Resource Sharing Strategies: Which Sharing to Better Protect Primary Shortest Paths?

Mohand Yazid Saidi, Bernard Cousin

October 2015

Resource Sharing Strategies: Which Sharing to Better Protect Primary Shortest Paths?

Mohand Yazid SAIDI [a], Bernard COUSIN [b]
[a] *L2TI/Institut Galilée, Université Paris 13 Sorbonne Paris Cité, 99 Avenue Jean Baptiste Clément, 93430 Villetaneuse, France; E-mail: saidi@univ-paris13.fr*
[b] *IRISA, Université de Rennes I, France*

Abstract. With the widespread use of real-time applications (VoIP, IPTV, Video conference, etc.) in Internet, protection and resource optimization become increasingly desired. Network protection aims to decrease the interruption time of communications by precomputing backup paths capable to receive and route traffics of affected primary paths upon failures. Resource optimization is achieved by improving data routing and resource sharing: data routing is often optimized by following the shortest paths whereas resource sharing is applied between the backup paths protecting against different failure risks. Two strategies of resource sharing are defined in literature: (1) backup sharing which limits the resource sharing to the backup paths and (2) global sharing which extends the resource sharing to the primary and backup paths.
In this paper, we compared the effects of resource sharing strategies on the resource utilization when the primary paths correspond to the shortest ones according to a static metric. With the single failure assumption, we show formally that the resource sharing between primary and backup paths is limited to some few links which cannot form a backup path. Thus, independently of the amount of resources (for instance: bandwidth) that can be shared between the primary and backup paths, the maximum number of backup paths is bounded. In our simulation, we comfort our formal result by showing that the two strategies have close acceptance rates of backup paths and protection bandwidth utilizations.

Keywords. Routing, backup path, local path protection, resource sharing, shortest paths, MPLS, virtual networks

1. Introduction

Most of today's applications (IPTV, videoconferences, VoIP, etc.) are very sensitive to the disruption of communications and consume more and more resources (such as bandwidth). Hence, protection against failures is becoming very desirable to prevent or reduce the disruption time of communications. In addition, since path protection consumes network resources if backup paths are pre-configured and their resource reserved, resource optimization is required to improve the network resource utilization.

Network protection [1,2,3,4,5] maintains the communication service continuity by precomputing and generally pre-configuring backup paths capable to reroute traffics of

affected primary paths upon a failure. To ensure resource availability[1] after a failure repair, the primary and backup paths should reserve their resources. Whereas the primary paths really use the resources they reserved, the backup paths doesn't consume any resources before a failure occurrence. Hence, under the single failure assumption, resources can be shared between all the backup paths which protect against different failure risks, since these backup paths cannot be active at the same time. In addition of the resource sharing, the primary paths should follow the shortest paths in order to achieve resource optimization. For instance, the internet routing protocols (RIP [6] and OSPF [7]) are designed to save resource by allowing the computation of shortest paths. Similarly, for virtual netwok embedding k-shortest paths are often chosen to map the virtual links [8,9].

With the arrival of MultiProtocol Label Switching (MPLS) [10] in the few past years, protection [11] and resource optimization [5] are provided efficiently.

Firstly, fast recovery and availability of resources are guaranteed with the preconfiguration of local backup paths capable to bypass any failure risk (a failure risk could be a link, a node or a SRLG[2]). Local backup paths ensure fast reaction to failure due to their locality : fast detection and fast rerouting. Two types of backup paths are defined in MPLS for local protection [3]: Next HOP backup Label Switched Path (NHOP LSP[3]) and Next Next HOP backup Label Switched Path (NNHOP LSP). A NHOP LSP (resp. NNHOP LSP) is a backup LSP protecting against link failure (resp. node failure); it is setup between a Label Switched Router (LSR) called Point of Local Repair (PLR) and one LSR called Merge Point (MP). The PLR is the LSR upstream to the failure point. The MP is located between the next-hop (resp. next-next-hop) of the PLR and the destination. The NHOP (resp. NNHOP) backup LSP bypasses the link downstream (resp. the node downstream) to the PLR on the primary LSP. When a link failure (resp. node failure) is detected by an upstream node, this later activates locally all its NHOP and NNHOP (resp. NNHOP) backup LSPs by switching traffic from the affected primary LSPs to their backup LSPs.

Secondly, much resources can be saved thanks to the flexibility in path selection offered by MPLS. Indeed, an appropriate selection of primary and backup paths can increase the bandwidth sharing and thus decrease the bandwidth allocations.

In the last recent years, more attention was given to the virtual networks. For a better use of resources, virtual networks are computed so that they consume less resources. Due to the NP-hardness of the problem of mapping a virtual network to a substrate network (Virtual Network Embedding or VNE), most of the proposed solutions use pre-computed (k-)shortest paths. Like in classical networks, two types of protection could be applied to ensure survivability : global and local. With the global protection, a primary virtual link (which corresponds to a substrate path) is protected by a disjoint backup virtual link connecting the same extremities [14,15]. Two virtual links are said disjoint if they don't share any link or internal node in the substrat network. With the local protection [16], each link or node belonging to a substrat primary path (i.e. primary virtual link) is protected locally by a backup path which bypasses it.

[1] In the rest of this document, *resource* refers to *bandwidth*.

[2] A SRLG [12,13] corresponds to a set of logical links that share a common physical component (optical fiber, crossconnect, etc.) whose single failure may impact all links in the set.

[3] A LSP is a path through an MPLS network.

Two main resources sharing strategies are defined in literature: 1) backup resource sharing [2] and 2) global resource sharing [1]. With the first strategy, the resource sharing is limited and applied to the backup paths protecting against different failure risks. As these backup paths cannot be active at the same time (due to the single failure assumption), they cannot ask for their resources simultaneously and thus they can share them. With the second strategy, the resource sharing is extended and applied to primary and backup paths. Concretely, since a backup path can bypass several links and/or nodes of a primary path, some resources can be freed on the primary affected paths. Such resources can be reallocated to the backup paths.

In this paper, we study the impact of resources sharing strategies on the resource utilization when the primary paths are the shortest ones. After reviewing works related to the resource sharing in Section 2, we introduce and explain in more details the principles of the backup and global resource sharing strategies. Then, we determine in Section 3 the formulas computing the amount of sharable and allocated resources with application to the two sharing strategies. In Section 4, we study formally the impact of resource sharing strategies on the resource utilization when the primary paths are the shortest ones. We show that the impact of the primary path resources freed upon a failure is very low and negligible on the protection capability. In Section 5, we compare and measure by simulations the gain obtained by global resource sharing instead of backup resource sharing. Finally, Section 6 is dedicated to the conclusions.

2. Related Works

In the last two decades, a great deal of work is addressing network protection to find efficient algorithms and mechanisms providing survivability and optimizing network resource utilization.

