A Map-and-Route Approach for Segment Shared Protection in Multi-domain Networks

D. L. Truong
Computer Science and Operations Research
Université de Montréal, Canada
Email: linh@crt.umontreal.ca

B. Jaumard
Concordia Institute for Information Systems Engineering
Concordia University, Canada
Email: bjaumard@ciise.concordia.ca

***Abstract*— Routing and protection with an** Overlapping Segment Shared Protection **(OSSP) scheme in multi-domain networks is more difficult than that in single domain networks because of scalability requirements. We propose a novel approach for OSSP routing where the underlying idea is the prior identification of Potential Intra-domain Paths (PIP) for carrying working and backup traffic between domain border nodes. These PIPs help to reduce the multi-domain network to a simpler aggregated network where routing is performed without unnecessarily going down to the physical links. The novel approach offers an exact and highly scalable routing thanks to the prior identification of the PIPs and the introduction of a maximal share risk group feature. Experiments show that the quality of the proposed approach is close to the optimal single-domain network solution and outperforms the existing multi-domain network solutions.**

I. Introduction

Much research has been conducted on path and segment protection for single-domain networks, much less for multi-domain networks. Overlapping Segment Shared Protection (OSSP) is a recently proposed protection scheme, which has not yet received a lot of interest, in particular, in the context of multi-domain networks. This paper tackles the routing problem for OSSP in multi-domain networks. Before describing the problem, we briefly recall the OSSP concept.

A. OSSP concept

In classical segment protection, an end-to-end working path is divided into concatenated segments and each one is protected by a unique backup segment. Upon a single link or node failure, only the failed working segment is replaced by its backup segment. As a result, segment protection offers a faster recovery than path protection. However, segment end nodes are not protected as the failures of those nodes impair both the working and backup segments. Overlapping Segment Protection (see, e.g., [6]) overcomes this weakness thanks to the overlapping between working segments (see Fig. 1) while still inheriting the fast recovery of segment protection.

For backup bandwidth saving, shared protection has been proposed for link, path and segment protection. In segment protection, in order to guarantee 100% recovery of any single link or node failure, two backup segments can share bandwidth if and only if their working segments are link and node-disjoint. This condition is called *segment sharing condition*, see Fig. 2 for an illustration. In case (a), the working segment

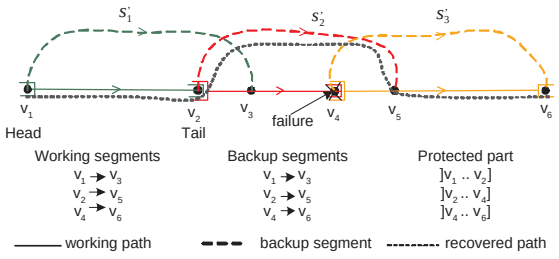

Fig. 1. Example of Overlapping Segment Protection when v_4 fails. The protected part $]v_2..v_4]$ contains all links and nodes between v_2 exclusively and v_4 inclusively, thus v_4 is recovered using segment s'_2.

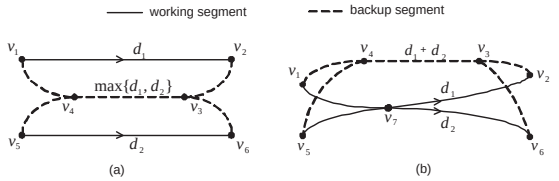

Fig. 2. Examples of cases where two backup segments can share backup bandwidth (a) and cannot (b).

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. Their backup segments can share bandwidth over the common link (v_4, v_3) and the needed bandwidth on this link is $\max\{d_1, d_2\}$ in order to ensure protection for both working paths. In case (b), the two working segments share node v_7, their backup segments cannot share backup bandwidth. The needed backup bandwidth on link (v_4, v_3) is $d_1 + d_2$ which is greater than in case (a).

With the addition of the backup bandwidth sharing feature, Overlapping Segment Protection becomes Overlapping Segment Shared Protection (OSSP). For a new incoming request, the dynamic OSSP routing problem consists of establishing a working path and associated backup segments, while minimizing the total used bandwidth. The routing is then done without making any forecast assumption on the upcoming requests. The amount of backup bandwidth to reserve for a backup segment depends on the working segment to be protected as well as on the already established working and backup segments. This dependency makes the problem quite

complex. Indeed, it is even more combinatorial in practice than the Shared Path Protection problem which is already NP-hard. Several solutions have been proposed in [2], [5], [6], [11]. They require detailed information on the bandwidth allocation on each network link and compute accurately the bandwidth costs. Such a requirement can be satisfied only in single domain networks. Therefore, we qualify all these solutions under single-domain solutions.

B. State of the art of OSSP in multi-domain networks

OSSP for multi-domain networks is much more complex than that for single domain networks due to the multi-domain network characteristics and size. A multi-domain network is made of the interconnection of several single-domain networks, see Fig. 3a. In order to satisfy the *scalability requirements*, only the aggregated routing information can be exchanged amongst domains [7]. Consequently, a given node is neither aware of the global multi-domain network topology nor of the detailed bandwidth allocation on each network link.

Few solutions have been proposed for multi-domain networks. The studies in [1], [4] propose to protect each domain individually. Consequently, inter-domain links and border nodes are left either unprotected or protected by a specific scheme. Other studies such as [3] focus on designing network backup capacity with static traffic.

