The current D_m is compared with each alternate solution built from $D_i, i \in [m+1..T-1]$ and the least cost backup segment that protects the part $m \rightarrow i$ and overlaps with the part $i \rightarrow T$. The backup segment is denoted by $p'_{m\rightarrow i}$. The best solution will be newly assigned to D_m. The algorithm starts by building the segment for the last hop ($m = T-1$) using the shortest backup path. The protected part is growing up until the entire working path is protected ($m = 0$) (Fig. 5).

In DYPOS, for computing each D_m, we consider only the alternate solutions associated with D_i such that the estimated length of the part $m \rightarrow i$ does not exceed l^{W}. In addition, while computing $p'_{m\rightarrow i}$, we use again the A*Prune algorithm in order to find a backup segment with an estimated length smaller than or equal to l^{B}.

The pseudo-code in Alg.1 describes formally DYPOS. Function $\mathtt{CSP}^{\mathrm{B}}(m, T, l^{\mathrm{B}})$ implements the A*Prune algorithm. It identifies the shortest path from m to T (using the backup cost $\beta^{\pi_i}_{e'}$) that must not be longer than l^{B} (in terms of estimated length). We denote by $||m \rightarrow i||$ the total estimated length of $m \rightarrow i$. $\mathtt{Backup\_seg}\ (m, i, l^{\mathrm{B}})$ computes $p'_{m\rightarrow i}$. The backup segment $p'_{m\rightarrow i}$ must end at a node $j > i$ in order to create overlapping between its working segment and the working part $i \rightarrow T$. $\mathtt{Backup\_seg}\ (m, i, l^{\mathrm{B}})$ identifies, if possible, up to N least cost candidate segments from m to j with $j = [i+1..i+N]$ using $\mathtt{CSP}^{\mathrm{B}}(m, j, l^{\mathrm{B}})$ and returns the least cost one.

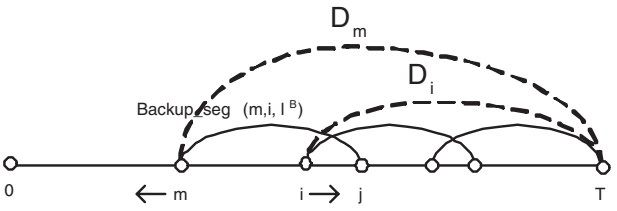

Fig. 5. Working mechanism of the Dynamic programming algorithm

DYPOS differs from GROS as the segment length constraints are hard constraints. If DYPOS finds no solution, it reports a failed routing.

Algorithm 1 DYPOS

for $m = T-1$ down to 0 **do**
  if $||m \rightarrow T|| \leq l^{\mathrm{W}}$ **then**
    $D_m \leftarrow \mathtt{CSP}^{\mathrm{B}}(m, T, l^{\mathrm{B}})$
  else
    $D_m = \infty$
  end if
  for $i = m+1$ to $T-1$ **do**
    if $||m \rightarrow i|| \leq l^{\mathrm{W}} - 1$ **then**
      $p'_{m\rightarrow i} = \mathtt{Backup\_seg}\ (m, i, l^{\mathrm{B}})$
      $D_m \leftarrow \min(D_m, \mathtt{Combine}\ (D_i, p'_{m\rightarrow i}))$
    end if
  end for
end for
return D_0

Algorithm 2 Backup seg (m, i, l^{B})

$bs = \infty$
for $j = i+1$ to $\min(i+N, T)$ **do**
  if $||m \rightarrow j|| \leq l^{\mathrm{W}}$ **then**
    $bs \leftarrow \min(bs, \mathtt{CSP}^{\mathrm{B}}(m, j, l^{\mathrm{B}}))$
  end if
end for
return bs

D. Blocking-go-back option

A request may be successfully routed at the *inter-domain step* but blocked at the *intra-domain step* because of insufficient bandwidth for mapping a virtual link or of the impossibility of mapping a virtual link of a backup segment while maintaining the disjointness with its working segment. Let us call *blocking virtual link* the virtual link where the blocking occurs. In order to avoid such blocking cases, a second routing iteration is added to GROS and DYPOS. The second routing iteration is identical to the first one except that in the *inter-domain step*, the blocking virtual link is removed before the working path or backup segment computation, depending on whether the blocking virtual link was on the working path or backup segments. This removal prevents from repeating the previous blocking. Then the intra-domain step, as described in IV-A, is applied again. A failed routing is reported if a new blocking is produced.

Although the Blocking-go-back step allows the reduction of blocking probability, it takes longer time to route a request when the routing fails at the first routing iteration. The routing time without the Blocking-go-back step is: $T_r = T_r^{\text{INTER}} + T_r^{\text{INTRA}}$ where T_r^{INTER} (resp. T_r^{INTRA}) is the execution time of the *inter-domain step* (resp. *intra-domain step*). When the routing is blocked, the Blocking-go-back step is triggered. An extra routing time is introduced from the second routing iteration. The total routing time is $T_r = 2 \times T_r^{\text{INTER}} + 2 \times T_r^{\text{INTRA}}$. In other words, the routing time with Blocking-go-back is twice the routing time without Blocking-go-back. We verified this through experimental results on two multi-domain networks LARGE-5 and LARGE-8 which will be described later. In LARGE-5, the average routing time when the Blocking-go-back step is involved is 89.06 milliseconds while it takes 37.16 milliseconds without the Blocking-go-back step. In LARGE-8, the average routing time with Blocking-go-back is 329.71 milliseconds, and without Blocking-go-back it is 134.09 milliseconds. Recall that this routing time sacrifice is compensated in lower blocking probability.