In [17,18], several network coding-based strategies are described to provide protection in optical and also higher layers. In [19], re-optimization heuristic is proposed in order to decrease the risk of link congestion and thus avoid service disruption. With such heuristic, resource allocations are balanced over the network, mainly by rearranging and rerouting the paths. In [20], an extensive survey of the recovery methods is given. These methods are classified according to different criteria such as the *layer* in which recovery methods are applied (Physical Layer, Link Layer, Network Layer, etc.), *computation and/or establishment moment* of the backup paths (before failure for protection and after failure occurrence for the restoration), *resource usage* (without resource sharing or with resource sharing), *scope* (global or local protection) and *domain* (intra-domain or inter-domain protection). In MPLS networks, and under different network parameters and constraints, [21,22] propose various comparison metrics, such as the packet loss, rejection probability and restoration time, to evaluate the level of protection. Unfortunately, nether [21,22] nor [20] consider global sharing in their studies.

For MPLS networks, global and local protection with/without resource sharing can be applied in both intra-domain and inter-domain. With the global protection [23,2], two disjoint paths connecting the source and target nodes are computed: one primary path used to transmit traffic before any failure occurrence and one backup path that should be activated and used for routing upon any failure affecting the primary path. With the local protection [1,24,25], for every link and/or node of the primary path, one local backup

path (NHOP LSP or NNHOP LSP) bypassing the protected link and/or node is computed. When a failure occurs, the traffic is switched locally at the PLR to the backup paths bypassing the failed risk. In [26], Li et al. proves that joint resource optimization of primary and local backup path is an NP-hard problem.

Recently, several methods were proposed for virtual network survivability [27,28]. Due to the complexity of the survivable virtual network embedding (SVNE) problem, this later is generally subdivided into two sub-problems (VNE and protection) which can be solved separately. Whereas on-line protection is applied to protect one path in classical networks, many substrat paths should be protected together to provide protection. In [15], the authors proposed to protect each primary substrat path by a disjoint shortest substrat path. To save resources, the backup paths minimize the additional bandwidth. In [16], the authors propose to protect locally the substrat links which are used to form the virtual links. In their approach, Guo et al. firstly chose a subset of primary and backup paths before calling a linear procedure that optimizes bandwidth allocation while balancing the load. In [27,28], various approaches providing survivability for virtual networks are described. These approaches are grouped in various categories according to the type of recovery (proactive or reactive), the protected network component (node or link), the recovery procedure (replication or flow rerouting), the scope (centralized or distributed), etc. [29] proposes to combine the failure dependent protection (FDP) with the failure independent protection (FIP) to improve the resource utilization. FDP provides node protection by duplicating nodes (i.e. associating backup nodes to the primary nodes) whereas FIP corresponds to the classical protection which configures a set of backup paths. In [30], disaster failures which involve multiple simultaneous failures located in the same geographic region are treated. After modeling disaster failures, the paper proposed mixed integer linear programming and prediction-based heuristics to deal with such type of failures.

To increase the acceptance rate of protection requests (i.e., to improve the resource utilization), [24] proposes a global resource sharing strategy for on-line protection. Contrarily to the backup resource sharing strategy which limits the resource sharing to the backup paths, Mélon et al. [24] suggest to pre-allocate the resources freed by the deactivated (or bypassed) primary path segments upon a failure to the backup paths which will be activated to recover from that failure (see Section3). In order to minimize the resource allocation, [1] proposes a resource sharing-based cost function that measures the amount of extra spare resources required to cross a given link. Obviously, larger are the primary resources freed on a link upon a failure, smaller is the cost of this link for that failure. As this link cost function depends only on the capacities of resource sharing, the backup path and thus the recovery time may be arbitrary long. Indeed, the backup paths optimizing the cost function may include very long paths which induce high transmission delays. In addition, optimizing the resource allocation does not systematically improve the acceptance rate because of resource sharing capabilities of backup path computations.

Although the primary paths often correspond to the shortest ones, none of the described works studies the impact of such primary routing decision on the protection rate and bandwidth sharing capabilities. In this paper, we try to fill the gap by extending the work in [31] to empirically measure the impact of an optimal primary routing on the quality of resource allocation. In addition, we provide the theoretical results proving that the improvements due to the freed resource reallocation are insignificant compared to those obtained with the utilization of resource sharing between the backup paths.

3. Bandwidth Allocation Model

Before the presentation of the link admission control that takes into account bandwidth sharing (section 3.1) for path computation, we first give some notations and definitions useful to the understanding of our control admission models (section 3.2). Furthermore, these definitions enable the description of the context of our study.

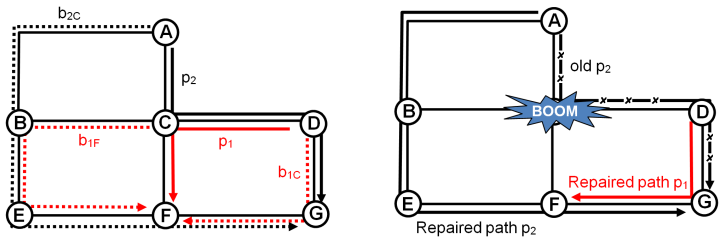
3.1. Notations

Let us consider a directed graph $G = (V, E, \overrightarrow{w}, \overrightarrow{C} = \overrightarrow{PC} + \overrightarrow{RC})$ where V is the set of vertices, E the set of links and $\overrightarrow{w}$, $\overrightarrow{PC}$ and $\overrightarrow{RC}$ are functions that associate respectively to each link ($\lambda \in E$) a strictly positive constant weight $w(\lambda)$, a primary bandwidth capacity expressed in bandwidth units PC^{λ} and a finite protection bandwidth capacity RC^{λ} expressed in bandwidth units. We define:

- the weight $w(\pi)$ of a primary path π as follows:
  $w(\pi) = \sum_{\lambda \in \pi} w(\lambda)$.
- $Pr_r^{(s,t)}$ as the set of primary paths crossing the failure risk r (link, node or a set of links and/or nodes) and interconnecting the source node s to the target node t. All the paths in $Pr_r^{(s,t)}$ are the shortest ones.
- P^{λ} as the primary bandwidth that should be reserved on each link λ to carry out traffic before failures. It is computed as the cumulated bandwidth of primary paths crossing the risk λ and connecting s to t.
- $Bp_r^{(s,t)}$ as the set of backup paths protecting the primary paths in the set $Pr_r^{(s,t)}$.
- δ_r^{λ} as the protection cost of risk r on link λ. It corresponds to the cumulated bandwidth of backup paths which will be activated on link λ to cope with the failure of risk r.
- L_r^{λ} corresponds to the (primary) bandwidth freed on link λ upon failure of risk r.
- two bandwidth allocation methods on links: bidirectional and unidirectional. With the first method, any two unidirectional links ($u \rightarrow v$ and $v \rightarrow u$) which share the same physical conductor $u - v$ use the same pool for bandwidth allocations. With the second bandwidth allocation method, each unidirectional link has its own autonomous pool that it uses for bandwidth allocations.
  For the ease of understanding and without loss of generality, we will focus in this paper on the case of unidirectional bandwidth allocations. As the case of bidirectional bandwidth allocations can be treated in the same way, only the results of simulations are given and discussed.

