Les Cahiers du GERAD ISSN: 0711–2440

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

D.-L. Truong
B. Jaumard

G–2006–66

November 2006

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

Dieu-Linh Truong

Department of Computer Science and Operations Research
Université de Montréal
Montréal (Québec) Canada H3C 3J7
truongtd@iro.umontreal.ca

Brigitte Jaumard

GERAD and Concordia Institute for Information Systems Engineering
Concordia University
Montréal (Québec) Canada H3G 1M8
bjaumard@ciise.concordia.ca

November 2006

Les Cahiers du GERAD

G–2006–66

Abstract

The dynamic routing problem for Overlapped Segment Shared Protection in multi-domain networks has not received a lot of interest so far as it is more difficult than in single-domain network. Difficulties lie in the lack of complete and global knowledge about network topology and bandwidth allocation meanwhile this knowledge is assumed to be easily available in single-domain networks. We propose a two-step routing approach for this problem based on a topology aggregation scheme and link cost estimations: 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" feature in order to reduce the blocking rate in the intra-domain step. We compare the performance of the two heuristics, and evaluate their solutions against an optimal single-domain solution. It shows that the proposed heuristics lead to resource efficient solutions that are not far from the optimal one. Moreover, both heuristics require a quite small computational effort and are scalable for multi-domain networks.

Key Words: Multi-domain Network, Protection, Routing.

Résumé

Le problème de routage pour la protection par des segments qui se chevauchent dans les réseaux multi-domaines n'a pas fait beaucoup l'objet d'études parce qu'il est plus difficile à résoudre que celui des réseaux dans un seul domaine. La difficulté vient du manque de connaissance globale de la topologie du réseau et de l'allocation des ressources dans les réseaux multi-domaines, à l'opposé de la disponibilité de la connaissance complète de ces informations dans un réseau appartenant à un unique domaine. Pour résoudre le problème de protection dans les réseaux multi-domaines, nous proposons une approche de routage à deux étapes en se basant sur une agrégation de la topologie et des estimations des coûts des liens. Les deux étapes sont : une étape inter-domaine et une étape intra-domaine. Nous proposons deux heuristiques différentes, GROS et DYPOS pour l'étape inter-domaine, ainsi qu'une option "Blocking-go-back" afin de réduire le taux de blocage à l'étape intra-domaine. Nous comparons la performance des heuristiques et comparons leurs solutions par rapport à une solution optimale obtenue en considérant que toute l'information est disponible dans le réseau multi-domaines. Les résultats d'expérimentation démontrent que les heuristiques proposées mènent à des solutions efficaces en termes de ressources utilisées, qui ne sont pas loin de la solution optimale. De plus, les deux heuristiques exigent un petit effort de calcul et peuvent facilement être utilisées pour des réseaux multi-domaines de grande taille sans perdre de leur efficacité.

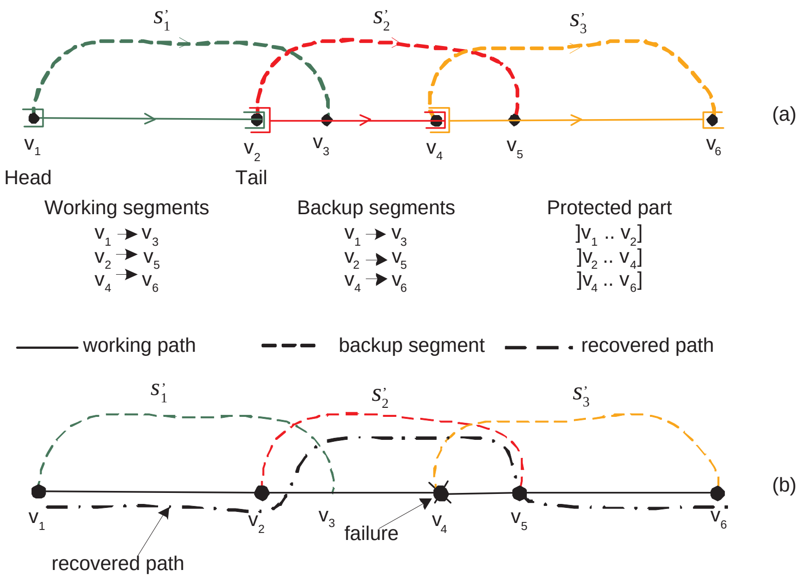

Figure 1: Example of Overlapped Segment Protection when v_4 fails.

1 Introduction

In segment protection, an end-to-end working path is divided into segments and each one is protected by a unique backup segment. Only one backup segment is activated upon a single link or node failure, the other working segments that are not impaired by the failure are still used. As a result, segment protection offers a faster recovery than that of path protection. In the classical segment protection, working segments are non-overlapping. Segment end nodes are then not protected because the failures of those nodes impair both the working and backup segments. Overlapped Segment Protection, firstly proposed in [1] and [2], overcomes this weakness thanks to the overlapping between working segments (see Figure 1) while still inheriting the fast recovery of segment protection.

For achieving backup bandwidth efficiency, shared protection has been proposed for link, path or segment protection [3]. In Segment Protection, in order to guarantee 100% recovery under a single link or node failure, two backup segments can share some bandwidth between them if only if their working segments are link and node-disjoint. This is called *Segment sharing condition*, see Figure 2 for 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_3, v_4) 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_1, therefore their backup segments must reserve separate backup bandwidth. The

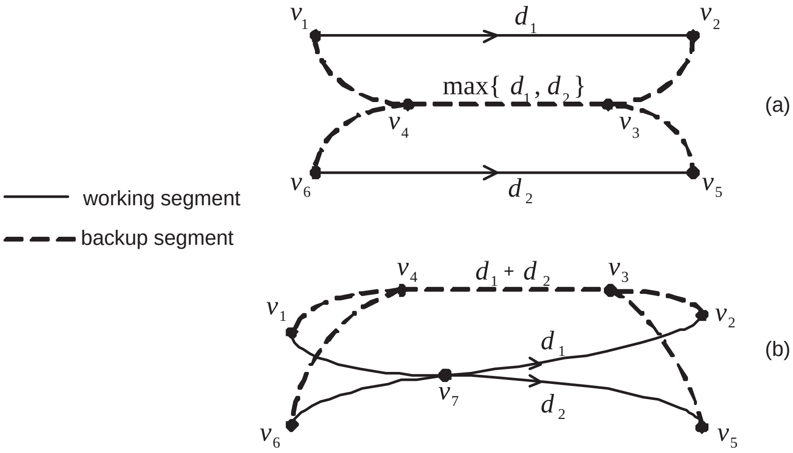

Figure 2: Examples of backup bandwidth sharable (a) and non-sharable (b) cases.

total backup bandwidth for both backup segments on link (v_3, v_4) is $d_1 + d_2$ which is greater than in case (a).