V. Signaling and routing information update

Contrary to the single-domain OSSP routings of the literature, the multi-domain OSSP routing must be performed in a distributed way in different domains and requires signaling processes for coordinating the segment computation, segment setup and routing information update. We will not discuss here the details of how the signaling protocols should be implemented as well as which message formats should be used. We describe only the interaction among network nodes.

A. Signaling for working and backup segment computation

The *inter-domain step* is performed centrally at the border source node without impairing the *scalability requirement* since the *inter-domain network* is considered as a single-domain network. In this step, the border source node computes the working and backup costs $\alpha_e, \beta_{e'}^{\pi_i}$ for each link of the *inter-domain network* by using link-states $\|e\|$, γ_e^{res}, $\overline{B}_{e'}$, $B_{\max}^e$ which are available at each border node thanks to the routing information update process that will be described later. Then GROS or DYPOS can be used for performing the *inter-domain step*. Once the computation is finished, the border source node asks the other border nodes along its working and backup segments to map subsequently the adjacent virtual links into intra-paths.

At the reception of the mapping request, a border node triggers the *intra-domain step* within its domain. It first computes the costs $a_\ell, b_\ell^{p_i}$ using the detailed information available in the domain and then solves the mapping problems (14) and (15). The border node returns the mapped intra-path to the border source node.

From the mapped intra-paths, the border source node builds the complete working and backup segments.

B. Signaling for working and backup segment setup

A message carrying the information of the complete working path and backup segments is propagated along the working path from the source node to the destination node. At each node on the working path, switch is made in order to establish the end-to-end working path. At each segment head node an additional message is created carrying the information on the corresponding backup segment. The message is propagated along the route of the backup segment until the segment tail node. At each node, it asks for reserving an additional amount of bandwidth $b_{\ell'}^v$ on its outgoing link that belongs to the backup segment. Note that no switch is made there. The process terminates when the destination node is reached.

C. Routing information update

After each routing, link-states of virtual links change. They should be updated for serving the *inter-domain step* for the next request. Link-states $\|e\|$, γ_e^{res}, $\overline{B}_{e'}$, $B_{\max}^e$ are computed locally in the domain containing e by a border node of e. The node writes all these link-states in one message and sends it to the other border nodes of the multi-domain network. A BGP like protocol could be used for link-state message diffusion.

Of course, for computing the link-states of e, the border nodes of e also need the detailed routing information of their domain. Routing information exchange among nodes of the domain is also needed.

Routing information update is the most expensive process regarding the flow of messages. A number of $O({V^{\text{BORDER}}}^2)$ messages are exchanged among border nodes and of $O(V_m^2)$ within each domain leading to an overall number of $O({V^{\text{BORDER}}}^2) + \sum_{m=1}^{M} O(V_m^2)$ messages. Nevertheless, this number is still smaller than $O(V^2) = O((\sum_{i=1}^{M} V_m)^2)$, the number of messages required by a single-domain solution.

For reducing furthermore the load of update message flows, the update could be triggered less regularly in a time driven way. However, the routing will be less accurate since some routing information will be out of date.

VI. Experimental results

We use different network and traffic instances for evaluating the efficiency of GROS and DYPOS. The following metrics: backup overhead and overall blocking probability are used for their performance evaluation.

A. Metrics

The working network cost is defined as the total working bandwidth used by all network links. The network cost is defined as the total working and backup bandwidth used by all network links.

The *Backup overhead* is defined as the ratio between the network cost and the smallest working network cost minus 1. This amounts to the backup bandwidth redundancy of a protection scheme. The smallest working network cost can be obtained when all working paths are the shortest paths.

The *Overall blocking probability* is defined as the percentage of the total rejected bandwidth out of the total bandwidth requested by all connections.

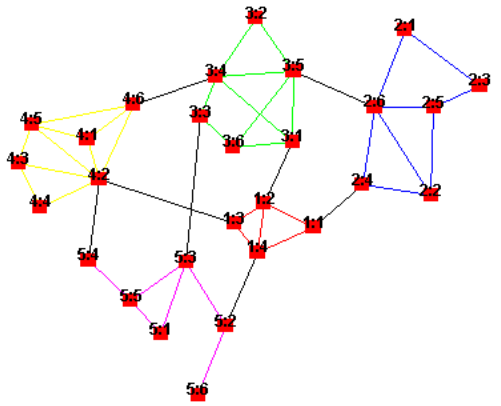

Fig. 6. SMALL-5 network.

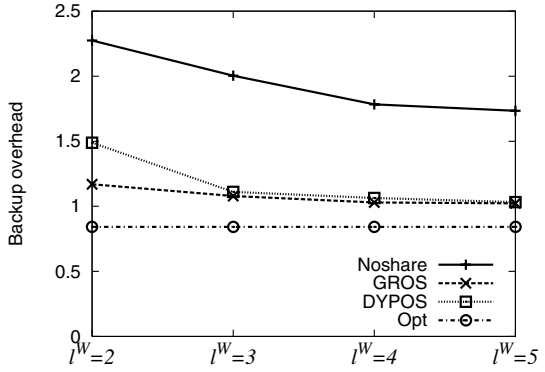

Fig. 7. Backup overhead in SMALL-5.