3.2. Bandwidth Constraints and Allocations for the Backup Paths

Using local protection mode, regardless of resource sharing strategies, $N-1$ local backup paths (detours) should be built to protect a path that traverses N nodes. For instance, to fully protect path $p_1 = D \rightarrow C \rightarrow F$ in Figure 1 (a), two backup paths $b_{1C} = D \rightarrow G \rightarrow F$ and $b_{1F} = C \rightarrow B \rightarrow E \rightarrow F$ are established. The first backup path interconnects the PLR node D to the merge node F whereas the second backup path connects the PLR node C to the merge node F. Thus, the first backup path is a NNHOP LSP protecting against the failures of PLR's downstream node (C) and link ($D-C$) whereas the second backup path is a NHOP LSP protecting against the PLR's downstream link ($C-F$).



(a) Before failure

(b) After the recovery from failure of node C

Figure 1. MPLS protection and bandwidth sharing

To improve the acceptance rate of path establishment requests, resources like the bandwidth should be saved by sharing them. Indeed, under the practical hypothesis of simple failure that we adopt in this paper (as in many articles [1,32,33]), some paths cannot carry traffic at the same time: they can therefore share their bandwidth allocations. For this purpose, two main bandwidth sharing strategies were defined: 1) backup bandwidth sharing and 2) global bandwidth sharing.

With the first bandwidth sharing strategy, the bandwidth sharing is applied and limited to the backup paths that protect against different failure risks. In Figure 1 (a), the backup path b_{2C} ($A \rightarrow B \rightarrow E \rightarrow F \rightarrow G$) protecting the primary path p_2 ($A \rightarrow C \rightarrow D \rightarrow G$) against failures of node C and link $A-C$ can share its resource allocation (for instance, on links $B-E$ and $E-F$) with the backup path b_{1F} ($C \rightarrow B \rightarrow E \rightarrow F$) which protects the primary path p_1 ($D \rightarrow C \rightarrow F$) against the failure of link $C-F$. Indeed, paths b_{2C} and b_{1F} cannot be active at the same time since they protect against disjoint sets of failure risks (failure of link $A-C$ for b_{2C} and, link $C-F$ for b_{1F}). Thus, after determining the protection cost δ_r^λ of risk r on link λ which correspond to the cumulative bandwidth of backup paths that should be activated to recover from failure r, we determine the protection bandwidth[4] R^λ that should be reserved for protection on the (unidirectional) link λ as follows:

$$R^\lambda = \max_r \delta_r^\lambda \tag{1}$$

The total bandwidth $bw(\lambda)$ allocated on λ must be always smaller than the capacity C^λ of link λ:

$$bw(\lambda) = P^\lambda + R^\lambda = P^\lambda + \max_r \delta_r^\lambda \leq C^\lambda \tag{2}$$

where P^λ is the cumulative bandwidth of the primary paths traversing link λ.

In addition of the bandwidth sharing between the backup paths, more bandwidth could be saved by reallocating the bandwidth freed by the bypassed part of the primary path affected by the failure [1]. For instance, to recover from failure of node C in Figure 1 (a), the traffics of the primary paths p_1 and p_2 will be rerouted and switched to the backup paths b_{1C} and b_{2C} respectively. As shown in Figure 1 (b), the recovery from the failure of node C frees up bandwidth on some primary links. Typically, when node C

[4]Protection bandwidth corresponds to the minimum amount of bandwidth that should be reserved for backup paths, to ensure the availability of enough bandwidth after any single failure.

fails, the traffic of primary path p_2 (old p_2 in Figure 1 (b)) will be switched onto backup path b_{2C} (repaired path p_2 in Figure 1 (b)). Thus, some bandwidth will be freed on link $D-G$ after the node failure recovery. In order to save bandwidth, the global bandwidth sharing strategy proposes to reuse the bandwidth freed up[5] after the recovery of failure r by reallocating it to the backup paths that protect against the same failure r. In our example of Figure 1, the bandwidth allocated on link $D-G$ can be shared between the primary path p_2 and the backup path b_{1C} (see Figure 1 (a)) since these two paths cannot be active at the same time. Indeed, in the absence of C failure, link $D-G$ carries only the traffic of the primary path p_2. If node C fails, only the traffic of the activated backup path b_{1C} will traverse the link $D-G$ since the traffic of the primary path p_2 will be switched to the backup path b_{2C} that does not traverse $D-G$.

$$R^{\lambda} = \max_{r}(\delta_r^{\lambda} - L_r^{\lambda}, 0) \tag{3}$$

We deduce the total amount of bandwidth $bw(\lambda)$ allocated on the link λ as follows:

$$bw(\lambda) = P^{\lambda} + R^{\lambda} = P^{\lambda} + \max_{r}(\delta_r^{\lambda} - L_r^{\lambda}, 0) \leq C^{\lambda} \tag{4}$$

Note that all the parameters (P^{λ}, δ_r^{λ} and L_r^{λ}) that are necessary to verify the admission control (i.e. ensure the availability of enough bandwidth before and after any failure) are known by the extremity nodes of link λ since these two nodes know all the paths that traverse them.

Furthermore, to control and specify the amount of resources that should be used for protection and to separate the computation task of primary paths from that of backup paths, the bandwidth capacity of each link λ can be divided in two separate pools: primary bandwidth pool PC^{λ} and protection bandwidth pool RC^{λ}. The primary bandwidth pool is used to allocate bandwidth for primary paths whereas the protection bandwidth pool is used to allocate bandwidth for backup paths. With such bandwidth allocation model, link λ can be included in the computation of a new backup path iff the resulted protection bandwidth on link λ remains lower or equal to the protection capacity.

When the backup resource sharing strategy is applied:

$$R^{\lambda} = \max_{r} \delta_r^{\lambda} \leq RC^{\lambda} \tag{5}$$

When of global resource sharing strategy is applied:

$$R^{\lambda} = \max_{r}(\delta_r^{\lambda} - L_r^{\lambda}, 0) \leq RC^{\lambda} \tag{6}$$

Although it seems that the global resource sharing strategy is more efficient than the backup resource sharing strategy, the blocking probabilities[6] of the two strategies could be very close, especially when the primary paths correspond to the shortest ones in terms of a static[7] metric. In the two following sections, we prove formally and by simulations that both the two strategies of resource sharing have close blocking probabilities.

[5]Only the primary links located between the extremity nodes of the activated backup path frees up bandwidth.

[6]The blocking probability corresponds to the probability that a request of path establishment will be rejected due to the lack of network resources (bandwidth).

[7]We recall that a metric is said to be static if its values on links do not change.

4. After-effect of the Amount of Freed Bandwidth on the Backup Path Acceptance

The majority of the well known IGP protocols computes the primary paths as the shortest ones in terms of a static metric (i.e., traffic independent costs). For instance, RIP minimizes the hop number (number of intermediate routers in a path) while OSPF applies the SPF (shortest path first) algorithm to optimize a static metric that depends generally on bandwidth capacities of links. With the advent of MPLS, the IP routing protocols (OSPF-TE and ISIS-TE) are extended to take into account the traffic characteristics in route computations. This leads to the definition of new (semi-)dynamic metric-based routing algorithms which often applies the Dijkstra's shortest path algorithm. For instance, CSPF (constrained shortest path first) prunes links that do not meet the configured constraints from the topology network before applying the SPF algorithm that derives the best available path based on the information in the traffic engineering database. In other words, CSPF always returns a shortest path while the pruned links don't cut all the possible shortest paths between two nodes.
In addition of the IGP protocols, we note that for VNE the k-shortest paths are often selected to map the virtual links to the primary substrat paths (i.e. primary virtual links).