With the shared protection feature, Overlapped Segment Protection becomes Overlapped Segment Shared Protection(OSSP). This paper aims at solving the OSSP routing problem in multi-domain networks with the objective of minimizing the total working and backup bandwidth capacity required by a request under a dynamic traffic pattern.

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 [4] and further developed for segment protection in [5], [6]. However, today network traffic changes unpredictably and dynamically, static traffic is no longer an appropriate assumption. For this reason, we focus only on dynamic traffic. Each request for connection should be routed without any forecast about the upcoming requests.

A multi-domain network is an interconnection of several single-domain networks [7] (Figure 3a). For the scalability requirement, only the aggregate routing information can be exchanged between domains [8] by an Exterior Gateway Protocol (EGP) such as Border Gateway Protocol (BGP) (this is so called *"scalability constraint"*). Consequently, a given node is aware neither of the global multi-domain network topology nor of the detailed bandwidth allocation on all network links. 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 Open Shortest Path First (OSPF), Routing Information Protocol (RIP) etc..

Most studies on OSSP remain within the single-domain network context. An optimal solution has been proposed in [9] although it requires a huge computational effort even for small networks. Several heuristics with smaller computational effort have been proposed such as the work in [1], SLSP-O in [10], CDR in [2] or PROMISE in [11]. The first study ignores the sharing possibility during the routing. The other ones as well as the optimal solution scheme [9] are restricted to single domain networks because of the assumption that the global and detailed information is available at any given core node.

Some solutions have also been proposed for multi-domain networks with drawbacks. In [12], the working path is divided into non-overlapping segments at domain border nodes. The border nodes are thus not protected. In [13], the authors try to cover those border nodes by using an end-to-end restoration rather than a protection when they fail. In [14], a simple multi-domain network without transit domain is assumed. A connection from one domain goes directly to an another one through some backbone links. In practice, a connection between distant domains goes often through one or more transit domains making the routing problem more complex.

We develop a two-step heuristic solution. The multi-domain network is first topologically aggregated 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 aggregate 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 4.2 and 4.3) on any single-domain solution. The working and backup segment lengths are also restricted in order to guarantee a fast recovery. In published OSSP solutions, this restriction is not considered leading usually to solutions with single segment patterns. The segment protection solutions erode to path protection solutions. The cost may decrease but the recovery time increases.

In this study, we consider networks with bandwidth guaranteed connections such as optical, SONET/SDH, MPLS-TE and ATM networks. In the case of optical networks, each network node is assumed to be equipped by Multi Service Provisioning Platform (MSPP, see i.e. [15]) with bandwidth grooming and wavelength conversion capacities. The wavelength continuity and wavelength assignment problems are thus relaxed.

This paper is organized as follows: Notations and fundamental concepts are introduced in the next section. Section 3 presents link costs which will be used in the routing algorithms proposed in Section 4. Section 5 outlines the signaling processes that coordinate the routing, the connection setup as well as the information update. Section 6 shows the computational results. Section 7 concludes the paper.

2 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

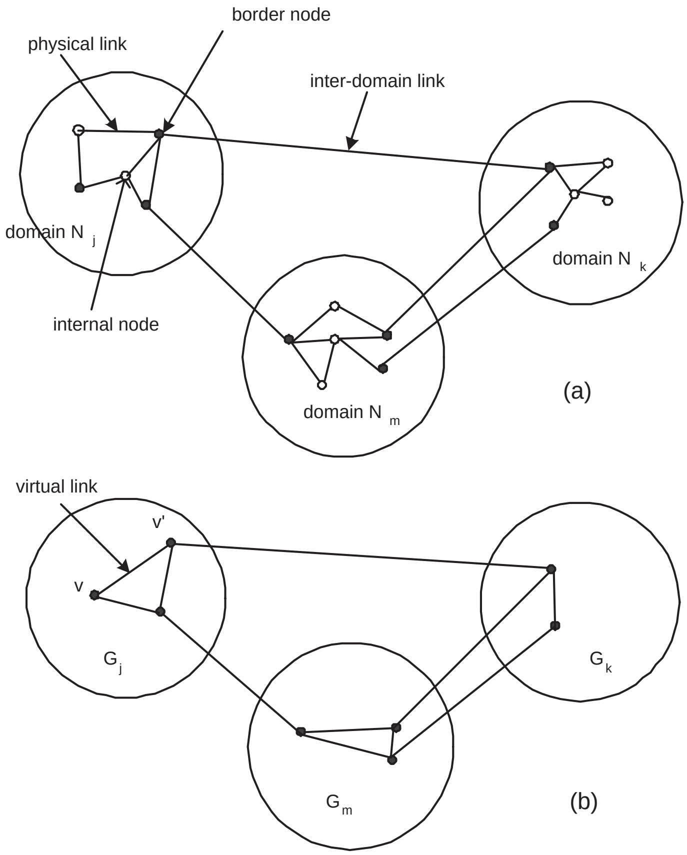

Figure 3: A multi-domain network (a) and its *inter-domain network* (b) obtained from Topology Aggregation.

with the border nodes of other domains through inter-domain links (see Figure 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:

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

$$L = \bigcup_{m=1..M} L_m \bigcup L^{\text{INTER}}.$$

A full mesh topology aggregation (TA) will be applied to each domain network. The TA on domain $\mathcal{N}_m$ results in an aggregate 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 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 Figure 3b), where

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

$$E = \bigcup_{m=1..M} E_m^{\text{VIRTUAL}} \bigcup L^{\text{INTER}}.$$

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 aggregate routing information obtained from its intra-paths. Such aggregate information can be exchanged between border nodes without impairing the *scalability constraint*. The *inter-domain network* is thus a single-domain network.

Let us consider a new connection request with bandwidth d from node v_s to node v_d over a single path. We have to find an end-to-end working path p and a set of backup segments $\{p'_i, i \in I\}$ for it. Let I be the set of segment indexes. The backup segment p'_i protects the working segment p_i. The working path consumes bandwidth d along it without any sharing. Before describing the routing algorithms, we need to introduce additional notations.

2.1 Notations used for the original multi-domain network

c_ℓ^{res} residual capacity on physical link $\ell \in L$.

a_ℓ bandwidth used by p on physical link $\ell \in L$.

$B_{\ell'}$ reserved backup bandwidth on physical link $\ell' \in L$.

$B_{\ell'}^v$ backup bandwidth on physical link $\ell' \in L$ that is already reserved for the backup segments whose working segments go through node $v \in V$. This backup bandwidth cannot be shared with the backup segments $p'_i, i \in I$ that protect v because it violates the *sharing condition*.