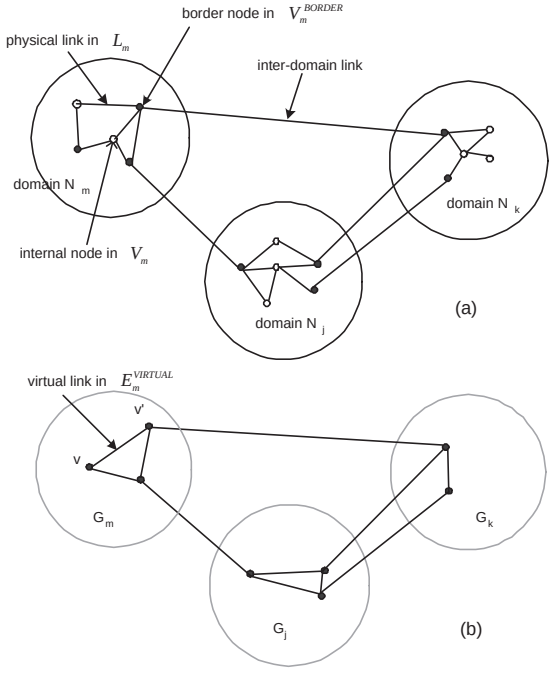

Fig. 3. A multi-domain network (a) and its *inter-domain network* (b).

This paper aims at solving the OSSP routing problem in optical multi-domain networks with the objective of minimizing the total working and backup bandwidth capacity required by a incoming request under a dynamic traffic pattern. All network nodes are assumed to be wavelength conversion capable.

In [10], some solutions for OSSP in multi-domain networks have been proposed where each domain network is topologically aggregated. A domain $N_m = (V_m, L_m)$, where V_m and L_m are the sets of nodes and links, becomes an aggregated graph $G_m = (V_m^{\text{BORDER}}, E_m^{\text{VIRTUAL}})$ composed of a border node set V_m^{BORDER} and a virtual link set E_m^{VIRTUAL} (see Fig. 3b). A virtual link represents the possibility of going from one border node to another one through intra-domain paths (or intra-paths for short). Each virtual link is associated with approximated costs for using its associated intra-paths in a working or a backup segment. Instead of performing routing directly on the multi-domain network, it starts with a rough routing in the inter-domain network. The resulting working and backup segments are the paths of virtual and inter-domain links. These virtual links are then mapped to intra-paths. The authors of [10] proposed several algorithms for the routing and the mapping resulting in two solutions GROS and DYPOS. They are both referred in this paper by "Route-and-Map" approaches, denoted by RaM.

In RaM, the approximation in cost computation is necessary for dealing with the scalability problem. However, the approximation reduces the routing quality. For a better routing quality, we propose in this paper to eliminate the approximation by reversing the mapping and routing steps. The resulting approach is called "Map-and-Route" or MaR for short. Each virtual link is mapped to several intra-paths whose working and backup costs are computed exactly. The routing is performed on a network where links are the selected intra-paths and the link costs are exact.

The paper is organized as follows. The next section provides the general idea of the MaR approach. Section III describes the Mapping problem. Its greedy solution is shown in Section IV. Section V presents the Routing solution and Section VI discusses its scalability. The experimental results are shown in Section VII. Conclusions follow in Section VIII.

II. A MAP-AND-ROUTE APPROACH

In MaR, each domain network N_m is first aggregated as in RaM. Next, for each virtual link e of N_m, a set of Potential intra-Paths (PIPs) $\mathcal{P}_e^{\text{W}}$ (resp. $\mathcal{P}_e^{\text{B}}$ with possibly $\mathcal{P}_e^{\text{B}} = \mathcal{P}_e^{\text{W}}$), is selected for carrying the working (resp. backup) traffic between border nodes of e (see Fig. 4). The border nodes correspond to the filled black points in all figures. The PIPs carrying working (resp. backup) traffic are referred as working (resp. backup) PIPs.

Definition 1: A direct intra-path is an intra-path that does not go through any intermediate border node.

In MaR, all PIPs must be direct intra-paths. This requirement does not reduce the choice of intra-paths as a non direct intra-path can be represented by multiple direct intra-paths.

In addition to the traditional *segment sharing condition*, in order to work at the level of PIP, we introduce the following condition in MaR:

Supplementary Sharing Condition: Two backup segments can be considered for bandwidth sharing if they go through an **identical PIP**.

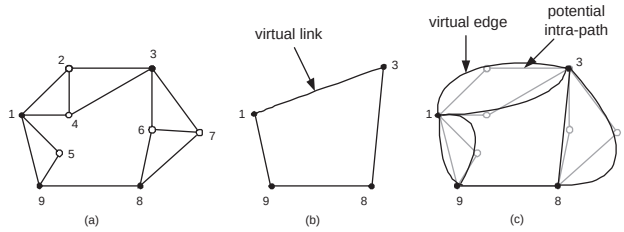

Fig. 4. (a) An original domain at the intra-domain level; (b) the aggregated domain at the inter-domain level; (c) the mapped domain at the mapped level with a maximum of 2 PIPs/virtual link for both working and backup traffic.

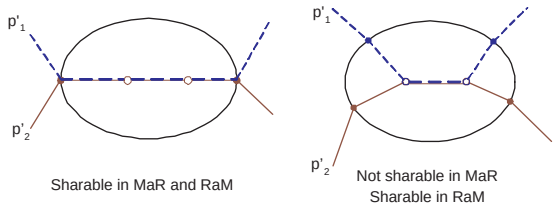

Fig. 5. Cases where two backup segments p'_1, p'_2 can share and cannot share backup bandwidth under MaR and RaM. Their working paths are disjoint in both cases and are not shown in the figure.