In this section, we show that when the primary paths follow the shortest paths according to any static metric, the maximum number of backup paths is bounded even if the primary bandwidth freed upon failure is infinite (i.e., the freed bandwidth is very larger than the protection bandwidth). This means that any backup path must cross at least one link which cannot free up bandwidth upon failure of the protected risk (i.e. we cannot build a backup path with only links freeing bandwidth upon the failure of the protected risk). Hence, the maximum number of backup paths which can be built is bounded at least by the capacities of links which cannot free up bandwidth (upon the considered failure).

Before detailing the proof of our assertion, let us consider an example. In Figure 2, a network topology of equal-cost links is depicted. Assume that any primary path follows a shortest path and requires a minimum of 1 bandwidth unit. To protect a primary path traversing node D, link $D-G$ then node G (in this order), it is sufficient to determine a backup path that connects the PLR D to node G or any node downstream to G (on the

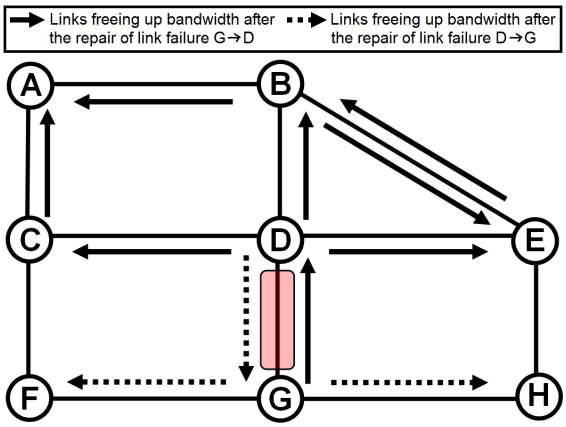


Figure 2. Links that are able to free up bandwidth upon a failure of link $D-G$

shortest primary path). As the primary paths should follow the shortest paths (dashed arrows in Figure 2), only nodes F and G can be located on the downstream of G. In addition, the shortest primary paths traversing node D, link $D-G$ and node G (in this order) can only free up bandwidth on links $G-F$ and $G-H$ located on the downstream of the failed link $D-G$. In a same way, we deduce that any backup path protecting a primary path traversing node G, link $G-D$ and node D (in this order) should connect the PLR G to node D or any node downstream to D on a shortest path (i.e. any node in $\{A, B, C, E\}$). The links that are able to free up bandwidth on a primary path traversing node G, link $G-D$ and node D (in this order), upon a failure of link $G-D$ are: $D-C$, $C-A$, $D-B$, $B-A$, $B-E$ (in the two directions) and $D-E$ (links associated to bold arrows in Figure 2).

From the precedent remarks, we conclude that any backup path protecting link $D-G$ should use link $H-E$ or link $F-C$. Since these two last links will not free up bandwidth after the failure of link $G-D$, we deduce that the number of backup paths protecting link $G-D$ is bounded by the capacities of links $H-E$ and $F-C$ (the protection costs δ_{G-D}^{H-E} and δ_{G-D}^{F-C} increase with the establishment of backup paths protecting against the failure of link $G-D$).

Even if we consider that the freed bandwidth upon failure r is infinite on all the links that are capable to free up bandwidth (example: when the primary capacities are infinite whereas the protection capacities are finite), we show formally in the next paragraphs that the maximum number of backup paths is bounded. Without loss of generality, we assume that any backup path requires a minimum of 1 bandwidth unit and the protection bandwidth is bounded.

Lemma 4.1 *Any backup path protecting a primary shortest path (according to a static metric) against a link failure risk must include a link which doesn't free up any bandwidth. Formally:*
$\forall r \in E, \forall \pi \in Bp_r^{(s,t)}, \exists \lambda \in \pi : L_r^{\lambda} = 0$

Proof. To free up bandwidth on a link λ upon failure of link $plr - p_1$[8], λ must belong to at least one shortest primary path traversing link $plr - p_1$ in one direction (from plr to p_1 or from p_1 to plr). In addition, link λ must be located on the downstream of link $plr - p_1$.

Let us define $Down_{(plr,p_1)}$ as a set of nodes located downstream to $plr \rightarrow p_1$ (in this direction) on the primary paths traversing plr and p_1 (see Figure 3). Here we prove that $Down_{(plr,p_1)} \cap Down_{(p_1,plr)} = \varnothing$.

Assume that there is a node b_i so that $b_i \in Down_{(plr,p_1)}$. This means that:

$$w(plr, b_i) < w(p_1, b_i) \tag{7}$$

Where $w(plr, b_i)$ is the weight of any shortest path connecting node plr to node b_i and $w(plr, b_i)$ is the weight of any shortest path connecting node p_1 to node b_i.

[8] Even if we consider that the PLR plr is not adjacent the failed link $p_f - p_i$ (case of the global protection), we can easily prove the correctness of Lemma 4.1. Indeed, if Lemma 4.1 is valid for any backup path $p_f \rightarrow .. \rightarrow p_{j_{j \geq i}}$ protecting against the failure risk $p_f - p_i$, it implies that it is valid for any backup path $p_f \rightarrow .. \rightarrow plr \rightarrow .. \rightarrow p_{j_{j \geq i}}$ protecting against the failure risk $p_f - p_i$. As all the links of the sub-path $p_f \rightarrow .. \rightarrow plr$ can free up bandwidth upon a failure of link $p_f - p_i$, we conclude that Lemma 4.1 is valid for any backup path $plr \rightarrow .. \rightarrow p_{j_{j \geq i}}$ protecting against the failure risk $p_f - p_i$.

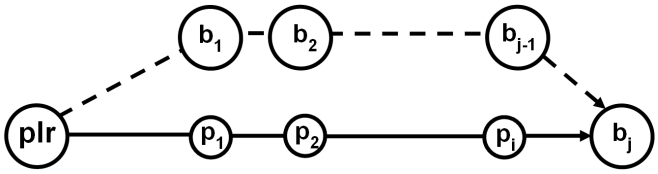


Figure 3. Links forming primary and backup paths

Actually, assume that $b_i \in Down_{(p_1,plr)}$. This means that:

$$w(p_1,b_i) < w(plr,b_i) \tag{8}$$

From formulas (7) and (8), we conclude that $b_i \in Down_{(plr,p_1)} \cap Down_{(p_1,plr)}$ leads to the following contradiction: $w(plr,b_i) < w(p_1,b_i) < w(plr,b_i)$. As a result, we deduce that:
$Down_{(plr,p_1)} \cap Down_{(p_1,plr)} = \varnothing$

As the PLR node belongs to $Down_{(p_1,plr)}$ whereas the merge node is in $Down_{(plr,p_1)}$, no backup path connecting the PLR to the merge point could be established. Indeed, any link connecting a node in $Down_{(p_1,plr)}$ to a node in $Down_{(plr,p_1)}$ cannot free up bandwidth (by definition, the extremity nodes of a link freeing bandwidth are in the same set $Down_{(plr,p_1)}$ or $Down_{(p_1,plr)}$).