$B_{\max}^v = \max\limits_{\ell' \in E} B_{\ell'}^v$ and $B_{\max}^q = \max\limits_{v \in q} B_{\ell'}^v$ are 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 that needs to be reserved on a physical link ℓ' lying on any selected backup segment that protects node v, with respect to the new request.

$b_{\ell'}^q$ additional backup bandwidth that needs to be reserved on a physical link ℓ' lying on any backup segment that protects sub-path q, with respect to the new request.

Note that, $b_{\ell'}^v$ is the difference between the requested bandwidth d and the existing sharable backup bandwidth $(B_{\ell'} - B_{\ell'}^v)$ on ℓ' for protecting a node v. As $b_{\ell'}^v$ must be non negative,

$$b_{\ell'}^v = \max\{0, B_{\ell'}^v + d - B_{\ell'}\}. \tag{1}$$

Readers are referred to [16] and [17] for the detailed and similar computations in case of link protection.

Observe that with OSSP, for a given node, the same backup segments must be activated when whether 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.

Theorem 1 *The backup bandwidth that a backup segment of a connection needs on a link in order to protect a working segment is the largest backup bandwidth needed on the same link to protect a node of the working segment.*

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

2.2 Notations used for the *inter-domain network*

$\pi, \pi_i, \pi'_i (i \in I)$ representations of p, p_i, p'_i in the *inter-domain network* G.

$q \mapsto e$ indicates that the intra-path $q \in \mathcal{P}_e$ is the part of working path p or backup segments $p'_i, i \in I$ that is represented by e in G.

α_e total working bandwidth that p consumes along its sub-path $q \mapsto e \in E$. Thus, $\alpha_e = \sum\limits_{\ell \in q} a_\ell$.

$\beta^e_{e'}$ (resp. $\beta^{\pi_i}_{e'}$) total additional backup bandwidth needed along $q' \mapsto e' \in \pi'_i$ to protect $q \mapsto e \in \pi_i$ (resp. p_i). Thus, $\beta^e_{e'} = \sum\limits_{\ell' \in q'} b^q_{\ell'}$ and $\beta^{\pi_i}_{e'} = \sum\limits_{\ell' \in q'} b^{p_i}_{\ell'}$.

$\overline{B}_{e'} = \begin{cases} B_{\ell'} & \text{if } e' = \ell' \in L^{\text{INTER}} \\ \max\limits_{\ell' \in L_m} B_{\ell'} & \text{if } e' \in E^{\text{VIRTUAL}}_m \end{cases}$. If e is a virtual link this is the maximum backup bandwidth on a physical link of the domain that e comes from. If e is an inter-domain link, this 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 $e \in E$. $\gamma^{\text{res}}_e = \max\limits_{q \in \mathcal{P}_e} \min\limits_{\ell \in p} c^{\text{res}}_\ell$.

$\|e\|$ 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 are called working and backup costs of physical links. Similarly, α and β are called working and backup costs of virtual links (and also inter-domain links).

3 Costs of virtual and physical links

3.1 Costs of virtual links

The exact values of the costs $\alpha_e, \beta^e_{e'}, \beta^{\pi_i}_{e'}$ of a virtual link $e \in E$ depend on the parts of p, $p'_i, i \in I$ that e represents. However, p and $p'_i, i \in I$ are still unknown at this stage. Moreover, the costs are associated with the *inter-domain network* where physical link information is inaccessible. Therefore, we will remove the physical link dependent parameters by approximations and making these costs as functions of virtual link dependent parameters.

The working cost of $e \in E$ is defined as the smallest total bandwidth that working path p should consume along e. Thus:

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

The approximation of the backup cost $\beta^{\pi_i}_{e'}$ is more complex. Let us begin with $b^v_{\ell'}$ which has been defined by:

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

In order to eliminate the dependency of $b^v_{\ell'}$ on detailed information $B^v_{\ell'}$, $b^v_{\ell'}$ is overestimated by: $\max\{0, B^v_{\max}+d-B_{\ell'}\}$. Remind that $b^v_{\ell'}$ cannot be greater than the required bandwidth. We get the following overestimation:

$$b^v_{\ell'} = \min\{\max\{0, B^v_{\max} + d - B_{\ell'}\}, d\}. \tag{5}$$

From this, it can be proved that the backup cost of a virtual or inter-domain link for protecting a working segment is not smaller than the cost for protecting a virtual/inter-domain link of the segment:

$$\beta^{\pi_i}_{e'} = \max_{e \in \pi_i} \beta^e_{e'}. \tag{6}$$

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':

$$\beta^e_{e'} \geq \min_{q \in \mathcal{P}_e, q' \in \mathcal{P}_{e'}} \sum_{\ell' \in q'} b^q_{\ell'}, \tag{7}$$

where

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

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

Thus:

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

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

$$\beta^e_{e'} \geq \min_{q\in\mathcal{P}_e} \|e'\| \times \min\{\max\{0, B^q_{\max} + d - \overline{B}_{e'}\}, d\}. \tag{8}$$

Let v_1, v_2 be two 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:

$$\beta^e_{e'} \geq \|e'\| \times \min\{\max\{0, B^e_{\max} + d - \overline{B}_{e'}\}, d\}. \tag{9}$$

Let us underestimate $\beta^e_{e'}$ by the right-hand side of (9) which is in fact the lower bound of the backup bandwidth that should be reserved along e' for p'. Taking into account the link capacity we define:

$$\beta^e_{e'} = \begin{cases} 0 & \text{if } B^e_{\max} + d \leq \overline{B}_{e'} \\ \|e'\| \times (B^e_{\max} + d - \overline{B}_{e'}) & \text{if } B^e_{\max} + d > \overline{B}_{e'} > B^e_{\max} \\ & \quad \text{and } \gamma^{\text{res}}_{e'} \geq B^e_{\max} + d - \overline{B}_{e'} \\ \|e'\| \times d & \text{if } B^e_{\max} \geq \overline{B}_{e'} \text{ and } \gamma^{\text{res}}_{e'} \geq d \\ \infty & \text{otherwise.} \end{cases} \tag{10}$$

In summary, the working and backup costs of a virtual or inter-domain link are represented as 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 diffuse among themselves these link-states in order to get a common view of the compact *inter-domain network*.