Two backup segments must satisfy both conditions in order to be allowed sharing bandwidth. The backup segments over two PIPs that differ by at least one link are not allowed to share bandwidth (Fig. 5). Therefore, *the amount of sharable backup bandwidth is identical for every link along an* PIP. Of course with this additional sharing condition, bandwidth sharing is sometimes ignored on some particular links. However, we do not have to go down to the level of the physical links in order to identify exactly the sharable bandwidth for protecting an PIP, which would impair the scalability. Note that in RaM, in order to avoid going down to the physical link level, approximations are made on sharable bandwidth computations.

The OSSP routing problem is divided into 2 sub-problems:

- Mapping sub-problem: PIP sets $\mathcal{P}_e^{\mathrm{W}}, \mathcal{P}_e^{\mathrm{B}}$ are selected for each virtual link e of each domain. Each PIP is then abstracted as a single link called "virtual edge" (Fig. 4c). The virtual edges between different nodes may not be disjoint. The multi-domain network resulting from this abstraction is called "mapped network". Mapping is performed once for a long term use.
- Routing sub-problem: The working path and backup segments are computed in the mapped network so that they minimize their used bandwidth. Unlike RaM, there is no need to do an intra-domain routing for identifying the intra-path within each domain because a virtual edge is one-to-one correspondence with a fix PIP.

Both sub-problems will be discussed in detail in the next sections.

III. Mapping sub-problem

The Mapping sub-problem consists of identifying a set of working PIPs and a set of backup PIPs for each virtual link. Such a Mapping is performed independently in each domain and is stated as follow for domain N_m.
Given:

- n^{W} and n^{B} the maximum numbers of working and backup PIPs needed for each virtual link of E_m^{VIRTUAL};
- n_e the number of *direct* intra-paths associated with e.

Let $n_e^{\mathrm{W}} = \min(n_e, n^{\mathrm{W}})$ and $n_e^{\mathrm{B}} = \min(n_e, n^{\mathrm{B}})$. They correspond to the exact number of working and backup PIPs for each $e \in E_m^{\mathrm{VIRTUAL}}$. We need to identify:

- $\mathcal{P}_e^{\mathrm{W}} = \{q_{e,i}^{\mathrm{W}}, i = 1..n_e^{\mathrm{W}}\}$, the set of working PIPs of e;
- $\mathcal{P}_e^{\mathrm{B}} = \{q_{e,i}^{\mathrm{B}}, i = 1..n_e^{\mathrm{B}}\}$, the set of backup PIPs of e.

As the Routing objective is to minimize the total working and backup cost of each request, in the Mapping, we encourage the intra-paths supporting this objective through the following selection criteria.

Criterion 1: A selected working PIP should minimize its working cost while maintaining enough residual bandwidth for allocating future connections. It amounts to balance the network load.

We can associate each physical link with a weight which is the inverse of the residual capacity of the link. The set of selected PIPs are then a set of weighted shortest paths:

$$\min \sum_{e \in E_m^{\mathrm{VIRTUAL}}} \sum_{q \in \mathcal{P}_e^{\mathrm{W}}} \sum_{\ell \in q} \frac{1}{c_\ell^{\mathrm{res}}} \tag{1}$$

where c_ℓ^{res} is the residual capacity of physical link ℓ.

Criterion 2: A selected backup PIP should minimize its backup cost while maintaining enough residual bandwidth for allocating future connections.

From a global sight, a backup segment uses an homogeneous amount of bandwidth along an PIP as in a working segment. Criterion 2 is thus modeled similarly to Criterion 1:

$$\min \sum_{e \in E_m^{\mathrm{VIRTUAL}}} \sum_{q \in \mathcal{P}_e^{\mathrm{B}}} \sum_{\ell \in q} \frac{1}{c_\ell^{\mathrm{res}}}. \tag{2}$$

Criterion 3: The working PIPs should be selected so as to increase the possibility of finding pairwise disjoint working PIPs.

This criterion originates from the fact that backup segments can share bandwidth only if their working segments are disjoint, according to the segment sharing condition. The criterion is interpreted as maximizing the number of pairs of disjoint working PIPs:

$$\max \sum_{\substack{q_1 \in \mathcal{P}_{e_1}^{\mathrm{W}}, q_2 \in \mathcal{P}_{e_2}^{\mathrm{W}}, \\ e_1, e_2 \in E_m^{\mathrm{VIRTUAL}}}} \delta_{q_1}^{q_2} \tag{3}$$

where $\delta_{q'}^{q}$ is 1 if q and q' are node disjoint and 0 otherwise.

Criterion 4: Virtual links should be mapped so that the possibility that a pair of working and backup virtual links is disjoint is maximized.

Definition 2: Two virtual links are disjoint iff there exists a PIP of one virtual link that is link and node disjoint with a PIP of the other virtual link.

The disjointness between virtual links is formally stated as:

$$\delta_{e'}^{e} = \begin{cases} 1 & \text{if } \exists q \in \mathcal{P}_e^{\mathrm{W}}, q' \in \mathcal{P}_{e'}^{\mathrm{B}} : q \cap q' = \emptyset, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

Criterion 4 is justified as follows. In the case of lightly interconnected multi-domain networks, we might have to route a request over two fixed virtual links, one for the working segment and the other one for the backup segment. These two virtual links should have at least one pair of disjoint PIPs otherwise the considered working and backup segments would have at least one common link or node, and this would impair the protection. From the global viewpoint, this criterion is interpreted as maximizing the number of pairs of disjoint working and backup virtual links:

$$\max \sum_{e,e' \in E_m^{\mathrm{VIRTUAL}}} \delta_{e'}^{e}. \quad (5)$$

A. Putting all together

With the above system of criteria, the Mapping is a multi criteria optimization problem. For solving it, we put the criteria all together in a single objective which is a weighted sum of the four criteria. The weights are denoted by μ_1, μ_2, $-\mu_3$, $-\mu_4$ respectively where $\mu_1, \mu_2, \mu_3, \mu_4 \geq 0$. The negative sign in the last two coefficients results from the maximization objectives (3) and (5) of criteria 3 and 4. The four coefficients should be considered carefully in order to prioritize some objectives. Large μ_1 and μ_2 prioritize the bandwidth saving.