Lemma 4.2 *Any backup path protecting a primary shortest path (according to a static metric) against a node failure risk must include a link which doesn't free up any bandwidth. Formally:* $\forall r \in V, \forall \pi \in Bp_r^{(s,t)}, \exists \lambda \in \pi : L_r^{\lambda} = 0$

Proof. We prove the validity of lemma 4.2 by contradiction. In other words, if such a backup path exists, it must be shorter than the primary path it protects.
Assume that there is one backup path $b = plr \rightarrow b_1 \rightarrow .. \rightarrow b_j$ (see Figure3) composed of only links freeing up bandwidth after the failure of node p_1 (downstream to the PLR node plr). The backup path b protects a primary shortest sub-path $p = plr \rightarrow p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j$ according to the static metric $\overrightarrow{w}$ (see Figure 3). Let us prove by induction on the k^{th} backup nodes that:

$$\begin{array}{l} \forall k \leq j,\ \exists s \in Paths^{(plr,b_k)}, \forall \pi \in Paths^{(p_1,b_k)} : \\ w(\pi) \geq w(p_1 \curvearrowright plr \rightarrow s \rightarrow b_k) \end{array} \tag{9}$$

where $p_1 \curvearrowright plr$ is any shortest path from p_1 to plr.
$\underline{k=1}$
To free up bandwidth upon failure of node p_1 (see Figure 3), link $plr \rightarrow b_1$ must belong to at least one shortest primary path traversing node p_1. In addition, link $plr \rightarrow b_1$ must be located on the downstream of node p_1. This implies that: $\exists s = plr \rightarrow b_1 \in Paths^{(plr,b_1)}$ so that:
$p_1 \curvearrowright plr \rightarrow s \rightarrow b_1$ is a shortest path
This means that formula (9) is valid for $k = 1$.
$\underline{\text{Step } 1 < k \leq j}$
Assume that formula (9) is valid for $n = \overline{1,k-1}$ and prove that it is valid for $n = k$. To

free up bandwidth upon failure of node p_1, link $b_{k-1} \rightarrow b_k$ must belong to at least one shortest primary path traversing node p_1. In addition, link $b_{k-1} \rightarrow b_k$ must be located on the downstream of node p_1. This implies that:
$\exists s \in Paths^{(p_1,b_{k-1})}, \forall \pi \in Paths^{(p_1,b_k)}$:
$w(\pi) \geq w(p_1 \rightarrow s \rightarrow b_{k-1} \rightarrow b_k)$
As for $n = k-1$, we have:
$\exists s' \in Paths^{(plr,b_{k-1})}, \forall \pi' \in Paths^{(p_1,b_{k-1})} : w(\pi') \geq w(p_1 \curvearrowright plr \rightarrow s' \rightarrow b_{k-1})$, we deduce that (for $\pi' = p_1 \rightarrow s \rightarrow b_{k-1}$):
$w(\pi) \geq w(p_1 \rightarrow s \rightarrow b_{k-1} \rightarrow b_k) = w(p_1 \rightarrow s \rightarrow b_{k-1}) + w(b_{k-1} \rightarrow b_k) \geq w(p_1 \curvearrowright plr \rightarrow s' \rightarrow b_{k-1}) + w(b_{k-1} \rightarrow b_k) = w(p_1 \curvearrowright plr \rightarrow s' \rightarrow b_{k-1} \rightarrow b_k)$
Thus, path $p_1 \curvearrowright plr \rightarrow s'' \rightarrow b_k$ (with $s'' = s' \rightarrow b_{k-1}$ and $s'' \in Paths^{(plr,b_k)}$) is also a shortest path according to the metric $\overrightarrow{w}$. In other words, formula (9) is verified.

To prove the correctness of Lemma 4.2, we show now that formula (9) contradicts the shortness of the primary path $plr \rightarrow p_1 \rightarrow p_i \rightarrow b_j$.

We recall that the primary path $plr \rightarrow p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j$ corresponds to a shortest path. This implies that:
$\forall \pi \in Paths^{(p_1,b_j)} : w(\pi) \geq w(plr \rightarrow p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j)$. Thus, for any segment path $s \in Paths^{(plr,b_j)}$, we have:
$w(plr \rightarrow s \rightarrow b_j) \geq w(plr \rightarrow p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j)$.

On the other hand, formula (9) implies for $k = j$ that:
$\exists s' \in Paths^{(plr,b_j)}\ \forall \pi' \in Paths^{(p_1,b_j)} : w(\pi') \geq w(p_1 \curvearrowright plr \rightarrow s' \rightarrow b_j) = w(p_1 \curvearrowright plr) + w(plr \rightarrow s' \rightarrow b_j) \geq w(p_1 \curvearrowright plr) + w(plr \rightarrow p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j) = w(p_1 \curvearrowright plr \rightarrow p_1) + w(p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j) > w(p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j)$. This leads to a contradiction since for $\pi' = p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j$ ($\pi' \in Paths^{(p_1,b_j)}$), we obtain: $w(\pi') = w(p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j) > w(p_1 \rightarrow .. \rightarrow p_i \rightarrow b_j)$. Thus, formula (9) cannot be verified. In other words, any backup path protecting against a node failure risk must utilize at least one link which cannot free up bandwidth upon that node failure.

Proposition 4.3 *Every backup path (NNHOP LSP or NHOP LSP) should traverse a link that cannot free up bandwidth after the failure of a protected risk.*

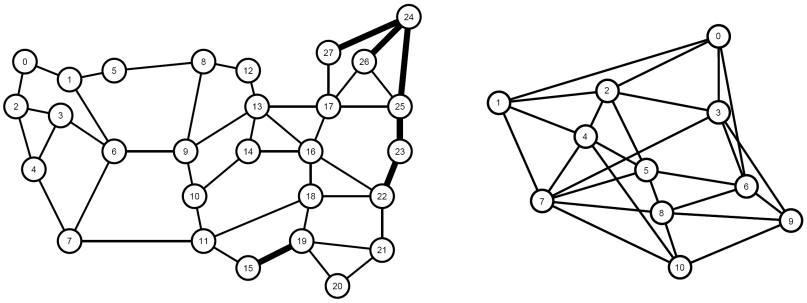
Proof. As both the NHOP and NNHOP paths should protect against link failures[9], we conclude from Lemma 4.1 that every backup path should traverse a link that don't free up bandwidth after a failure of a protected link.