3.2 Costs of physical links

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

$$a_\ell = \begin{cases} d & \text{if } d \leq c^{\text{res}}_\ell \\ \infty & \text{otherwise.} \end{cases} \tag{11}$$

From (5) 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.:

$$b^{p_i}_{\ell'} = \begin{cases} 0 & \text{if } B^{p_i}_{\max} + d - B_{\ell'} \leq 0 \\ B^{p_i}_{\max} + d - B_{\ell'} & \text{if } B^{p_i}_{\max} + d > B_{\ell'} > B^{p_i}_{\max}, \\ & \quad c^{\text{res}}_{\ell'} \geq B^{p_i}_{\max} + d - B_{\ell'} \\ d & \text{if } B^{p_i}_{\max} \geq B_{\ell'}, c^{\text{res}}_{\ell'} \geq d \\ \infty & \text{otherwise.} \end{cases} \tag{12}$$

4 Routing solutions

4.1 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 a request. It can be expressed as follows:

$$\min \sum_{\ell \in p} a_\ell + \sum_{p'_i, i \in I} \sum_{\ell' \in p'_i} b^{p_i}_{\ell'}. \tag{13}$$

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

$$\min \sum_{e \in \pi} \alpha_e + \sum_{\pi'_i, i \in I} \sum_{e' \in \pi'_i} \beta^{\pi_i}_{e'}. \tag{14}$$

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 the 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 (14) 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. The result gives us π_i and $\pi'_i, i \in I$ as paths of virtual/inter-domain links and the *intra-domain step* will follow. If no solution is found, the routing fails.
  In fact, (14) is an OSSP single-domain routing problem. All OSSP single-domain routing solutions cited in this paper can be used to solve it as long as they are applied on the *inter-domain network* and the segment length constraints are integrated. Two solution schemes, GROS and DYPOS, are proposed in the next two paragraphs 4.2, 4.3.
- *Intra-domain step*: The pairs $(\pi_i, \pi'_i), i \in I$ are subsequently considered. For each pair, the virtual links of the working segment are mapped first to the least working cost intra-path:

  $$\min_{q \in \mathcal{P}_e} \sum_{\ell \in q} a_\ell (= \alpha_e). \tag{15}$$

  The selected intra-path for the virtual link e is indeed the Shortest Path (SP) in terms of the physical working cost 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 $b^{p_i}_{\ell'}$:

  $$\min_{q' \in \mathcal{P}_{e'}} \sum_{\ell' \in q'} b^{p_i}_{\ell'} (= \beta^{\pi_i}_{e'}). \tag{16}$$

Note that the nodes along p_i are excluded in this mapping in order to guarantee the disjointedness between each working and backup segment pair.

Each mapping relates to only one domain and can be solved using Dijsktra SP algorithm within the domain while respecting the scalability constraint.

4.2 GROS: A greedy solution

The first routing solution for the *inter-domain step* is a greedy heuristic denoted by GROS (GReedy Overlapped Short segment shared protection). For each new 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 the working cost α_e.
2. The working path is greedily divided into segments. A segment π_i begins from a head node which is the source node for the first segment. The segment tail node is chosen so that the segment is the longest possible with a total estimated length that does not exceed l^{W}. If no such tail node is found, the shortest segment will be taken. From the tail node, we go back 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 cost $\beta_{e'}^{\pi_i}$ between 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) [18] is used for computing each backup segment.

In the GROS heuristic, we do not strictly require that the working segment length must be smaller or equal to l^{W}. In other words the constraint is soft.

If the algorithm does not find a solution at a given step, the routing fails.

GROS defers from CDR in [9]. In CDR, a set of segment end nodes are predefined for each pair of source and destination before the working path identification. 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*.

4.3 Dynamic Programming solution

The second routing solution for the *inter-domain step* is inspired from PROMISE Dynamic programing solution (PRO-D) [19] for single-domain networks. The difference is the integration of the working and backup segment length constraints. Our solution is denoted by DYPOS (DYnamic Programming Overlapped Short segment shared protection).

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.

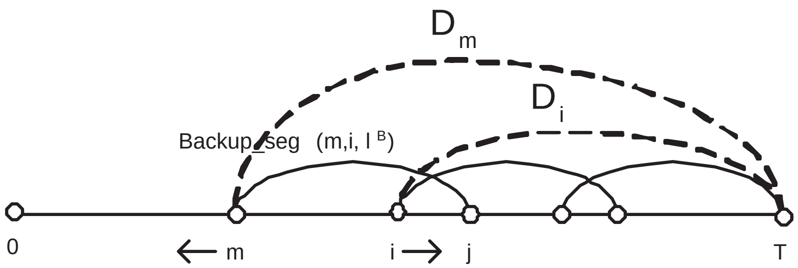

Figure 4: Working mechanism of the Dynamic programming algorithm

Assume that the nodes along the working path are numbering from 0 to T. Let $i \rightarrow j$ denotes the working segment from node i to node j. Let D_m be the "best known" solution to protect the part of working path from node m to node T exclusively. D_m divides possibly that part of working path into multiple overlapped segments and protects each of them by one segment. The current D_m is compared with each alternative 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 $(m = 0)$ (Figure 4).

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

The pseudo-code in Alg.1 describes formally DYPOS. Function `CSP`$^{\mathrm{B}}(m, T, l^{\mathrm{B}})$ implements 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 the working part from $m \rightarrow i$. `Backup_seg` (m, i, l^{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 $i \rightarrow T$. `Backup_seg` (m, i, l^{B}) identifies N least cost segment candidates from m to j with $j = [i+1..i+N]$ using `CSP`$^{\mathrm{B}}(m, j, l^{\mathrm{B}})$ and returns the least cost one.

Different to GROS, here the segment length constraints are hard constraints. If DYPOS finds no solution, it reports a failed routing.

4.4 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 impossibility of mapping a virtual link of a backup segment while maintaining the disjointedness with its

Algorithm 1 DYPOS

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

working segment. Let call the virtual link where the blocking occurs: the blocking virtual link. In order to avoid such blocking cases, a second routing is added to GROS and DYPOS. The second routing 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 if the virtual link was on the working path or backup segments. This removal helps to overcome the previous blocking. Then the intra-domain step, as described in 4.1, is applied again. A failed routing is reported if a new blocking is produced.

5 Signaling and routing information update

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

5.1 Signaling for working and backup segment computation

The *inter-domain step* is performed centrally at the border source node without impairing the *scalability constraint* since the *inter-domain network* is considered as a single-domain network. First of all, 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 $\|e\|$, $\gamma_e^{\rm res}$, $\overline{B}_{e'}$, $B_{\max}^e$ which are available at each border node thanks to the routing information update process that will be described later. Then GROS or DYPOS could be used for performing the inter-domain step. Once the computation is finished, the border source node asks 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, the 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 mapping problems (15) and (16). 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.

5.2 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 border 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 of 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 to reserve an additional amount of bandwidth $b_{\ell'}^v$ on the outgoing link of the backup segment. Note that here, no switch is made. The process terminates when the destination node is reached.