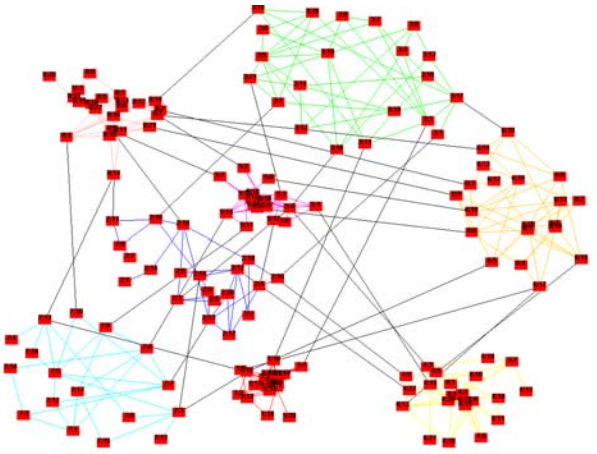

Fig. 8. LARGE-8 network.

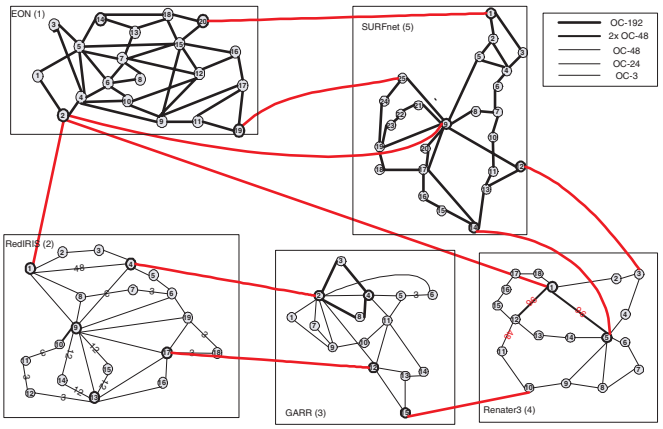

Fig. 9. LARGE-5 network.

B. Comparison with optimal single-domain solution

We first evaluate the efficiency of GROS and DYPOS by comparing their results on a multi-domain network with the results of the single-domain optimal solution [12], denoted by Opt, on the equivalent flattened network. Due to the extremely high computational effort required by Opt, the comparison is made only on a small 5-domain network of 28 nodes with 70 requests. The Transit-Stub model of GT-ITM [33], a well known multi-domain network generator, is used for generating this network instance that we denote by SMALL-5 and present in Fig. 6. To route a request, GROS and DYPOS take few milliseconds whereas Opt takes few minutes. That means GROS and DYPOS are thousands of times faster than Opt. In a larger network, we could not obtain any result using Opt. Due to the small size of SMALL-5, the constraint on backup segment length is ignored in GROS and DYPOS. In Opt, neither working nor backup segment lengths are restricted. We made also comparison with the results obtained from dedicated protection denoted by NoShare.

Fig. 7 shows that the proposed two-step solution with either GROS or DYPOS provides a backup overhead close to the backup overhead of Opt and far better than the backup overhead of NoShare. In other words, GROS and DYPOS yield very good bandwidth saving rates. Do not forget that the constraint on l^{W} is present in GROS and DYPOS, while it is absent in Opt, therefore giving a slight advantage to Opt. Recall also that, while GROS and DYPOS are scalable for multi-domain networks, Opt is clearly not. In this experiment and also in others afterward, DYPOS yields sometimes larger backup overhead than GROS due to the working segment length constraint that is hard in DYPOS and soft in GROS. That forces DYPOS to consider solutions with larger costs than those of GROS if the latter violate the working segment length constraint. This phenomena reduces when l^{W} increases.

C. Backup overhead

From now on, the experiments are made on larger multi-domain networks with heuristics only. The Transit-Stub model of GT-ITM, is again used for generating a larger multi-domain network with 8 domains, 36 inter-domain links and 60 border nodes. The network is denoted by LARGE-8 and is shown in Fig. 8. Each domain has in average 4 neighboring domains. According to [17], this number reflects faithfully the Internet interconnections. The numbers of nodes and links of each domain are: $(20, 53)$, $(20, 29)$, $(21, 48)$, $(22, 41)$, $(18, 36)$, $(20, 44)$, $(17, 27)$, $(22, 47)$, see [2] for the details of the topology.

We also consider another multi-domain network that we used for experiments in previous papers [13], [28], [29]. The network is built from 5 real optical networks: EON [20], RedIris [5], Garr [1], Renater [3], SURFnet [4]. Inter-domain

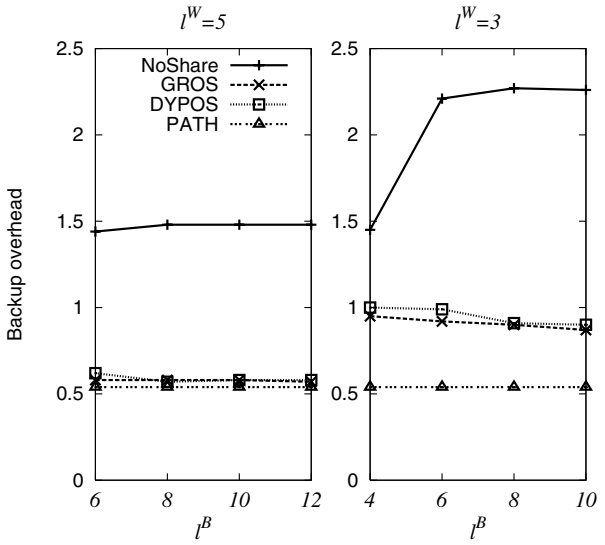

Fig. 10. Backup overhead in LARGE-8.