Since working and backup PIPs are symmetrical in the Mapping model, we can simply use a unique set $\mathcal{P}_e^{\mathrm{WB}} = \mathcal{P}_e^{\mathrm{W}}$ of $n_e^{\mathrm{WB}} = n_e^{\mathrm{W}}$ PIPs for both working and backup traffic over virtual link e. The constraint and objective term related to backup PIPs, which is weighted by μ_2, are removed. We propose MaR-O an exact solution based on Integer Linear Programming (see [9]) and MaR-G an heuristic solution (the next section), for solving this Mapping sub-problem. From time to time, the Mapping should be refreshed for obtaining new PIPs which are more appropriate with the actual network residual capacity.

IV. Heuristic Mapping solution

This section presents MaR-G, the heuristic for solving the Mapping sub-problem. The main idea is as follows. For selecting PIPs for each virtual link e, we do not consider all possible intra-paths but only a subset $\mathcal{P}_e^{\mathrm{CAN}} \subset \mathcal{P}_e$ of n_e^{CAN} intra-path candidates. Due to Criterion 1, $\mathcal{P}_e^{\mathrm{CAN}}$ will be the set of shortest intra-paths weighted by their residual capacities.

Alg. 1 presents pseudo-code of the Mapping for each domain. For a domain, the list of virtual links of the domain under study is browsed. For each virtual link, we try to find several intra-paths that increase the most the number of disjoint virtual link pairs. Amongst these intra-paths, we select the one that is disjoint with the largest number of PIPs that have been selected for the domain. The next virtual link will be considered in the same way. Once all virtual links are visited, another round is started again and again until each virtual link receives the required number of PIPs.

Algorithm 1 Greedy_maping(N_m)

```
for all e ∈ E_m^VIRTUAL do
  P_e^CAN = set of n^CAN shortest intra-paths weighted by residual capacity.
end for
while ∃e ∈ E_m^VIRTUAL so that |P_e^WB| < n_e^WB do
  for all e ∈ E_m^VIRTUAL do
    if |P_e^WB| < n_e^WB then
      {—-P_e^WB is not full, select an intra-path for e—-}
      for all q ∈ P_e^CAN do
        dj_q ← Number of backup virtual links that is newly disjoint with e thanks to q
      end for
      S_e ← Set of n intra-paths that has the highest dj_q
      q ← The intra-path in S_e that is disjoint with the most working PIPs in ⋃_{e1 ∈ E_m^VIRTUAL} P_{e1}^WB
      P_e^WB = P_e^WB ∪ {q}
      P_e^CAN = P_e^CAN \ {q}
    end if
  end for
end while
```

V. Routing sub-problem

The objective of the Routing sub-problem is: for a new incoming request, find a working path and a set of backup segments so that their total bandwidth cost is minimized. Let d be the requested bandwidth. Before detailing the analytical expression of the total bandwidth cost of the incoming request, we introduce some further notations and define cost functions of virtual edges. Since a virtual edge is in one-to-one correspondence with an PIP, we use the two terms alternatively depending on whether we are dealing with the mapped or the detailed (intra-domain) level.

Let q (resp. q') be an PIP/virtual edge that is considered for a working segment (resp. backup segment) for the new incoming request. Let us assume that each bandwidth unit on a physical link has a unit cost and the bandwidth cost of a segment is the sum of the bandwidth costs of its links.

- $B_{q'}$ backup bandwidth reserved by backup segments going through the entire virtual edge q'. Be aware that $B_{q'}$ may differ from the total backup bandwidth reserved on a physical link of q'.
- $B_{q'}^{v}$ sum of requested bandwidth for the connections of which a backup segment goes through q' and the corresponding working segment goes through node v. Those backup segments cannot share bandwidth amongst them because they will be activated simultaneously when v fails. Their backup bandwidth is not profitable for a backup segment of the new incoming request if this segment goes through q' and protects a working segment going through v.
- $B_{q'}^{q}$ sum of requested bandwidth for the connections of which a backup segment goes through q' and the corresponding working segment goes through q.

α_q total working bandwidth cost of the new incoming request on virtual edge q.

$\beta^q_{q'}$ backup bandwidth cost of the new incoming request on virtual edge q' for protecting virtual edge q against any single link or node failure. Note that $\beta^q_{q'}$ is the additional bandwidth that the new incoming request needs on q' excluding the fraction of sharable backup bandwidth it can benefit from $B_{q'}$.

π working path of the new incoming request in the mapped network. It is a path made of virtual edges.

π_i working segment of the new incoming request indexed by i in the mapped network. It is a path made of virtual edges.

π'_i backup segment of the working segment π_i in the mapped network. It is a path made of virtual edges.

I set of segment indexes of the new incoming request.

$\beta^{\pi_i}_{q'}$ backup bandwidth cost of the new incoming request on virtual edge q' for protecting working segment π_i against a single failure on any node or link.