5.3 Routing information update

After each routing, link-states of virtual links change. They should be updated for serving the *inter-domain step* of the next routing. Link-states $\|e\|$, $\gamma_e^{\rm res}$, $\overline{B}_{e'}$, $B_{\max}^e$ are computed locally in the domain containing e by a border node of e. This node writes all these link-states in one message and sends it to 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 needs also the detailed routing information of its domain. A domain scope routing information exchange between domain nodes is also needed.

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

For reducing furthermore the charge of update message flow, 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.

6 Experimental results

We use different network and traffic instances to evaluate the efficiency of GROS and DYPOS through the backup overhead and overall blocking probability metrics that we next introduce.

6.1 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 less 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 from the total bandwidth requested by all connections.

6.2 Comparison with optimal single-domain solution

We evaluate the efficiency of GROS and DYPOS by comparing their results on a multi-domain network with the result of the single-domain optimal solution [9], 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 dynamic requests. The Transit-Stub model of GT-ITM [20], a well known multi-domain network generator, is used for generating this network instance that we denote by SMALL-5 and represent in Figure 5. GROS and DYPOS take milliseconds to route a request. Due to the small scale of the network, the constraint on backup segment length is ignored by setting l^{B} very large for GROS and DYPOS. In Opt, neither working and

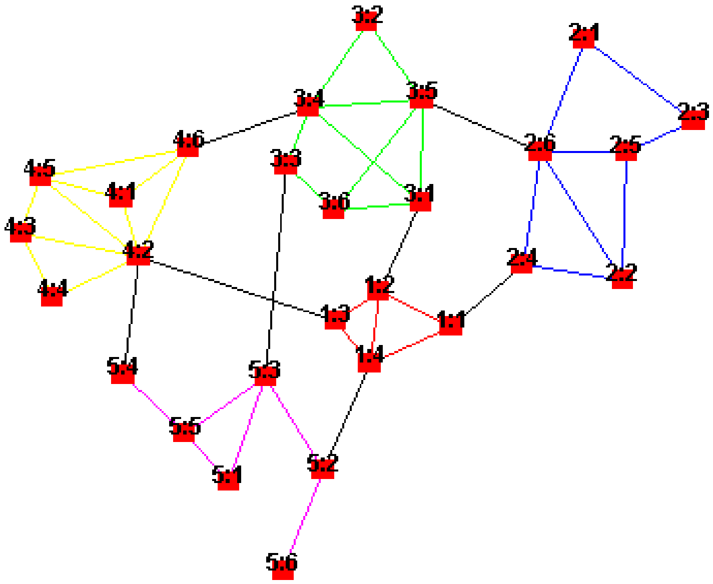

Figure 5: SMALL-5 network.

backup segment lengths are restricted. We made also comparison with the results obtained from dedicated protection denoted by NoShare.

Figure 6 shows that the proposed two-step solution with either GROS or DYPOS provides the backup overhead close to Opt and far better than NoShare. In other words, GROS and DYPOS yield a very good bandwidth saving rate. Do not forget that the constraint on l^{W} is present in GROS and DYPOS, while it is absent in Opt, therefore giving a slightly 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 take solution with larger cost than that of GROS if the later violates the constraint on l^{W}. This phenomena reduces when l^{W} increases.

6.3 Backup overhead

From now on, the experiments are made on large multi-domain networks with heuristics only. The Transit-Stub model of GT-ITM, is again used for generating one 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 Figure 7. Each domain has in average 4 neighboring domains. According to [21], this number reflects faithfully the Internet interconnection.

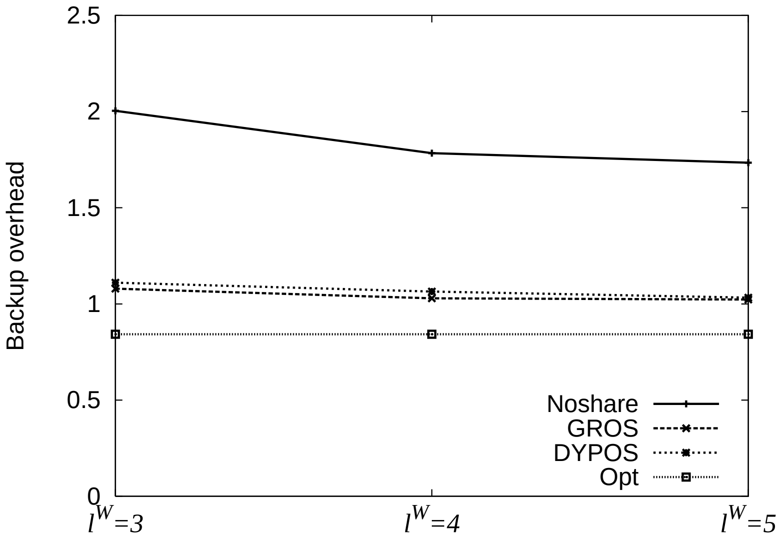

Figure 6: Backup overhead in SMALL-5.

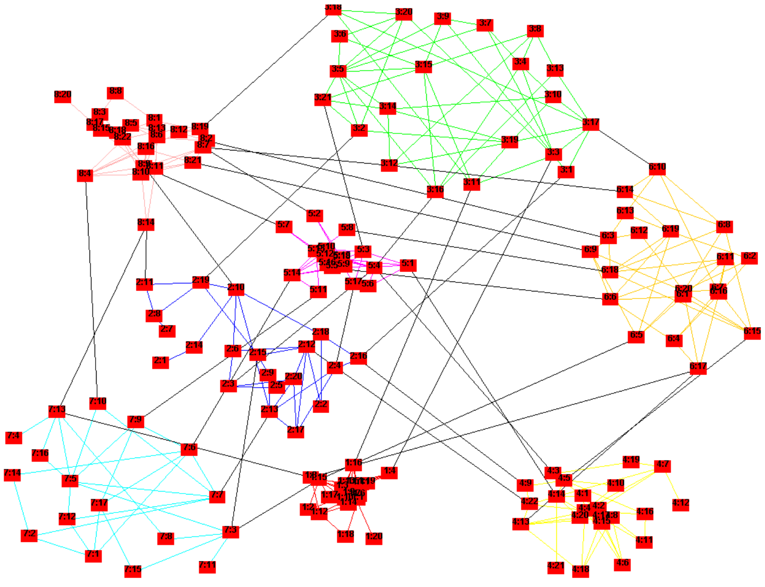

Figure 7: LARGE-8 network.