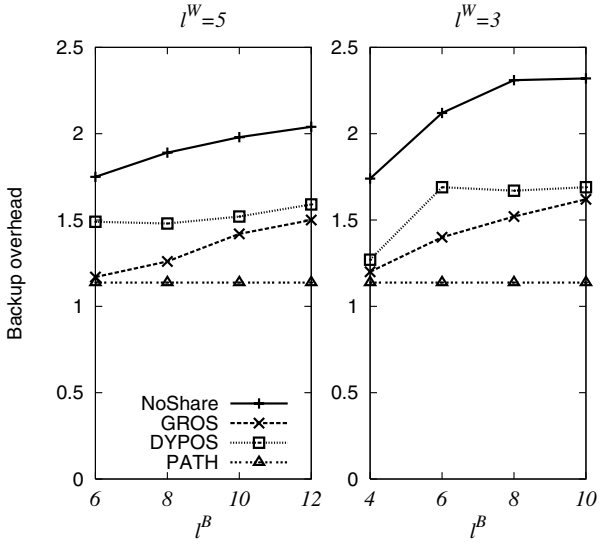

Fig. 11. Backup overhead in LARGE-5.

links are added with capacity OC-192. The network is denoted by LARGE-5 and is shown in Fig. 9.

An incremental traffic is generated by submitting iteratively 1000 requests to the network with all requests remaining active. The incremental traffic allows keeping active more requests in the network and thus allows evaluating more accurately the bandwidth allocation characteristic of each solution scheme. Network links are uncapacitated in order to avoid the impact of blocking which is different from one protection scheme to the other. Backup overhead is computed after 1000 requests.

Fig. 10 depicts the obtained backup overheads in LARGE-8 when using GROS and DYPOS in comparison with those of NoShare and WPF. This last scheme is a multi-domain Shared Path Protection proposed in [29], which will be renamed PATH in order to distinguish it from the other protection schemes, i.e., of segment type. The working segment length thresholds are $l^{\mathrm{W}} = 3$ and $l^{\mathrm{W}} = 5$ and backup segment length threshold varies. Similar backup overheads are found in GROS and DYPOS with both $l^{\mathrm{W}} = 3$ and $l^{\mathrm{W}} = 5$. We observe that GROS and DYPOS require only around 0.55 and less than 1 time the working capacity for their backup, meanwhile NoShare requires 1.5 and up to 2.2 times the same amount with $l^{\mathrm{W}} = 5$ and $l^{\mathrm{W}} = 3$ respectively. In LARGE-5 (Fig. 11), we find a smaller but still significant difference between the backup overheads of NoShare and of the other schemes. This shows the advantage of shared protection over dedicated protection as well as the efficiency of GROS and DYPOS in favoring backup bandwidth sharing.

In general, we should also accept that OSSP (GROS and DYPOS in particular) can use more backup resources than Shared Path Protection (PATH in particular) because the backup segments need some extra links in order to join the working path at segment end nodes. For some configurations, for example when $l^{\mathrm{W}} = 5$ in LARGE-8 or when $l^{\mathrm{W}} = 5, l^{\mathrm{B}} = 6$ and $l^{\mathrm{W}} = 3, l^{\mathrm{B}} = 4$ in LARGE-5, OSSP can be close to Shared Path Protection. In spite of the disadvantages with backup overhead, OSSP is still attractive due to its fast recovery characteristic which will be presented later in other experiments.

D. Blocking probability

The blocking probability is examined under dynamic traffic. In dynamic traffic, connections arrive and tear down after a holding time. Requests arrive according to a Poisson process with rate $r = 1$ and holding time exponentially distributed with mean $h = 320$. There are, on average, 320 active connections in the network.

Fig. 12 depicts the overall blocking probability of GROS, of GROS with the Blocking-go-back option (denoted by GROS-BGB), of DYPOS and of DYPOS with the Blocking-go-back option (denoted by DYPOS-BGB) in LARGE-8. The four schemes keep NoShare at a distance. In LARGE-5 (see Fig. 13), the four schemes are still better than NoShare when $l^{\mathrm{W}} = 5$. However, when $l^{\mathrm{W}} = 3$, DYPOS and DYPOS-BGB become worse than GROS and sometimes even than NoShare. Two explanations can be given. First, the constraint on working segment length is hard in DYPOS and soft in the others. Second, LARGE-5 is less connected than LARGE-8 leading to less opportunity for dividing working paths into segments of 3 hops or less. This shows also the relevance to properly define segment lengths in low connected networks.

The blocking probabilities drop off for all schemes in both network topologies when the Blocking-go-back option is adopted. Fig. 14 and 15 show more clearly the advantage of the Blocking-go-back step. The GROS Inter and DYPOS Inter curves depict the percentages of the requests that are successfully routed in the *inter-domain step* of the second routing. Similarly, the GROS Intra and DYPOS Intra curves depict the percentages of the requests that are successfully routed in the *intra-domain step* of the second routing. A large number of requests that fails in the first routing is successfully routed in the *inter-domain step* of the second routing and about 30%-50% of them are successfully routed in the *intra-domain step* except when thresholds become two small, e.g. $l^{\mathrm{W}} = 3, l^{\mathrm{B}} = 4$. We can conclude that the Blocking-go-back step is useful for increasing the grade of service.

