PLR-based heuristic for backup path computation in MPLS networks

Mohand Yazid Saidi, Bernard Cousin, Jean-Louis Le Roux

IRISA/INRIA, Université de Rennes 1 – Campus de Beaulieu, 35042 Rennes, France
IRISA, Université de Rennes 1 – Campus de Beaulieu, 35042 Rennes, France
France Télécom, 2 Avenue Pierre Marzin, 22300 Lannion, France

ARTICLE INFO

Article history:
Received 15 November 2007
Received in revised form 14 November 2008
Accepted 26 January 2009
Available online xxxx

Responsible Editor: R. LoCigno

Keywords:
Recovery
Local protection
Backup LSP
Failure risk
SRLG
MPLS
Bandwidth sharing
Path computation
Network

ABSTRACT

To ensure service continuity in networks, local protection pre-configuring the backup paths is preferred to global protection. Under the practical hypothesis of single physical failures in the network, the backup paths which protect against different logical failure risks (node, link and shared risk link group (SRLG)) cannot be active at the same time. Thus, sharing bandwidth between such backup paths is crucial to increase the bandwidth availability.

In this article, we focus on the optimal on-line distributed computation of the bandwidth-guaranteed backup paths in MPLS networks. As the requests for connection establishment and release arrive dynamically without knowledge of future arrivals, we choose to use the on-line mode to avoid LSP reconfigurations. We also selected a distributed computation to offer scalability and decrease the LSP setup time. Finally, the optimization of bandwidth utilization can be achieved thanks to the flexibility of the path choice offered by MPLS and to the bandwidth sharing.

For a good bandwidth sharing, the backup path computation entities (BPCEs) require the knowledge and maintenance of a great quantity of bandwidth information (e.g. non-aggregated link information or per path information) which is undesirable in distributed environments. To get around this problem, we propose here a PLR (point of local repair)-based heuristic (PLRH) which aggregates and noticeably decreases the size of the bandwidth information advertised in the network while offering a high bandwidth sharing. PLRH permits an efficient computation of backup paths. It is scalable, easy to be deployed and balances equitably computations on the network nodes.

Simulations show that with the transmission of a small quantity of aggregated information per link, the ratio of rejected backup paths is low and close to the optimum.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

The proactive protection of communication becomes increasingly important with the explosion of the number of network real time applications (voice over IP, network games, video on demand, etc). Thus, to ensure network service continuity upon a failure, the proactive protection techniques [1,2] precompute and generally pre-establish backup paths capable to receive and reroute the traffic of the affected primary paths. Two schemes of protection exist: global (end-to-end) and local. With global scheme [1], each primary path is protected by one vertex (or link) disjoint backup path interconnecting the primary source and destination nodes. This protection scheme presents the disadvantage of increasing the recovery cycle [3] since it requires that failure notification reaches the source before the switching from the primary toward the backup path. This last drawback is eliminated with the use of local protection where the recovery is achieved locally and without any control plane notification by the upstream node to the failing component.

With the advent of MPLS [4] in the last decade, the local protection is provided in an efficient manner. In fact, MPLS offers a great flexibility for choosing paths (called label
switched paths or LSPs) and thus, the backup paths can be
determined so that bandwidth availability is maximized.
Two types of backup LSP are defined for MPLS local protec-
tion [5]: Next HOP (NHOP) LSP and Next HOP (NNHOP)
LSP. A NHOP LSP (resp. NNHOP LSP) is a backup path pro-
tecting against link failure (resp. node failure); it is set up
between a primary node called point of local repair (PLR)
and one primary node downstream to the PLR (resp. to
the PLR next hop) called merge point (MP). Such backup
LSP bypasses the link (resp. the node) downstream to the
PLR on the primary LSP. When a link failure (resp. node
failure) is detected by a node, this later activates locally
all its NHOP and NNHOP (resp. all its NNHOP) backup
LSPs by switching traffic from the affected primary LSPs
to their backup LSPs.

To ensure enough resource (particularly the bandwidth)
after the recovery from a failure, the backup LSPs must re-
serve the resources they need beforehand. In this way, if
we consider that each backup path has its own exclusive
resources, the network will be overbooked rapidly since
the available resources decrease quickly. Instead and under
the practical single failure assumption, resource utilization
can be improved by sharing the resources as much as pos-
able between the backup LSPs. For instance, all the backup
LSPs protecting against different failure risks (a risk is
formed of the network components which can fail simulta-
nously) can share their resource allocation on their com-
mon links. Indeed, such backup paths cannot use their
resources simultaneously since they cannot be active at
the same time (there is at most one failure occurrence at
any time).

To increase the number of LSPs that can be setup in a
network (i.e. to decrease the blocking probability), the res-
source sharing should be taken into account when the
backup LSPs are computed. Three functionalities are ne-
necessary to perform such computations in a distributed envi-
ronment: information collection, information distribution
and path determination. The first functionally gathers the
structures and properties of the backup LSPs setup in the
network. In practice, each network node stores the path
links, the bandwidth and the risks protected by the backup
LSPs traversing it. Such information can be obtained easily
and without any additional overhead when the backup
LSPs are signaled as in [5]. The second functionality reorga-
nizes and transmits the collected information to nodes
supporting the BPCEs (backup path computation entities).
We note that for a same capability of bandwidth sharing,
the same time (there is at most one failure occurrence at
any time).