$\|q\|$ length of PIP q in terms of the number of hops.

γ_q residual capacity of virtual edge q.

Definition 3: The residual capacity of a virtual edge is the maximum bandwidth that we can route along it.

The bandwidth we can route over q is limited by the smallest residual capacity of the physical links of q, which leads to:

$$\gamma_q = \min_{\ell \in q} c^{\text{res}}_\ell. \tag{6}$$

If in RaM, the costs are computed approximately in order to preserve the scalability of the solution, in MaR all costs will be computed exactly. Since working segments do not share any bandwidth, each working segment uses exactly bandwidth d on each of its physical links. The working cost of the new incoming request on virtual edge q amounts to:

$$\alpha_q = \begin{cases} \|q\| \times d & \text{if } d \leq \gamma_q \\ \infty & \text{otherwise.} \end{cases} \tag{7}$$

We remark that a failure on a node affects all connections going through the node. A failure on a link adjacent to the node affects only a subset of these connections. Hence, the bandwidth needed for protecting the node is sufficient for protecting any link adjacent to it. We deduce:

Proposition 1: In order to protect a working segment against failures on nodes and links, we only need to protect nodes and then links will be automatically protected.

Theorem 1: The backup cost of the new incoming request on virtual edge q' for protecting a virtual edge q against any single link or node failure is:

$$\beta^q_{q'} = \begin{cases} \|q'\| \times (\max\limits_{v \in q} B^v_{q'} + d - B_{q'}) & \text{if } \delta^q_{q'} = 1 \text{ and} \\ & 0 \leq \max\limits_{v \in q} B^v_{q'} + d - B_{q'} \leq \gamma_{q'} \\ 0 & \text{if } B_{q'} - \max\limits_{v \in q} B^v_{q'} \geq d \\ \infty & \text{otherwise.} \end{cases} \tag{8}$$

Proof: When q, q' are not disjoint, i.e., $\delta^q_{q'} = 0$, they both fail upon a single failure at a common link or node, therefore $\beta^q_{q'} = \infty$. Otherwise, let us consider the backup bandwidth needed by the new incoming request on a physical link of q' in order to protect q against a failure on node $v \in q$. Within the existing backup bandwidth $B_{q'}$ on q', $B^v_{q'}$ is non sharable for covering a failure at v. The remaining bandwidth $B_{q'} - B^v_{q'}$ is sharable for every link of q'. Thus, the additional bandwidth that the new incoming request needs on each link of q' for protecting v is: $B^v_{q'} + d - B_{q'}$. In the single failure context, only one node can fail at a time. Hence, the additional backup bandwidth needed on a physical link of q' for protecting q against any single failure is: $\max\limits_{v \in q}(B^v_{q'} + d - B_{q'})$. Thus, the amount of additional backup bandwidth needed on the whole q' is $\|q\| \times \max\limits_{v \in q}(B^v_{q'} + d - B_{q'})$ which corresponds to (8) with the consideration of residual capacity. ■

From the backup cost $\beta^q_{q'}$, we deduce the backup cost $\beta^{\pi_i}_{q'}$ of virtual edge q' for protecting working segment π_i against any single link or node failure:

$$\beta^{\pi_i}_{q'} = \max_{q \in \pi_i} \beta^q_{q'} \tag{9}$$

With the defined working and backup costs, the Routing sub-problem is defined formally as finding a working path π and a set of backup segments $\{\pi'_i, i \in I\}$ so that the total bandwidth cost is minimized, i.e.,:

$$\min \sum_{q \in \pi} \alpha_q + \sum_{\pi'_i, i \in I} \sum_{q' \in \pi'_i} \beta^{\pi_i}_{q'}$$

It is equivalent to the single domain OSSP routing problem where the single domain network is the mapped network and links are virtual edges. Single domain OSSP routing solutions as well as the inter-domain routing solutions GROS and DYPOS proposed in [10] can be used to solve it. In the experimental results presented in this paper, DYPOS is used for the Routing step.

A. An exact and scalable solution for computing the backup cost of a virtual edge

The computation of the backup cost $\beta^q_{q'}$ as expressed in (8) requires the knowledge of $B^v_{q'}$ for each node v and PIP q'. It is an intra-domain information which changes dynamically after each routing. Therefore, maintaining all $B^v_{q'}$ up-to-date is a non-scalable requirement.

We propose a more scalable method for computing $\beta^q_{q'}$ with the following main idea. In each domain, there exists some critical nodes which belong to many PIP. The protection of these nodes can be sufficient for protecting some other nodes (see Th. 2 below). Therefore, the backup cost for protecting an PIP can be deduced from the backup cost for protecting certain critical nodes.

Definition 4: For a given domain, the Share Risk Group (SRG) of a node v, denoted by $\text{SRG}(v)$, is the set of virtual edges of the domain that share the same risk at v. It corresponds to the set of PIPs going through v.

SRGs have the following characteristic:

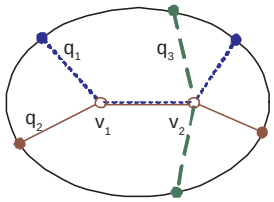

Fig. 6. $\text{SRG}(v_1) = \{q_1, q_2\} \subset \text{SRG}(v_2) = \{q_1, q_2, q_3\}$ because all PIPs going through v_1 are going through v_2.