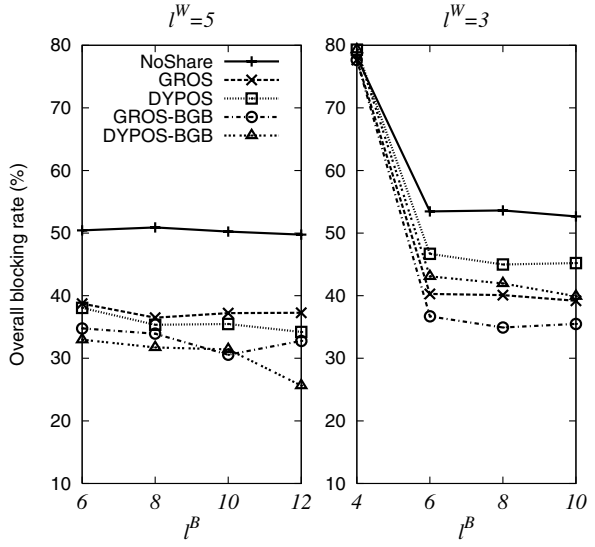

Fig. 12. Overall blocking probabilities in LARGE-8.

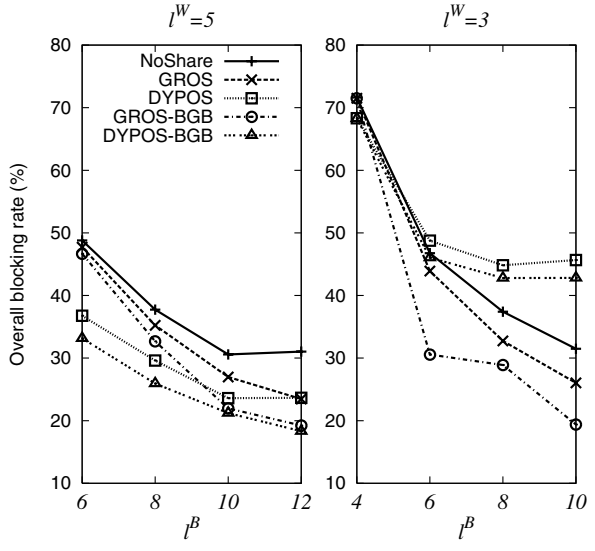

Fig. 13. Overall blocking probabilities in LARGE-5.

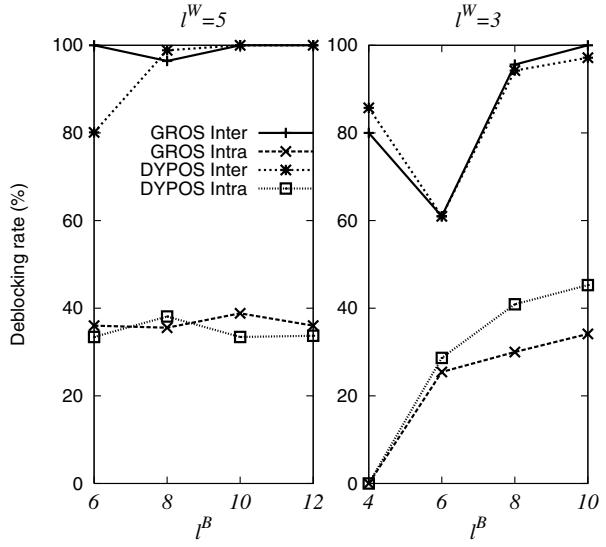

Fig. 14. De-blocking capacity of the Blocking-go-back step in LARGE-8.

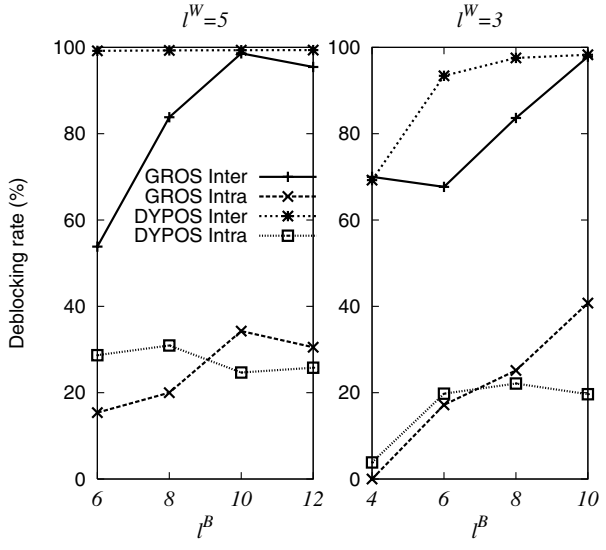

Fig. 15. De-blocking capacity of the Blocking-go-back step in LARGE-5.

E. Impact of segment length

Fig. 16 and 17 show the average working segment lengths of GROS and DYPOS in comparison with other schemes in LARGE-5 and LARGE-8. Fig. 18 and 19 show the average backup segment lengths of these schemes. For GROS and DYPOS, since the segment length constraints are enforced in the inter-domain step with estimated virtual link lengths, the actual segment lengths in the original multi-domain network may exceed the thresholds l^{W} and l^{B}. However, the experiment results still show that the average working and backup segment lengths are always under the given limits l^{W} and l^{B}. The results with $l^{\mathrm{B}} = \infty$ corresponds to the elimination of the backup segment length constraint. Logically, working segments are longer when threshold l^{W} increases from 3 to 5. Similarly, for a given working segment length threshold l^{W}, the average backup segment length becomes generally larger when l^{B} increases. Note that in both LARGE-5 and LARGE-8, 10% to 50% of the solutions contain from 2 to 5 segments.