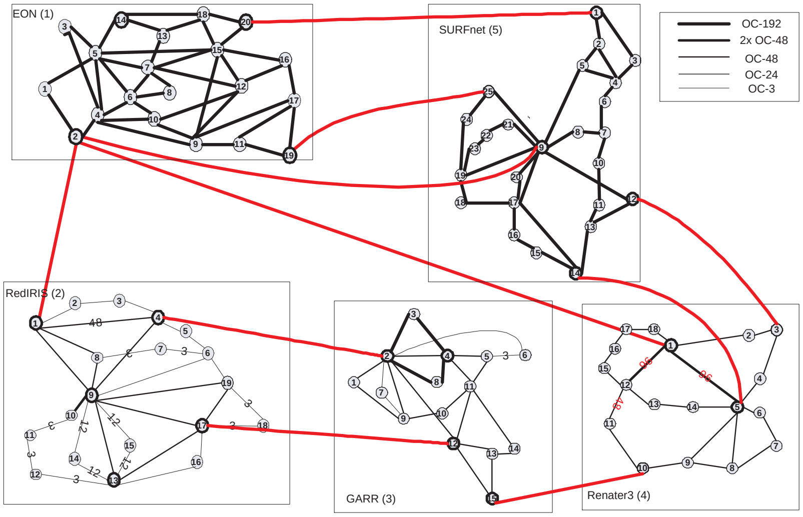

Figure 8: LARGE-5 network.

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 [22] for the details of the topology.

We also consider another multi-domain network that we used for experiments in previous papers [23–25]. The network is built from 5 real optical networks: EON [26], [27], [28], [29], [30]. Inter-domain links are added with capacity OC-192. The network is denoted by LARGE-5 and is shown in Figure 8.

An incremental traffic is generated by submitting subsequently 1000 connection requests to the network, all requests remain active in the network. The incremental traffic allows keeping active more requests and thus allow evaluating more accuratly the bandwidth allocation characteristics of each solution scheme. Network links are uncapacitated in order to avoid the impact of blocking which is different from one scheme to the other. Backup overhead is computed after 1000 requests.

Figure 9 depicts backup overhead of GROS, DYPOS in comparison with NoShare in LARGE-8 when working segment length thresholds are $l^{\mathrm{W}} = 3$ and $l^{\mathrm{W}} = 5$ and backup segment thresholds vary. Similar backup overheads are found in GROS and DYPOS. We notice also that GROS and DYPOS require only 0.55 and 0.8 times 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 (Figure 10), we find a smaller but still significant difference between the backup over heads of NoShare and of 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.

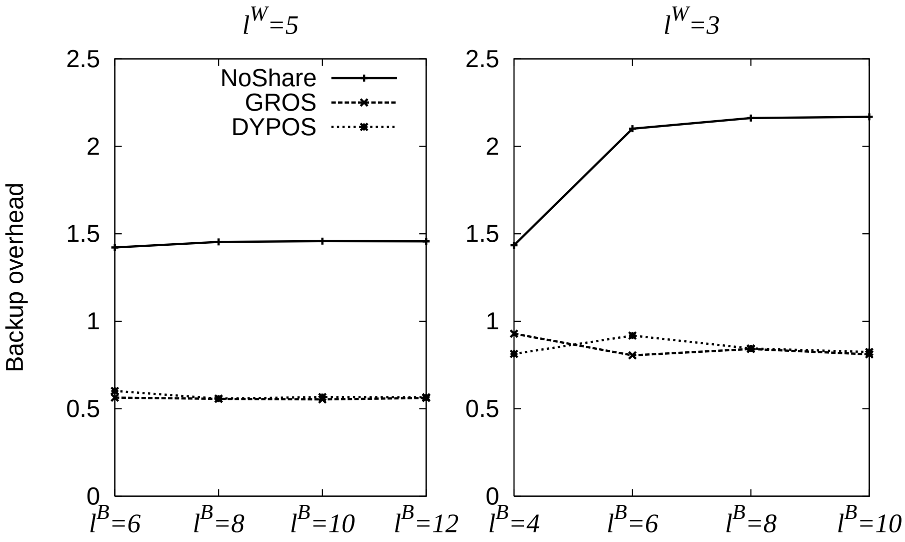

Figure 9: Backup overhead in LARGE-8.

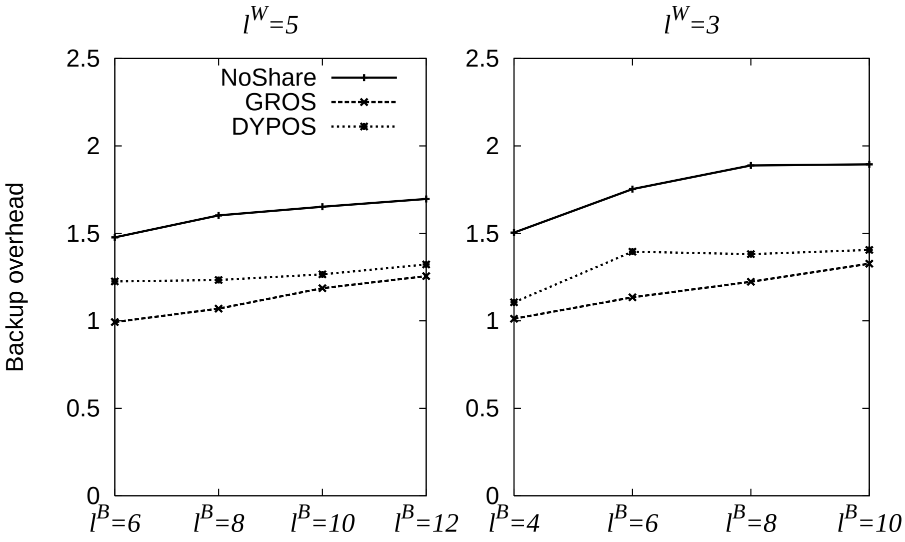

Figure 10: Backup overhead in LARGE-5.

6.4 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 Poisson process with rate $r = 1$ and their holding time is exponentially distributed with mean $h = 320$. There are, on average, 320 active connections in the network.

Figure 11 depicts the overall blocking probability of GROS, GROS with the Blocking-go-back option (denoted by GROS-BGB), DYPOS and 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 Figure 12), a similar phenomenon is again observed with $l^{\mathrm{W}} = 5$. However, DYPOS and DYPOS-BGB become worse than GROS and sometimes even than NoShare. This is explained by two reasons. 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 possibility of dividing working paths into segments of 3 hops or less. This reveals the pertinence of properly defining 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. Figure 13 and 14 show more clearly the advantage of the Blocking-go-back step. The curves GROS Inter and DYPOS Inter depict the percentages of the requests that are successfully routed in the *inter-domain step* of the second routing. Similarly, the curves GROS Intra and DYPOS Intra depict the percentages of the requests that are successfully routed after 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 for the case of too small thresholds $l^{\mathrm{W}} = 3, l^{\mathrm{B}} = 4$. We can conclude that the second routing is useful to increase the grade of service.