Theorem 2: Let $\text{SRG}(v_i)$ and $\text{SRG}(v_j)$ be two SRGs so that $\text{SRG}(v_i) \subseteq \text{SRG}(v_j)$, then

$$B_{q'}^{v_i} \leq B_{q'}^{v_j}. \tag{10}$$

In other words, if all PIPs that go through one node (v_i), go also through another node (v_j), the backup bandwidth reserved on an PIP for protecting the first node (v_i) does not exceed the backup bandwidth reserved on the same PIP for protecting the second node (v_j), see Fig. 6.

Proof: Since $\text{SRG}(v_i) \subseteq \text{SRG}(v_j)$ then:

$$\text{SRG}(v_j) = \text{SRG}(v_i) \cup \Big(\text{SRG}(v_j) \setminus \text{SRG}(v_i) \Big).$$

Thus:

$$\sum_{q \in \text{SRG}(v_j)} B_{q'}^{q} = \sum_{q \in \text{SRG}(v_i)} B_{q'}^{q} + \sum_{q \in \text{SRG}(v_j) \setminus \text{SRG}(v_i)} B_{q'}^{q}.$$

Consequently,

$$\sum_{q \in \text{SRG}(v_j)} B_{q'}^{q} \geq \sum_{q \in \text{SRG}(v_i)} B_{q'}^{q}.$$

From the definition of $B_{q'}^{v}$, we have $B_{q'}^{v} = \sum_{q \in \text{SRG}(v)} B_{q'}^{q}$. Thus,

$$B_{q'}^{v_j} \geq B_{q'}^{v_i}. \qquad \blacksquare$$

Definition 5: A SRG is maximal if it is not contained in another SRG.

If two SRGs of two different nodes are identical and maximal, one node will be chosen as the representative node for the maximal SRG. If one of the two nodes is a border node, it will be chosen. When both nodes are internal nodes, we can choose any of them.

From Th. 2 we deduce the following proposition.

Proposition 2: Let q be a sub-path and v_j be a node on q such that $\text{SRG}(v_j)$ is maximal and denoted by $\text{SRG}^{\text{MAX}}(v_j)$. We have:

$$\max_{v \in q} B_{q'}^{v} = \max_{v_j : q \in \text{SRG}^{\text{MAX}}(v_j)} B_{q'}^{v_j}. \tag{11}$$

Readers are referred to [9, ch. 7] for the detailed proof.

Prop. 2 provides a way to compute $\max_{v \in q} B_{q'}^{v}$.

Let v_1, v_2 be two border nodes of PIP q and N_m be the domain containing q. By separating the v_j border node and v_j non-border node in (2), we obtain:

$$\max_{v \in q} B_{q'}^{v} = \max\{B_{q'}^{v_1}, B_{q'}^{v_2}, \max_{\substack{v_j \in V_m \setminus V_m^{\text{BORDER}}: \\ q \in \text{SRG}^{\text{MAX}}(v_j)}} B_{q'}^{v_j}\}. \tag{12}$$

Thus, $\max_{v \in q} B_{q'}^{v}$ is identified from $B_{q'}^{v_1}$, $B_{q'}^{v_2}$ of border nodes and $B_{q'}^{v_j}$ of non-border nodes $v_j \in V_m \setminus V_m^{\text{BORDER}}$ whose SRG are maximal (since q is direct intra-path, there is no other border nodes to be considered). In substituting the right hand-side of (12) in (8), we found that the backup costs $\beta_{q'}^{q}$ can be computed from the backup cost of q' for protecting some non-border maximal SRG nodes and those for protecting border nodes of the working PIP q. In comparison with (8), this new computation is more scalable since it relates only some nodes of the working PIP.

VI. Scalability

Since the Mapping step is performed independently within each domain, it does not encounter any scalability problem. Let us discuss the scalability issue in the Routing process. Let E^{VEDGE} be the set of virtual edges in the mapped network. The following parameters are required by the Routing process for computing the working and backup costs associated with each virtual edge $q \in E^{\text{VEDGE}}$ by using (7), (8) and (12):

- Cat.A : $\|q\|, B_q$;
- Cat.B : B_q^v for all $v \in V^{\text{BORDER}}$;
- Cat.C : All non-border SRG^{MAX} in every domain as well as their associated internal nodes v;
- Cat.D : B_q^v for the internal nodes v associated to the non-border SRG^{MAX} of Cat.C;
- Cat.E : γ_q.

Each border node should store, maintain up-to-date and exchange the above parameters, for each $q \in E^{\text{VEDGE}}$, with the other border nodes by using a BGP like protocol. The values in Cat.A and Cat.B are per virtual edge or border node. They can be updated without impairing the scalability. In Cat.C, the set of non-border SRG^{MAX} depends uniquely on the Mapping step and are therefore stable values. The experimental results in Section VII will show that the number of non-border SRG^{MAX} is quite small leading to a small number of B_q^v in Cat.D.

While the residual capacity on every physical link that participates in q is not smaller than the maximal requested bandwidth, the residual capacity γ_q of virtual edge q is sufficient for any new request and does not need to be updated. Otherwise, γ_q needs to be recalculated exactly by using (6).

In summary, most of the information required in the routing of MaR is per virtual edge and is managed at the mapped level. The quantity of required internal domain information is small. The scalability is thus preserved.

VII. Experimental results

MaR-O and MaR-G are compared with the optimal single domain OSSP solution [6], denoted by Opt, and the multi-domain OSSP solutions GROS and DYPOS [10]. We set $\mu_1 = \max_{\ell \in L_m} c_\ell^{\text{res}}$, $\mu_3 = 1/(n^{\text{W}} \times |E_m^{\text{VIRTUAL}}|)^2$, $\mu_4 = 1$ and $\mu_2 = 0$. The number of needed PIPs per virtual link is $n^{\text{WB}} = 2$. The number of intra-path candidates for MaR-G is $n^{\text{CAN}} = 4$. In MaR-O, MaR-G, GROS and DYPOS, working (resp. backup) segment length is limited by threshold l^{W} (resp. l^{B}).