To deal with any single physical failure in a logical
(MPLS) layer, three types of (logical) failure risks are de-
defined: link, node and SRLG. The first type of failure risk cor-
responds to the risk of a logical link failure due to the
breakdown of an exclusive physical component of the log-
ic link. The second type of failure risk corresponds to the
risk of a logical node failure due to the breakdown of an
exclusive physical component of the logical node. Finally,
the third type of risk corresponds to failure risk of a com-
mon physical component (optical fiber, crossconnect, etc.)
shared by a group of logical links [6].

In Fig. 1, two topologies corresponding to the same net-
work are depicted. The first one (Fig. 1a) is obtained
according to the data link neighbourhood information;
the second one (Fig. 1b) is determined with the use of only
the (IP) Network neighbourhood information. As we see,
the optical crossconnect OXC in Fig. 1a is not visible by
the IP (and MPLS) layer. This crossconnect is an optical
component used to connect router E to routers B and D.
Hence, the network link E-B (resp. link E-D) in Fig. 1b
corresponds to the optical path E-OXC-B (resp. optical path E-
OXC-D) in Fig. 1a. As a result, the two IP (MPLS) links E-B
and E-D should be gathered in one SRLG risk1 to cope with

1 We note that the IGP-TE protocols (OSPF-TE [7] and ISIS-TE [8]) are
extended to transmit the structures of the SRLGs.
the failure of the crossconnect OXC. Similarly, to protect against the failure of a physical link A-B (resp. physical node A) for instance in MPLS layer, one (MPLS) failure risk A-B (resp. A) of type link (resp. node) must be defined.

In order to ensure enough bandwidth upon a failure, minimal quantities of bandwidth must be reserved on links. To determine such quantities, [9] defines two concepts: the protection failure risk group (PFRG) and the protection cost. The PFRG of a given arc \( \lambda \), noted PFRG(\( \lambda \)), corresponds to a set which includes all the risks protected by the backup LSPs traversing the arc \( \lambda \). The protection cost of a risk \( r \) on an arc \( \lambda \), noted \( \delta_r^\lambda \), is defined as the cumulative bandwidth of the backup LSPS which will be activated on the arc \( \lambda \) upon a failure of the risk \( r \). For a SRLG risk \( srlg \) composed of links \( \{l_1, l_2, \ldots, l_n\} \), the protection cost on an arc \( \lambda \) is determined as follows:

\[
\delta_{srlg}^\lambda = \sum_{0 \leq i < n} \delta_{l_i}^\lambda
\]

To cope with any single failure, a minimal quantity of protection bandwidth \( G_{\lambda} \) must be reserved on the arc \( \lambda \). Such quantity \( G_{\lambda} \) is determined as the maximum of the protection costs on the arc \( \lambda \).

\[
G_{\lambda} = \max_{r \in \text{PFRG}(\lambda)} \delta_{r}^\lambda
\]

In order to control and specify the quantity of bandwidth dedicated for protection and to separate the task of primary LSP computation from that of backup LSP computation, the bandwidth capacity \( C_{\lambda} \) on an arc \( \lambda \) can be divided in two pools: primary bandwidth pool and protection bandwidth pool (Fig. 2). The primary bandwidth pool on an arc \( \lambda \) has a capacity \( PC_{\lambda} \) and it is used to allocate bandwidth for primary LSPs. The protection bandwidth pool on an arc \( \lambda \) has a capacity \( BC_{\lambda} \) and it is used to allocate bandwidth for backup LSPs. We note that the separation of the bandwidth in two pools is not necessary to apply all the backup LSP computation techniques described in this article (except that described in Ref. [10]). It is adopted here to limit the amount of bandwidth used for protection and thus increase the bandwidth sharing.

To ensure the respect of bandwidth constraints upon a failure, the minimal protection bandwidth reserved on each arc \( \lambda \) must verify:

\[
G_{\lambda} \leq BC_{\lambda}
\]

To keep (3) valid after the setup of a backup LSP \( b \) of bandwidth \( bw(b) \) and protecting against the risks in \( FR(b) \) (\( FR(b) \) is a set composed of all the risks whose failure activates the backup path \( b \) ), only the arcs \( \lambda \) verifying the following inequality can be selected to be in the LSP \( b \):

\[
\max_{r \in \text{FR}(b)} \delta_r^\lambda \leq BC_{\lambda} - bw(b)
\]

Finally, we define the residual protection bandwidth \( RB_{\lambda} \), as the amount of protection bandwidth which is not used on an arc \( \lambda \). It corresponds to:

\[
RB_{\lambda} = BC_{\lambda} - G_{\lambda}
\]

### 3. Related work

Recently, a great deal of work is addressing the path computation in order to find algorithms and mechanisms allowing on-line computation of the optimized backup paths. Several solutions are then proposed but a large number of them, like [11,12], deals only with failure risks of type link or node.