In Fig. 19, we observe that the backup segment lengths of NoShare are nearly constant with $l^{\mathrm{B}} \geq 6$ and that they are smaller than those of GROS and DYPOS. This can be explained by the fact that the backup segments of GROS and DYPOS take sometimes longer routes in order to benefit from the sharable backup bandwidth on some network links. On the contrary, NoShare does not allow backup bandwidth sharing, it has thus no interest to take these long routes, it simply takes the shortest routes in order to get the smallest costs.

We also run DYPOS without both working and backup segment length constraints. This is similar to using PROMISE on the inter-domain routing. The corresponding results are denoted by PROMISE-Inter. The results show that GROS and DYPOS always give smaller average working segment lengths (Fig. 16, 17) and mostly give smaller average backup segment lengths (Fig. 18 and 19) than PROMISE-Inter. This illustrates the role of segment length constraints implemented in GROS and DYPOS. This demonstrates also that GROS and DYPOS recover from failures faster than PROMISE-Inter.

In comparison with the multi-domain Shared Path Pro-

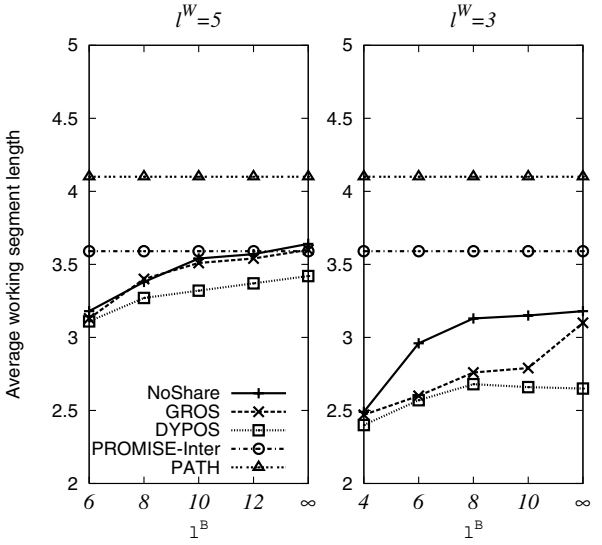

Fig. 16. Average working segment lengths in LARGE-5

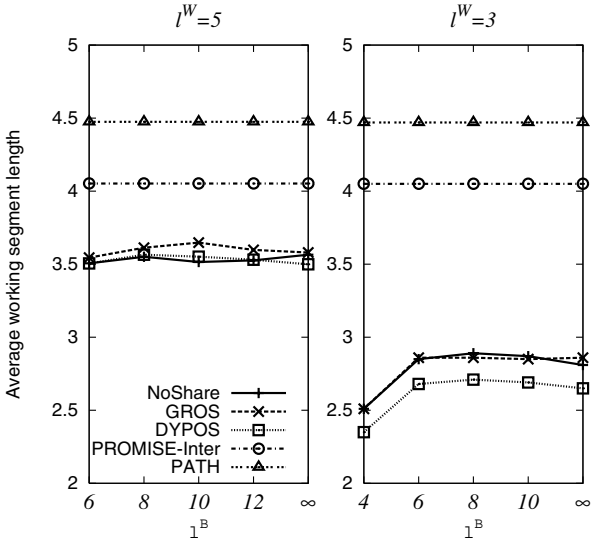

Fig. 17. Average working segment lengths in LARGE-8

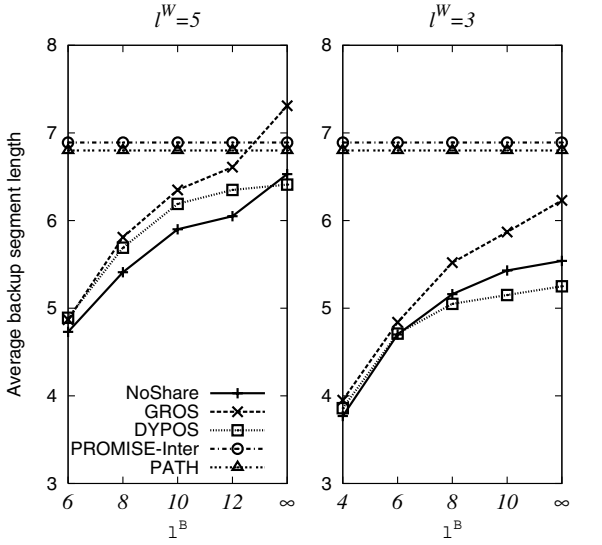

Fig. 18. Average backup segment lengths in LARGE-5

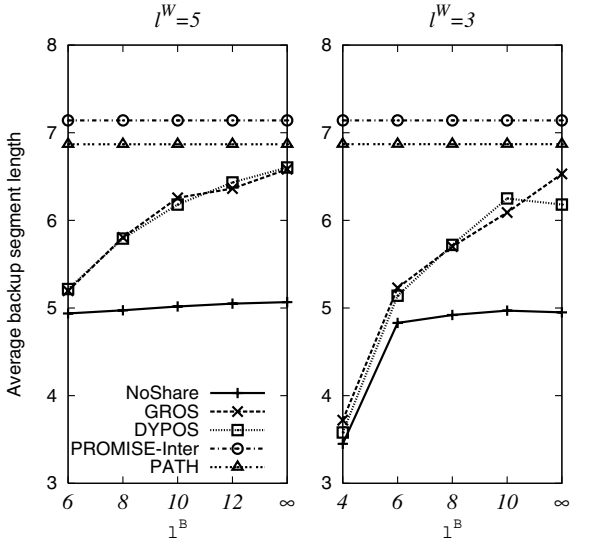

Fig. 19. Average backup segment lengths in LARGE-8

tection scheme PATH, the average working segment lengths of GROS and DYPOS are clearly smaller than the average working path lengths of PATH (Fig. 16 and 17). Backup segments of GROS and DYPOS are also generally shorter than backup paths of PATH (Fig. 18 and 19). As a result, the recovery times of GROS and DYPOS are shorter than that of PATH.

Although small segment lengths promise fast recovery, they sometimes impairs backup overhead. When the segment length thresholds are too small, there are few choices for the division of working paths and for the building of backup segments. This leads to the selection of solutions with high backup costs which satisfy the segment length constraints. As a result, the overall backup overhead increases. Indeed, in LARGE-8 as shown in Fig. 10, backup overhead increases from around 0.55 when $l^{\mathrm{W}} = 5$ to around 0.8 when l^{W} reduces to 3. An increment is also found with LARGE-5 in Fig. 11, but it is smaller.