Domains	(μ_1) cost	($-\mu_3$) dj_{ip}	($-\mu_4$) dj_{vl}	obj
EON (1)	-2,17	-33	0	2.52
RedIRIS (2)	0	0	0	0
GARR (3)	0	0	0	0
Renater (4)	25	0	0	26.48
SURFnet (5)	7,81	0	0	10.89

cost (%): relative gap on PIP cost.
dj_{vl} (%): relative gap on number of disjoint virtual links.
dj_{ip} (%): relative gap on number of disjoint PIPs.
obj (%): relative gap on the overall objective function.

TABLE I

RELATIVE GAP OF MaR-G VS. MaR-O IN LARGE-5.

Domains	1	2	3	4	5	6	7	8	Total
LARGE-5									
Nb. org. SRGs	12	16	13	16	22	-	-	-	79
Nb. adv. SRGs	3	1	1	0	1	-	-	-	6
LARGE-8									
Nb. org. SRGs	16	16	21	18	18	20	15	17	141
Nb. adv. SRGs	1	4	8	3	8	10	4	6	44

TABLE II

NUMBER OF SRGS IN LARGE-5 AND LARGE-8 WITH MaR-G

Two multi-domain network topologies are mainly used for the experiments: LARGE-5 and LARGE-8 [9]. They compose respectively of 5 and 8 domains, each one has $15-29$ nodes and $23-53$ links. LARGE-5 is built from 5 real optical networks. LARGE-8 is generated using the Transit-Stub model of the multi-domain network generator GT-ITM [12]. Each domain has on average 4 neighboring domains in order to reflect faithfully the Internet interconnections [8].

A. Mapping evaluation

The greedy Mapping MaR-G is compared with the optimal Mapping MaR-O on LARGE-5 only due to high computational effort of MaR-O in LARGE-8. Table I gives the relative gaps of MaR-G over MaR-O on each mapping criterion and the overall mapping objective. The gaps remain small or null in most of cases illustrating the efficiency of the proposed greedy Mapping. Therefore, from now on, the experimental results on large networks are shown only with MaR-G.

B. Scalability in using non-border maximal SRGs

The scalability is evaluated through the number of non-border maximal SRGs needed to be advertised amongst domains. The smaller this number is, the more scalable the solution is. Tables II shows the significantly small number of SRGs that needs to be advertised (denoted by adv.) in LARGE-5 and LARGE-8 in comparison with the number of original SRGs (denoted by org.). This confirms the scalability efficiency in using only non-border maximal SRGs in backup cost computation while maintaining the accuracy of the cost.

C. Routing evaluation

Let us first introduce the metrics for evaluating the routing.

The working (resp. backup) network cost is the total working (resp. backup) bandwidth used by all network links.

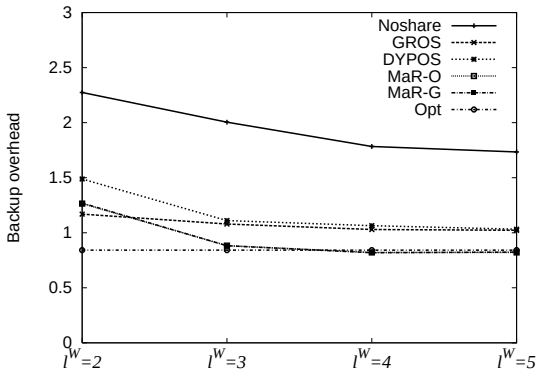

Fig. 7. Comparison with Opt on Backup overhead in SMALL-5

The *Backup overhead* is the ratio between the total working and backup network costs and the smallest working network cost less 1. This amounts to the backup bandwidth redundancy of a protection scheme.

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

1) Comparison with optimal single domain OSSP solution: Due to the extremely high computational effort needed for Opt, the schemes MaR-O, MaR-G, Opt, GROS, DYPOS and NoShare, a dedicated protection scheme, are compared only on SMALL-5, a small 5 domain network. The network is composed of 28 nodes and is generated again by GT-ITM. All requests remain active in the network.

Fig. 7 depicts the backup overheads in SMALL-5. Due to the small size of the network, the backup segment length constraint is removed. In most of cases, MaR-O, MaR-G outperform GROS, DYPOS and provides nearly identical backup overheads to Opt, revealing their high performances in bandwidth saving. The absence of the segment length constraint in Opt explains partially why it is better than MaR-O, MaR-G when the segment length threshold is very small.

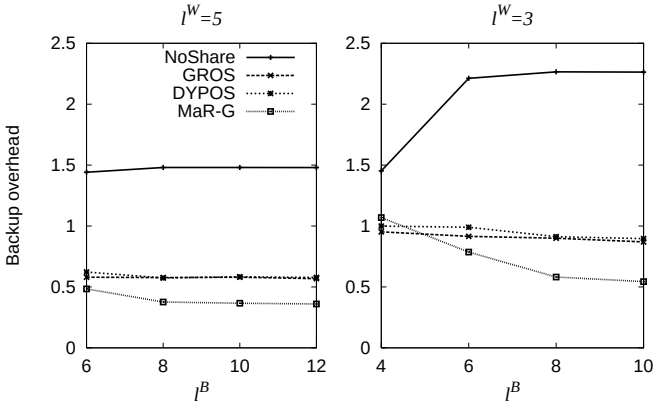

Fig. 8. Backup overhead in LARGE-8

2) Backup overhead: The Routing with a single optimization and the exact cost computing brings to MaR a better

bandwidth saving quality over RaM, which uses multiple optimizations in routing and approximation in working and backup cost computations.