6.5 Impact of segment length

Table 1, give us some ideas about the distribution of routed requests according to the number of segments. Given the segment length thresholds and the network topology, we obtained cases with up to 3 segments. Although short segment length promises fast recovery, it sometimes impairs backup overhead. When the segment length thresholds are

Table 1: Distribution of number of segments.

Nb. of segment	1 segment	2 segments	3 segments
8Dom, $l^{\mathrm{W}} = 3$	47-95 %	5-42%	1-15%
8Dom, $l^{\mathrm{W}} = 5$	82-92 %	8-16%	0-2%
5Dom, $l^{\mathrm{W}} = 3$	60-93 %	6-31%	0-13%
5Dom, $l^{\mathrm{W}} = 5$	71-97 %	3-23%	0-8%

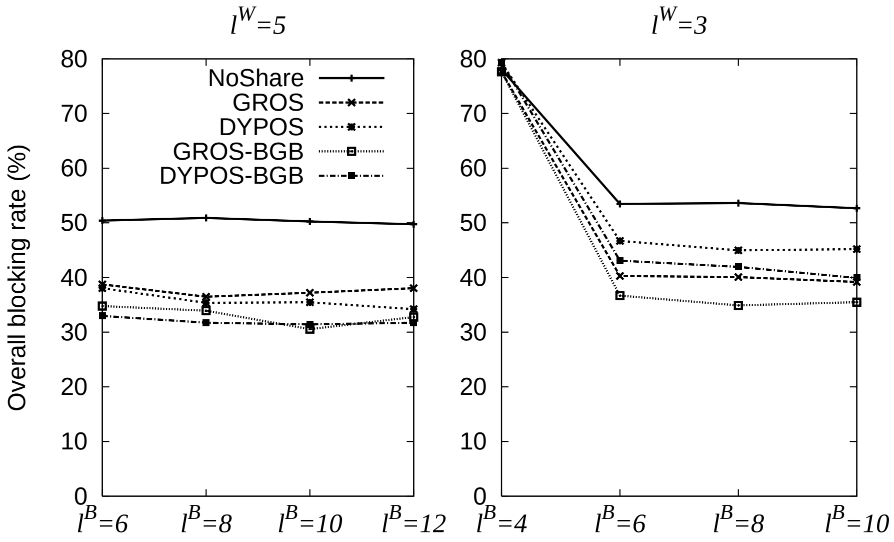

Figure 11: Overall blocking probabilities in LARGE-8.

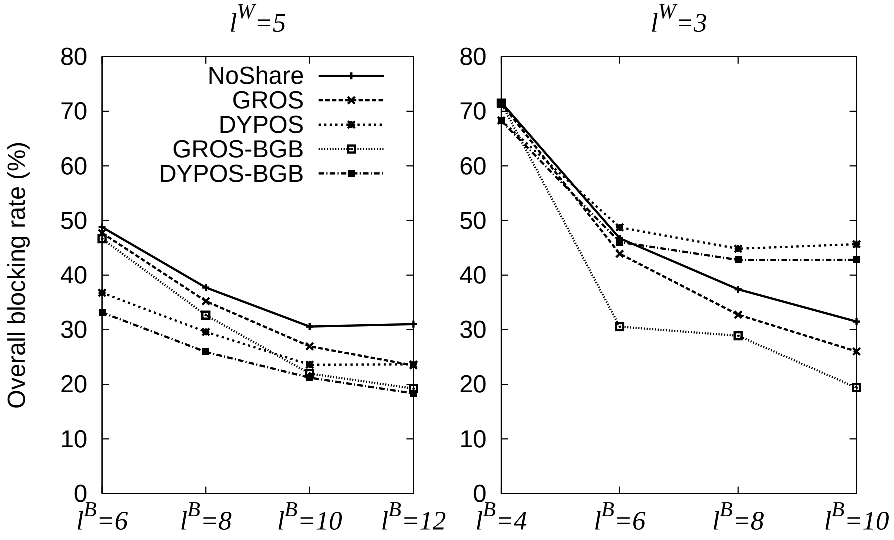

Figure 12: Overall blocking probabilities in LARGE-5.

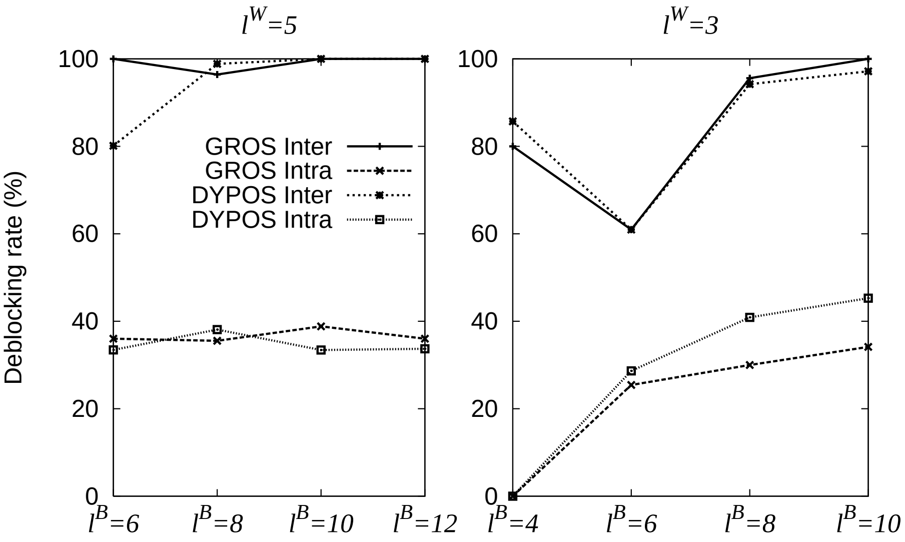

Figure 13: De-blocking capacity of the Blocking-go-back step in LARGE-8.