Theorem 4.4 *The number of NHOP and NNHOP backup paths that can be build in a network $G = (V, E, \overrightarrow{w}, \overrightarrow{PC}, \overrightarrow{RC})$ is bounded if $|E|$ and $(RC^\lambda)_{\forall \lambda \in E}$ are bounded (by constants).*

Proof. For the proof, we first show that for any link, the number of backup paths protecting against its failure is bounded. From Lemma 4.1, we know that any backup path protecting against any link failure risk r_l should traverse at least one link λ that cannot free up bandwidth after the failure r_l. From formula (6), we have:
$\delta^\lambda_{r_l} - L^\lambda_{r_l} = \delta^\lambda_{r_l} \leq R^\lambda = \max_r(\delta^\lambda_r - L^\lambda_r, 0) \leq RC^\lambda$

As the protection cost $\delta^\lambda_{r_l}$ corresponds to the number of (1 bandwidth unit) backup paths protecting against the failure r_l and traversing link λ, we deduce that this num-

[9] This is due to the difficulty to distinguish link and node failures.



(a) Long Haul (b) Cost 239

Figure 4. Network topologies

ber of backup paths is bounded by RC^{λ}. Because the number of links freeing up some bandwidth after the failure of r_l is lower than $|E|$ (it is always equal to 0 if we apply the backup bandwidth sharing strategy), we conclude that the number of backup paths protecting against the link failure risk r_l is bounded by $\sum_{\lambda \in E} RC^{\lambda} \leq |E| \times \max_{\lambda \in E}(RC^{\lambda})$.

Similarly, we deduce that the maximum number of backup paths that can be built in the network is bounded by $|E|^2 \times \max_{\lambda \in E}(RC^{\lambda})$ since the number of distinct link failure risks is lower or equal to $|E|$.

Interpretation:

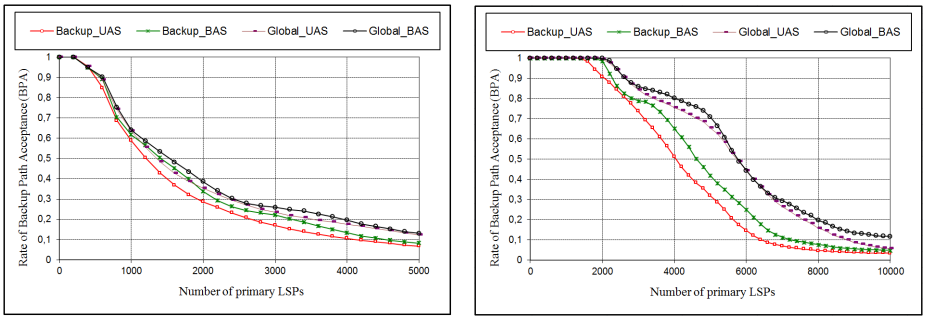
- With both the global and backup bandwidth sharing strategies, the number of backup paths is bounded when the protection capacities (or the link capacities) are bounded and lower than given constants. As any backup path should traverse at least one link that don't free up bandwidth, the use of the global bandwidth sharing strategy instead of the backup bandwidth sharing strategy could not avoid network redimensioning over the long term.
- When a great amount of traffic is not protected (for instance, best-effort traffic does not require protection), the freed bandwidth on some links upon failure could be high. Even in this case, the maximum number of backup paths is bounded specifically by the capacity of links that cannot free up bandwidth.

Whereas the maximum number of backup paths depends on all the network links with the backup bandwidth sharing strategy, this number depends more on the links that cannot free up bandwidth with the global bandwidth sharing strategy. In the next section, we compare by simulations these two bandwidth sharing strategies to quantify the gain in performances due to the exploitation of the freed bandwidth.

5. Performance Evaluation

5.1. Simulation Model

To compare and measure the performances of global and backup bandwidth sharing strategies, we used two well known topologies of network: Long Haul and Cost 239. The first network topology, depicted in Figure 4 (a), is composed of 28 nodes and 45 bidi-



(a) Long Haul (b) Cost 239

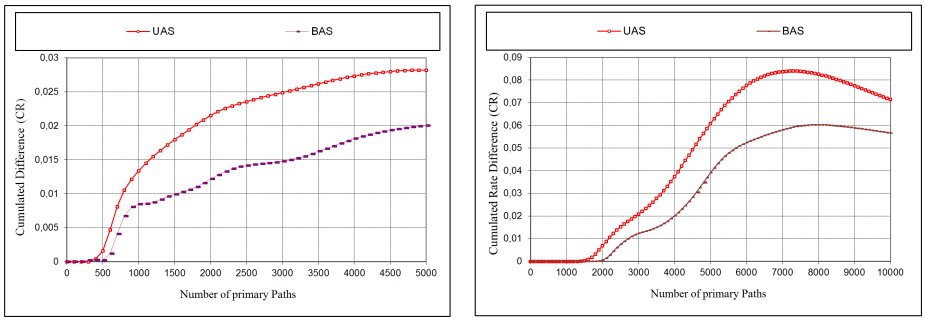
Figure 5. Evolution of the mean rate of backup path acceptance

rectional links. The protection capacities are equal to 600 units in each direction for the bold links and 200 units for the light links. This network topology is relatively wide and presents a mean connectivity degree of 3.21. The second network topology, depicted in Figure 4 (b), is composed of 11 nodes and 26 bidirectional links. It is small and strongly connected since its mean connectivity degree is equal to 4.73. All the links of this network have the same protection capacity that is equal to 200 units in each direction.

To take into account the two possible models of bandwidth allocation (unidirectional bandwidth allocation and bidirectional bandwidth allocation), we considered two test scenarios: unidirectional allocation-based scenario (*UAS*) and bidirectional allocation-based scenario (*BAS*). In the first test scenario, the unidirectional bandwidth allocation method is applied for bandwidth allocation. It means that two protection pools are associated to each bidirectional link in Figure 4. In the second test scenario, the bidirectional bandwidth allocation method is applied for bandwidth allocation. It means that only one protection pool is associated to each bidirectional link in Figure 4. Thus, the protection capacities of bold links are equal to 1200 units (600×2) whereas they are equal to 400 units (200×2) on the light links.

In our simulations, we generated sequentially 1000 demands of primary path protection asking for bandwidth quantities uniformly distributed between 1 and 10 units. Each demand corresponds to one primary path establishment request that is always satisfied (i.e., we assumed that the primary pool capacities of links are sufficient to satisfy all the requests of primary path establishment) and several requests of backup path establishment allowing the protection of the built primary path. The source and target nodes of each primary path are selected uniformly among the set of network nodes. For the computation of primary paths, we applied the shortest path first (SPF) algorithm that optimizes the number of hops whereas we used the constrained shortest path first (CSPF) algorithm for the computation of backup paths. With the backup resource sharing strategy, a request of backup path establishment is satisfied iff equation (5) is verified. With the global resource sharing strategy, equation (6) must be verified to establish the requested backup path.