We conducted experiments with an incremental traffic of 1000 requests where all requests remain active. Network links are uncapacitated in order to avoid the blocking cases which varies from one scheme to the other and thus make the analysis more complex. The experiments are performed on LARGE-5 and LARGE-8 however due to the similar results, we show only the results in LARGE-8, the other can be found in [9].

Fig. 8 depicts backup overheads. Obviously, MaR-G, GROS and DYPOS give better backup overheads than NoShare. As expected, MaR-G provides generally a smaller backup overhead than GROS and DYPOS.

3) Blocking probability: The blocking probability is examined under dynamic traffic. Requests arrive according to a Poisson process with rate $r = 1$ and exponential holding time with mean $h = 320$.

In general, MaR-G provides clearly smaller blocking probability than DYPOS, GROS and NoShare (Fig. 9). An insight in GROS and DYPOS reveals that most of their blockings are caused by bad guidances obtained from the inter-domain routing due to the cost approximation and the impossibility of mapping virtual links in the intra-domain step so that their working and backup segments are disjoint. MaR-G overcomes these weaknesses by using a unique routing based on precise working and backup costs of virtual edges as well as their disjointness indexes.

However, we observe from the results on both backup overhead and blocking probability that when segment lengths are highly limited, i.e., $l^{\mathrm{W}} = 3$ or small l^{B}, MaR-G sometimes looses its advantage. The reason is that it is more difficult for MaR-G to build a solution satisfying segment length constraints from the restricted number of PIPs, $n^{\mathrm{WB}} = 2$, than GROS and DYPOS which have no restriction in PIPs.

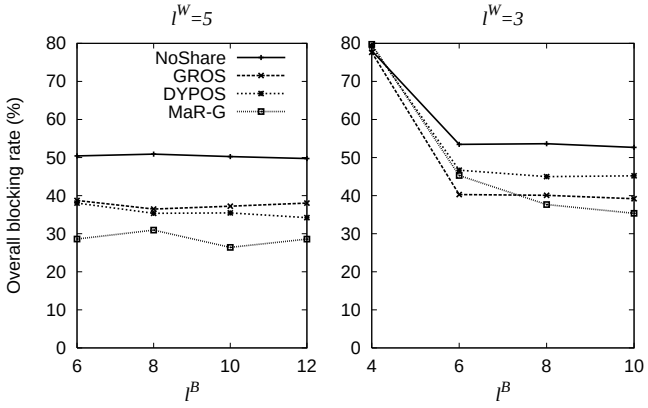

Fig. 9. Overall blocking probability in LARGE-8

VIII. Conclusion

The MaR approach, with the restriction of the number of PIPs, benefits from an exact highly scalable routing, although it sacrifices some small possible backup bandwidth sharing and leaves, a priori, less choices for building working and backup segments. Nevertheless, the Mapping with multiple well defined criteria transforms this restriction into a mechanism which directs the Routing to the best intra-paths in terms of cost, disjointness and sharing possibility. In addition, the routing of MaR with a single step improves the quality of bandwidth optimization over the two step routing of RaM.

The experimental results also confirm that MaR outperforms RaM on bandwidth saving and blocking probability. Furthermore, in bandwidth saving, MaR is close to the optimal single domain solution while the latter is not scalable even for a large single domain network.

MaR can also be applied for WDM multi-domain networks. Since PIPs are fixed after the Mapping step, we can allocate statically one wavelength for each PIP which becomes an optical lightpath. Wavelengths may need to be changed only at border nodes. Each network domain remains all optical without wavelength conversion at internal nodes.

References

[1] 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 Conf.*, Sept. 2002.

[2] J. Cao, L. Guo, H. Yu, and L. Li, "A novel recursive shared segment protection algorithm in survivable WDM networks," *Journal of Network and Computer Application*, vol. 30, no. 2, pp. 677–694, Apr. 2007.

[3] A. Farkas, J. Szigeti, and T. Cinkler, "P-cycle Based Protection Schemes for Multi-Domain Networks," in *Proc. Intl. Workshop on Design of Reliable Communication Networks (DRCN)*, Oct. 2005, pp. 223–230.

[4] L. Guo, "LSSP: A novel local segment-shared protection for multi-domain optical mesh networks," *Computer Communications*, vol. 30, no. 8, pp. 1794–1801, June 2007.

[5] 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, Feb. 2002.

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

[7] J. L. Le 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.

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

[9] D. L. Truong, "Protection partagée pour les réseaux de transport multidomaines," Ph.D. dissertation, Dept. of Computer Science and Operations Research, Université de Montréal, 2007, http://www.iro.umontreal.ca/~truongtd/papers/ThesisPhD2007.pdf.

[10] D. L. Truong and B. Jaumard, "Using Topology Aggregation for Efficient Segment Shared Protection Solutions in Multi-domain networks," *IEEE J. of Selected Areas in Communication/ Optical Communications and Networking series*, vol. 25, no. 9, pp. 96–107, Dec. 2007.

[11] 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 Conf. on Comm.*, Oct. 2002, pp. 1320–1331.

[12] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to Model an Internetwork," in *IEEE Infocom*, vol. 2, Mar. 1996, pp. 594–602.