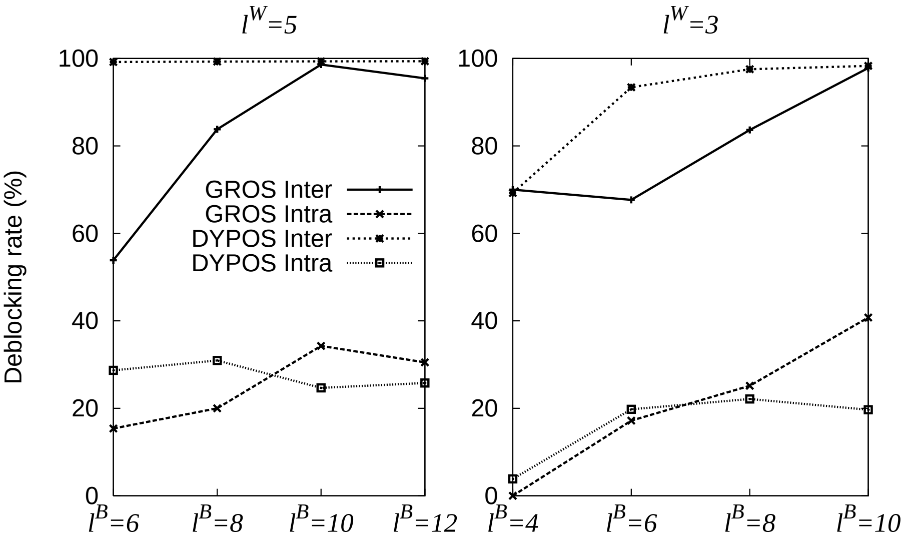

Figure 14: De-blocking capacity of the Blocking-go-back step in LARGE-5.

too small, there are few choices for working path division and backup segment building. This leads to the selection of the solution that has high backup cost but satisfies the segment length constraints. As a result the overall backup overhead increases. Indeed, in LARGE-8 as shown in Figure 9, backup overhead increases from around 0.55 when $l^{\mathrm{W}} = 5$ to around 0.8 when l^{W} reduces to 3. A smaller increment is also found with LARGE-5 in Figure 10.

Again, too small segment length thresholds make worsen the blocking probability. There might be no solution satisfying the required working and back up segment lengths. This is illustrated in Figure 11 and 12. 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 LARGE-5, at $l^{\mathrm{W}} = 3$, the blocking probabilities raise up drastically when the backup segment length threshold reduces to $l^{\mathrm{B}} = 4$. Smaller impact is observed with $l^{\mathrm{W}} = 5$ because the thresholds are nevertheless large enough to provide a reasonable number of segment choices.

7 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 the backup bandwidth sharing. They also show the advantage of the Blocking-go-back phase in reducing the blocking probability.

The proposed solutions guarantee fast recovery because the working and backup segments are restricted in length. Obviously, the smaller the segment lengths are, the shorter the recovery is. However, the experiment results show that segment length thresholds should be considered carefully because too small thresholds may entail in significant increment of blocking probability as well as backup overhead.

References

[1] 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.

[2] P.-H. Ho and H. T. Mouftah, “A framework for service-guaranteed shared protection in WDM mesh networks,” *IEEE Communications Magazine*, vol. 40, no. 2, pp. 97–103, 2002.

[3] S. Ramamurty and B. Mukherjee, “Survivable WDM Mesh Networks, Part I Protection,” in *Proc. IEEE INFOCOM*, vol. 2, New York, NY, March 1999, pp. 744–751.

[4] W. D. Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration: ring-like speed with mesh-like capacity for self-plaining network restoration," in *Proc. IEEE ICC*, 1998, pp. 537–543.

[5] 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.

[6] ——, "Segment-based approaches to survivable translucent network design under various ultra-long-haul system reach capabilities," *OSA Journal of Optical Networking*, vol. 3, no. 1, pp. 1–24, Jan. 2004.

[7] G. Bernstein, V. Sharma, and L. Ong, "Interdomain Optical routing," *OSA Journal of Optical Networking*, vol. 1, no. 2, pp. 80–92, Feb. 2002.

[8] J. L. Roux, J. 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., Jun. 2004.

[9] P.-H. Ho, J. Tapolcai, and T. Cinkler, "Segment shared protection in mesh communications networks with bandwidth guaranteed tunnels," *IEEE/ACM Transactions on Networking*, vol. 12, no. 6, pp. 1105–1118, 2004.

[10] P.-H. Ho and H. Mouftah, "Spare capacity allocation for WDM mesh networks with partial wavelength conversion capacity," in *Workshop on High Performance Switching and Routing*, 2003, pp. 194–199.

[11] D. Xu and et al., "Protection with Multi-Segments (PROMISE) in Networks with Shared Risk Link Groups (SRLG)," in *Proc. The 40th Annual Allerton Conference on Communication*, 2002.

[12] 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, France, Feb. 2001, p. ThO6.

[13] 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*, Dallas, Texas, Sep. 2002.

[14] 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*, Kobe, Japan, Apr. 2004.

[15] B. Mukherjee, *Optical WDM Networks.* Springer, 2006.

[16] M. Kodialam and T. Lakshman, "Dynamic Routing of Bandwidth Guaranteed Tunnels with Restoration," *IEEE INFOCOM*, pp. 902–911, 2000.

[17] D. Xu, C. Chunming, 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*, France, Nov. 2002, pp. 344–353.

[18] 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.

[19] D. Xu, Y. Xiong, and C. Qiao, "Novel Algorithms for Shared Segment Protection," *IEEE/Journal on Selected Areas in Communications*, vol. 21, no. 8, pp. 1320–1331, Oct. 2003.

[20] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to Model an Internetwork," in *IEEE Infocom*, vol. 2. San Francisco, CA: IEEE, March 1996, pp. 594–602.

[21] D. Magoni and J. J. Pansiot, "Analysis of the autonomous system network topology," *SIGCOMM Computer Communication Review*, vol. 31, no. 3, pp. 26–37, 2001.

[22] "LARGE-8 network," http://www.iro.umontreal.ca/∼laborc.

[23] D. L. Truong and B. Thiongane, "Dynamic routing for Shared Path Protection in Multidomain optical mesh networks," *OSA Journal of Optical Networking*, vol. 5, no. 1, pp. 58–74, Jan. 2006.

[24] B. Jaumard and D. L. Truong, "Backup Path Re-optimizations for Shared Path Protection in Multi-domain Networks," in *Proc. IEEE Globecom 2006 (to appear)*, Nov. 2006.

[25] D. L. Truong and B. Jaumard, "Overlapped Segment Shared Protection in Multidomain Networks," in *Proc. APOC*, Gwangju, Korea, Sept 2006, pp. 594–602.

[26] M. O'Mahony, D. Simeonidu, A. Yu, and J. Zhou, "The Design of the European Optical Network," *Journal of Ligthwave Technology*, vol. 13, no. 5, pp. 817–828, 1995.

[27] "REDIrid," 2005, http://www.rediris.es/red/index.en.html#red%20troncal.

[28] "Consortium GARR," http://www.garr.it.

[29] "RENATER-4 network," http://www.renater.fr.

[30] "Surfnet," http://www.surfnet.nl.