Two criteria are selected to compare the global and backup resource sharing strategies: acceptance rate of backup paths (*BPA*) and rate of protection bandwidth utilization (*PBwU*). The first criterion *BPA* is computed for different network loads. It corresponds to the instantaneous ratio between the number of backup path requests that are accepted and the total number of backup paths required to protect entirely the last 50 primary



(a) Long Haul (b) Cost 239

Figure 6. Evolution of the difference in acceptance rates of backup paths

paths. Formally, it is determined as follows:
$BPR = \frac{\texttt{\#accepted protection requests}}{\texttt{\#protection requests}}$
Note that a backup path is accepted if and only if there are enough resources.
The second criterion *PBwU* determines and measures the efficiency of bandwidth sharing. It corresponds to the ratio between the sum of all the protection costs and the amount of the bandwidth allocated in the network for the protection. Formally, it is computed as follows:
$PBwU = \sum_{(\lambda,r)\in E\times(V\cup E)} \delta_r^\lambda \,/\, \sum_{\lambda\in E} R^\lambda.$

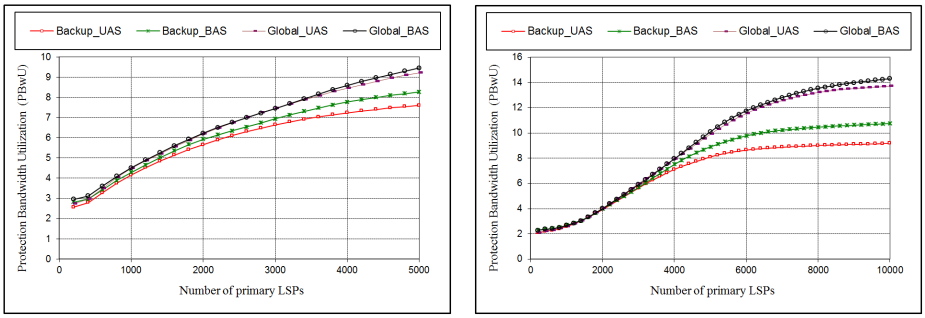
For each test scenario (*UAS* and *BAS*) and at each establishment of 50 primary paths, the two metrics *BPR* and *PBwU* are computed for the two compared strategies. We note that our results correspond to mean values over 1000 experiments.

5.2. Results and Analysis

Figure 5 depicts the evolution of the *instantaneous* acceptance rate of backup paths (BPA) as a function of the number of primary paths setup in the network for the unidirectional and bidirectional bandwidth allocations. We recall that an instantaneous acceptance rate concerns the 50 last primary paths only. Figure 5 shows that the bidirectional bandwidth allocation method is slightly better that the unidirectional bandwidth allocation method.

In addition, Figure 5 clearly shows that the global and backup bandwidth sharing strategies using the bidirectional bandwidth allocation method have respectively larger acceptance rates than the global and backup bandwidth sharing strategies using the unidirectional bandwidth allocation method. These observations can be explained by the distribution of the protection costs on links (especially on opposite links) which is heterogeneous [4].

Figure 6 depicts the difference in *cumulated* acceptance rates of backup paths for the unidirectional and bidirectional bandwidth allocations. It shows that the difference is small and even imperceptible sometimes. For instance, in Longhaul network topology, the difference in instantaneous acceptance rates does not exceed 8 %, even for high loads of traffic (a large number of primary paths) where the instantaneous acceptance rate of backup paths is small and inefficient (see Figure 5 (a)). For usual instantaneous acceptance rates that should be larger than 90 %, the difference between the compared strate-



(a) Long Haul (b) Cost 239

Figure 7. Evolution of the mean rate of protection bandwidth utilization

gies is often imperceptible. With regards to the cumulated acceptance rate, Figure 6 (a) shows that the difference is very low and smaller than 3 % in Longhaul network topology. In Cost 239 network topology, the differences in instantaneous acceptance rates reaches 30 % (see Figure 5 (b)) for high loads whereas it does not exceed 9 % for the cumulated rates (see Figure 5 (b)).

Obviously, the difference in the acceptance rates of backup paths is directly related to the amount and distribution of the freed bandwidth on links. Since the freed bandwidth is statically high on the links close to PLRs and generally low on the links located far from PLRs, the difference in acceptance rates of the compared strategies is slightly higher in COST 239 network topology than in Longhaul network topology. Indeed, the links are closer to the PLRs in COST 239 since it is more homogeneous and it has a larger connectivity degree than Longhaul.

In addition of the previous observations, we note that even for high freed bandwidth values, the acceptance rates of backup paths decrease with the augmentation of the traffic load and they converge to the saturation state where almost all the new protection requests are rejected. This corroborates our theoretical results which announces the existence of an upper bound for the number of backup paths that can be established in the network even with unlimited resources.

With regards to the second metric (bandwidth sharing utilization), Figure 7 shows that both the global and backup bandwidth sharing strategies have similar bandwidth utilization rates for small and usual acceptance rates of backup paths. For instance, the difference in bandwidth sharing utilization for the compared strategies is very small in Longhaul network (see Figure 7 (a)) when the number of primary paths is lower than 1000 (all the acceptance rates are larger than 0.7) whereas the difference is imperceptible in COST 239 network (see Figure 7 (b)) when the number of primary paths is lower than 3000 (all the acceptance rates are larger than the usual value 0.85). For high traffic loads, Figure 7 shows that the global bandwidth sharing strategy is better than the backup bandwidth sharing strategy. This is essentially due to the amount of freed bandwidth which increases with the decrease of the acceptance rate of backup paths. Indeed, whereas the protection bandwidth is completely independent of the freed bandwidth variation when the backup bandwidth sharing strategy is applied, it decreases with the augmentation of the freed bandwidth when we apply the global bandwidth sharing strategy.

To summarize, these simulations show that the difference in performances between the global and backup bandwidth sharing strategies is almost imperceptible for low traf-

fic loads where the acceptance rate of backup paths is high and usual. For high traffic loads where the acceptance rate of backup paths is low, the global bandwidth sharing strategy is slightly better than the backup bandwidth sharing strategy. In addition to the precedent remarks, our simulations comfort our theoretical results (see Therem 4.4) and show clearly that the freed primary bandwidth has very slight effect on the acceptance rate of backup paths compared to the backup bandwidth sharing.

6. Conclusion

Two known strategies of resource (bandwidth) sharing are described in this paper: backup bandwidth sharing and global bandwidth sharing. The first strategy restricts the bandwidth sharing to the backup paths whereas the second strategy extends the bandwidth sharing to the primary and backup paths.

To quantify the gain due to the extension of the bandwidth sharing to the primary and backup paths, we firstly proved theoretically that the bandwidth sharing between the primary and backup paths can never be applied on some backup links when the primary paths correspond to the shortest ones according to a static metric. Thus, the acceptance rate of backup paths is always limited and bounded by the protection capacities of links. Secondly, to measure the enhancement due to the bandwidth sharing between the primary and backup paths, we showed by simulations that the gain in performances (acceptance rate of backup paths and bandwidth utilization) is often imperceptible, particularly for low traffic loads where the acceptance rate of backup paths is large and usual. For high traffic loads where the acceptance rates are small, the global bandwidth sharing strategy outperforms slightly the backup bandwidth sharing strategy, especially in strongly connected networks.

As a result, the global bandwidth sharing strategy cannot be a long term solution for supporting bandwidth-intensive applications especially since the global bandwidth sharing strategy induces an overcost. Indeed, in return of the slight performance improvements the global bandwidth sharing allows, we note the complication of path computation and the necessity to maintain larger information. For instance, additional computations should be done with the global bandwidth sharing strategy to determine the amount of freed bandwidth after each establishment or liberation of a primary path.