Again, too small segment length thresholds make the blocking probability worse. There might be no solution satisfying the required working and backup segment lengths. This is illustrated in Fig. 12 and 13. The blocking probability increases slightly from $l^{\mathrm{W}} = 5$ to $l^{\mathrm{W}} = 3$ in the case of LARGE-8 and even more in the case of LARGE-5. In both topologies, at $l^{\mathrm{W}} = 3$, the blocking probabilities raise up drastically when the backup segment length threshold reduces (to $l^{\mathrm{B}} = 4$). A smaller impact is observed with $l^{\mathrm{W}} = 5$ in LARGE-5 and barely visible in LARGE-8 because the thresholds are still large enough to provide a reasonable number of segment choices.

VII. Conclusion

In this paper, we have presented a two-step routing solution for OSSP in multi-domain networks. The solution is scalable for multi-domain networks thanks to the use of Topology Aggregation. A greedy and a dynamic programming algorithms, GROS and DYPOS, with and without Blocking-go-back option are also proposed for the *inter-domain step*. The

comparison with optimal single-domain solution shows the efficiency of GROS and DYPOS. Other experiments illustrate that GROS and DYPOS promote well the backup bandwidth sharing. They also show the advantage of the Blocking-go-back phase in reducing the blocking probability.

As the working and backup segments are restricted in length, the proposed solutions guarantee fast recovery. The experimental results show that these solutions can offer a faster recovery than Shared Path Protection as well as the other OSSP solution without segment length restriction.

Obviously, the smaller the segment lengths are, the shorter the recovery is. However, the experimental results show that segment length thresholds should be set carefully as too small thresholds may entail a significant increase of the blocking probability as well as of the backup overhead.

ACKNOWLEDGMENT

The work of the second author has been supported by the NSERC (Natural Sciences and Engineering Research Council of Canada) grant GP0036426 and the Concordia University Research Chair on Optimization of Communication Networks.

REFERENCES

[1] "Consortium GARR," http://www.garr.it.
[2] "LARGE-5 and LARGE-8 networks," http://www.iro.umontreal.ca/~truongtd/topo/.
[3] "RENATER-4 network," http://www.renater.fr.
[4] "Surfnet," http://www.surfnet.nl.
[5] "REDIrid," 2005, http://www.rediris.es/red/index.en.html#red%20troncal.
[6] A. Akyamac, S. Sengupta, J.-F. Labourdette, S. Chaudhuri, and S. French, "Reliability in single domain vs. multi domain optical mesh networks," in *Proc. National Fiber Optic Engineers Conference*, Sept. 2002.
[7] G. Bernstein, V. Sharma, and L. Ong, "Interdomain optical routing," *OSA J. Optical Networking*, vol. 1, no. 2, pp. 80–92, Feb. 2002.
[8] J. Cao, L. Guo, H. Yu, and L. Li, "A novel recursive shared segment protection algorithm in survivable WDM networks," *J. Network and Computer Application*, vol. 30, no. 2, pp. 677–694, 2007.
[9] W. D. Grover and D. Stamatelakis, "Cycle-oriented distributed pre-configuration: ring-like speed with mesh-like capacity for self-plaining network restoration," in *Proc. IEEE ICC 1998*, pp. 537–543.
[10] P.-H. Ho and H. T. Mouftah, "A framework for service-guaranteed shared protection in WDM mesh networks," *IEEE Commun. Mag.*, vol. 40, no. 2, pp. 97–103, 2002.
[11] ——, "Spare capacity allocation for WDM mesh networks with partial wavelength conversion capacity," in *Proc. Workshop on High Performance Switching and Routing 2003*, pp. 195–199.
[12] P.-H. Ho, J. Tapolcai, and T. Cinkler, "Segment shared protection in mesh communications networks with bandwidth guaranteed tunnels," *IEEE/ACM Trans. Networking*, vol. 12, no. 6, pp. 1105–1118, 2004.
[13] B. Jaumard and D. L. Truong, "Backup path re-optimizations for shared path protection in multi-domain networks," in *Proc. IEEE Globecom 2006*, Nov. 2006.
[14] M. Kodialam and T. Lakshman, "Dynamic routing of bandwidth guaranteed tunnels with restoration," in *Proc. IEEE INFOCOM 2000*, pp. 902–911.
[15] G. Liu and K. G. Ramakrishnan, "A*Prune: an algorithm for finding K shortest paths subject to multiple constraints," in *Proc. Infocom 2001*, vol. 1, Feb. 2001, pp. 743–749.
[16] H. Luo, H. Yu, and L. Li, "A heuristic algorithm for shared segment protection in mesh WDM networks with limited backup path/segments length," *ScienceDirect/Computer Commun.*, vol. 29, no. 16, pp. 3197–3213, Oct. 2006.
[17] D. Magoni and J. J. Pansiot, "Analysis of the autonomous system network topology," *SIGCOMM Computer Commun. Rev.*, vol. 31, no. 3, pp. 26–37, 2001.
[18] T. Miyamura, T. Kurimoto, M. Aoku, and A. Misawa, "An inter-area SRLG-disjoint routing algorithm for multi-segment protection in GMPLS networks," in *Proc. ICBN Conference*, Apr. 2004.
[19] B. Mukherjee, *Optical WDM Networks*. Springer, 2006.
[20] M. O'Mahony, D. Simeonidu, A. Yu, and J. Zhou, "The design of the European optical network," *J. Ligthwave Technol.*, vol. 13, no. 5, pp. 817–828, 1995.
[21] C. Ou, B. Mukherjee, and H. Zang, "Sub-path protection for scalability and fast recovery in WDM mesh networks," in *Proc. OSA Optical Fiber Communication Conference (OFC)*, vol. 54, Feb. 2001, p. ThO6.
[22] S. Ramamurty and B. Mukherjee, "Survivable WDM mesh networks, part I-protection," in *Proc. IEEE INFOCOM*, vol. 2, Mar. 1999, pp. 744–751.
[23] G. Ranjith, G. P. Krishna, and C. S. R. Murthy, "A distributed primary-segmented backup scheme for dependable real-time communication in multihop networks," in *Proc. International Parallel and Distributed Processing Symposium*, Apr. 2002, pp. 139–146.
[24] J. L. L. Roux, J. P. Vasseur, and J. Boyle, "Requirements for inter-area MPLS traffic engineering," IETF Internet-Draft, draft-ietf-tewg-interarea-mpls-te-req-02.txt, Tech. Rep., June 2004.
[25] G. Shen and W. D. Grover, "Extending the p-cycle concept to path segment protection for span and node failure recovery," *IEEE JSAC Optical Communications and Networking*, vol. 21, no. 8, pp. 1306–1319, Oct. 2003.
[26] ——, "Segment-based approaches to survivable translucent network design under various ultra-long-haul system reach capabilities," *OSA J. Optical Networking*, vol. 3, no. 1, pp. 1–24, Jan. 2004.
[27] J. W. Suurballe and R. E. Tarjan, "A quick methods for finding shortest pairs of disjoint paths," *Networks*, vol. 14, no. 2, pp. 325–336, 1984.
[28] D. L. Truong and B. Jaumard, "Overlapped segment shared protection in multi-domain networks," in *Proc. APOC*, vol. 6354, Sept. 2006, pp. 63 541K–1–63 541K–10.
[29] D. L. Truong and B. Thiongane, "Dynamic routing for shared path protection in multidomain optical mesh networks," *OSA J. Optical Networking*, vol. 5, no. 1, pp. 58–74, Jan. 2006.
[30] D. Xu, C. Qiao, and Y. Xiong, "An ultra-fast shared path protection scheme distributed partial information management, part II," in *Proc. 10th IEEE International Conference in Network Protocols*, Nov. 2002, pp. 344–353.
[31] D. Xu, Y. Xiong, and C. Qiao, "Protection with multi-segments (PROMISE) in networks with shared risk link groups (SRLG)," in *Proc. 40th Annual Allerton Conference on Communication*, 2002, pp. 1320–1331.
[32] ——, "Novel algorithms for shared segment protection," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 8, pp. 1320–1331, Oct. 2003.
[33] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. IEEE Infocom*, vol. 2, Mar. 1996, pp. 594–602.

Dieu-Linh Truong received her Eng. and MSc. in Computer Science at Hanoi University of Technology in 1999 and Institut de la Francophonie pour l'Informatique in 2001, respectively. She obtained her a Ph.D. in Computer Science in the Department of Computer Science and Operations Research at Université de Montréal in 2007. Her interests include multi-domain networks, protection and routing problems.

Brigitte Jaumard holds a Concordia University Research Chair, Tier 1, on the Optimization of Communication Networks in Concordia Institute for Information Systems and Engineering, Concordia University. She was previously awarded a Canada Research Chair, Tier 1, in the Department of Computer Science and Operations Research at Université de Montréal. She is also member of the Group for Research in Decision Analysis (GERAD), a multi university research center, and the Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT). She is an active researcher in combinatorial optimization and mathematical programming, with a focus on applications in telecommunications and artificial intelligence. Recent contributions include the development of efficient methods for solving large-scale mathematical programs, their applications to the design and the management of optical, wireless and 3G/4G networks as well as the development of efficient optimization algorithms for probabilistic logic and automated mechanical design. She has published over 150 papers in international journals in Operations Research and in Telecommunications.