References

[1] S. Balon, L. Mélon, and G. Leduc, "A Scalable and Decentralized Fast-Rerouting Scheme with Efficient Bandwidth Sharing," *Computer Networks*, vol. 50, pp. 3043–3063, November 2006.

[2] M. S. Kodialam and T. V. Lakshman, "Dynamic Routing of Restorable Bandwidth-Guaranteed Tunnels using Aggregated Network Resource Usage Information," *IEEE/ACM Transactions On Networking*, vol. 11, pp. 399–410, June 2003.

[3] P. Pan, G. Swallow, and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels." RFC 4090, May 2005.

[4] M. Y. Saidi, B. Cousin, and J. L. Le Roux, "PLR-based Heuristic for Backup Path Computation in MPLS Networks," *Computer Networks*, vol. 53, pp. 1467–1479, June 2009.

[5] J. P. Vasseur, A. Charny, F. Le Faucheur, J. Achirica, and J. L. Le Roux, "Framework for PCE-based MPLS-TE Fast Reroute Backup Path Computation," Internet Draft draft-leroux-pce-backup-comp-frwk-00.txt, IETF, July 2004.

[6] G. Malkin, "RIP Version 2." RFC 2453, November 1998.

[7] J. Moy, "OSPF Version 2." RFC 2328, April 1998.

[8] X. Cheng, S. Su, Z. Zhang, H. Wang, F. Yang, Y. Luo, and J. Wang, "Virtual Network Embedding Through Topology-Aware Node Ranking," *SIGCOMM Comput. Commun. Rev.*, vol. 41, pp. 38–47, April 2011.

[9] W. Gang, Z. Zhenmin, L. Zhaoming, T. Yi, and W. Xiangming, "A Virtual Network Embedding Algorithm based on Mapping Tree," in *13th International Symposium on Communications and Information Technologies (ISCIT)*, 2013.

[10] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture." RFC 3031, January 2001.

[11] V. Sharma and F. Hellstrand, "Framework for Multi-Protocol Label Switching (MPLS)-based Recovery." RFC 3469, February 2003.

[12] K. Kompella and Y. Rekhter, "Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)." RFC 4202, October 2005.

[13] M. Y. Saidi, B. Cousin, and J. L. Le Roux, "Using Shared Risk Link Groups to Enhance Backup Path Computation," *Computer Networks*, vol. 53, pp. 1341–1353, June 2009.

[14] M. R. Rahman and R. Boutaba, "SVNE: Survivable Virtual Network Embedding Algorithms for Network Virtualization," *IEEE Transactions on Network and Service Management*, vol. 10, no. 2, pp. 105–118, 2013.

[15] H. Yu, V. Anand, and C. Qiao, "Virtual Infrastructure Design for Surviving Physical Link Failures," *The Computure Journal*, vol. 55, no. 8, pp. 965–978, 2012.

[16] T. Guo, N. Wang, K. Moessner, and R. Tafazolli, "Shared Backup Network Provision for Virtual Network Embedding," in *IEEE International Conference on Communications (ICC)*, pp. 1–5, 2011.

[17] A. E. Kamal, "1+n Network Protection for Mesh Networks: Network coding-based protection using p-cycles," *IEEE/ACM Transactions on Networking*, vol. 18, no. 1, pp. 67–80, 2010.

[18] A. E. Kamal and M. Mohandespour, "Network Coding-based Protection," *Optical Switching and Networking*, vol. 11, no. B, pp. 189–201, 2014.

[19] F. Palmieria, U. Fioreb, S. Ricciardic, and A. Castiglioned, "GRASP-based Resource Re-optimization for Effective Big Data Access in Federated Clouds," *Future Generation Computer Systems*, vol. 54, pp. 168–179, 2016.

[20] P. Cholda, A. Mykkeltveit, B. E. Helvik, O. Wittner, and A. Jajszczyk, "A Survey of Resilience Differentiation Frameworks in Communication Networks," *IEEE Communications Surveys and Tutorials*, vol. 9, no. 1-4, pp. 32–55, 2007.

[21] E. Calle, J. L. Marzo, and A. Urra, "Protection Performance Components in MPLS Networks," *Computer Communications*, vol. 27, no. 9, pp. 1220–1228, 2004.

[22] M. Y. Hariyawan, "Comparison Analysis Of Recovery Mechanism At MPLS Network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 1, no. 2, pp. 151–160, 2011.

[23] R. Bhandari, *Survivable Networks : Algorithms for Diverse Routing*. Kluwer Academic Publishers, 1999.

[24] L. Mélon, F. Blanchy, and G. Leduc, "Decentralized Local Backup LSP Calculation with Efficient Bandwidth Sharing," in *Proceedings of 10th International Conference on Telecommunications (ICT'2003)*, February 2003.

[25] J. P. Vasseur, M. Pickavet, and P. Demeester, *Network Recovery: Protection and Restoration of Optical, SONET-SDH, IP, and MPLS*. Morgan Kaufmann Publishers, 2004.

[26] L. Li, M. M. Buddhikot, C. Chekuri, and K. Guo, "Routing Bandwidth Guaranteed Paths with Local Restoration in Label Switched Networks," in *10th IEEE International Conference on Network Protocols (ICNP'02)*, p. 110, 2002.

[27] C. C. Meixner, C. Develder, M. Tornatore, and B. Mukherjee, "A Survey on Resiliency Techniques in Cloud Computing Infrastructures and Applications," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 3, pp. 2244–2281, 2016.

[28] S. Herker, A. Khan, and X. An, "Survey on Survivable Virtual Network Embedding Problem and Solutions," in *The 9th International Conference on Networking and Services (ICNS)*, pp. 99–104, 2013.

[29] B. Guo, C. Qiao, J. Wang, H. Yu, Y. Zuo, J. Li, Z. Chen, and Y. He, "Survivable Virtual Network Design and Embedding to Survive a Facility Node Failure," *Journal of Lightwave Technology*, vol. 32, no. 3, pp. 483–493, 2014.

[30] Y. Wang, X. Liu, X. Qiu, and W. Li, "Prediction-based Survivable Virtual Network Mapping against Disaster Failures," *International Journal of Network Management*, vol. 26, pp. 336–354, 2016.

[31] M. Y. Saidi and B. Cousin, "Resource Saving: Which Resource Sharing Strategy to Protect Primary Shortest Paths?" in *13th IEEE Annual Consumer Communications & Networking Conference, CCNC 2016, Las Vegas, NV, USA, January 9-12, 2016*, pp. 297–298, January 2016.

[32] W. Grover and D. Stamatelakis, "Cycle-Oriented Distributed Preconfiguration: Ring-like Speed with Mesh-like Capacity for Self-planning Network Restoration," *In Proc. International Conference on Communications*, 1998.

[33] S. Kini, K. Kodialam, T. V. Lakshman, S. Sengupta, and C. Villamizar, "Shared Backup Label Switched Path Restoration," Internet Draft draft-kini-restoration-shared-backup-01.txt, IETF, May 2001.