In this section, we will be more interested on computation techniques (especially, on bandwidth information distribution methods) of bandwidth-guaranteed backup LSPS which deal with all the types of failure risks (SRLG, link and node). According to the computation environment, we distinguish two types of techniques: centralized and distributed.

In a centralized environment, the server can memorize all the information (topology, structures and properties of all the backup paths) necessary to compute the optimized backup paths. Such information is obtained from the IGP protocols and from the paths computed by the server itself. Obviously, with complete information knowledge, the bandwidth sharing capabilities can be exploited efficiently to compute the optimized backup paths. For instance, [13] uses an ILP (integer linear programming) formulation to compute a primary path and its backup path (with the use of the end-to-end protection) so that the additional bandwidth they need is minimal. Although the high-quality of bandwidth sharing obtained with centralized servers, their utilization can increases significantly the LSP setup time after the occurrence of a failure and present some well known disadvantages like the formation of bottlenecks around the server and the sensitivity to the failure or overload of the server.

Instead and to avoid long LSP setup time upon a failure, distributed techniques are preferred to centralized techniques. As the quality of the distributed techniques computing the backup paths depends closely on the algorithms implementing the functionality of information distribution (cf. Section 1), we will focus below on the study of these algorithms; for path computation, various variants of the Dijkstra’s algorithm or ILP formulation can be applied.

In a first obvious solution [14], Kini proposes to flood within the IGP-TE (interior gateway protocol traffic engineering) protocols the topology information, the primary bandwidth, the capacities and all the protection costs of the risks on the topology arcs in the network. In this way, each node has a complete knowledge of the information necessary to the backup LSP computation and as a result, it can use a similar model as in the centralized environment to perform the computations. This computation technique increases the bandwidth availability but it overloads the network with large and frequent messages advertising the protection costs (the number of protection costs advertised for an arc maybe large and up to the
number of failure risks in the network). Another similar solution for backup LSP computation is proposed in Ref. [9]. To decrease the size of messages advertising the bandwidth information, [9] suggests to flood within IGP-TE protocols the structures and properties of the backup LSPs. To reduce the advertisement frequency, [9] recommends the use of the facility backup protection (described in [5]). Like the first computation technique, this second technique floods the network with the transmission of a large quantity of information advertising the structures and properties of the backup LSPs. To scale well, [10] proposed the PCE-based MPLS-TE fast reroute technique in which no control message is necessary to compute the bandwidth-guaranteed backup LSPs. With this technique, a separate PCE (path computation element) is associated with each failure risk in order to compute the backup LSPs which will be activated at the failure of that risk. This computation technique is efficient when there are no SRLGs in the network. Otherwise, the PCE-based MPLS-TE fast reroute technique requires a mechanism distinguishing a node failure from a link failure (as [15]). This increases significantly the recovery cycle of all the communications. In addition, the PCE-based MPLS-TE fast reroute technique requires that non disjoint SRLGs be managed by a same PCE. This centralizes the path computations and introduces same problems as that encountered in the centralized environments. To get around the drawbacks of the PCE-based MPLS-TE fast reroute technique, [16] proposed to share the protection costs of each SRLG between all the end nodes of that SRLG. This last backup LSP computation technique balances equitably the computations on the network nodes but it introduces a new drawback which consists in the size increase of the control messages.

To offer scalability without increasing the recovery cycle, new computation heuristics which approximate and reduce the bandwidth information (protection costs especially) transmitted in the network have emerged. Hence, once the bandwidth information is collected, nodes aggregate it before its flooding in the network.

In a first computation heuristic (residual bandwidth-based heuristic) presented in Ref. [14], Kini proposes to approximate the protection cost \( \delta^i \) of a risk \( r \) on a (unidirectional) link \( \lambda \) by the maximum of protection costs \( \text{Max}_r(\delta^i) \) on that link. In this way, only one aggregated value per link is advertised in the network. This heuristic has the advantage of facility of its deployment. Indeed, this requires only slight modifications to IGP-TE protocols [7,8] for the advertisement of the minimal quantities of protection bandwidth on links. However, this heuristic does not exploit efficiently the bandwidth sharing. As a result, the number of backup LSPs that can be built with this heuristic is low (i.e. the blocking probability is high).

In order to improve the bandwidth sharing and resource availability, the previous heuristic can be enhanced to better estimate the protection costs. Hence, with the improved heuristic of Kini (IKH), the protection cost \( \delta^i \) is approximated by the minimum between the highest protection cost \( \text{Max}_r(\delta^i) \) on the arc \( \lambda \) and the primary bandwidth \( F_r \) reserved on the risk \( r \). In practice, this last heuristic has performances comparable to those of the residual bandwidth-based heuristic since the primary bandwidth \( F_r \) reserved on the risk \( r \) (case of a node) is often greater than the highest protection cost \( \text{Max}_r(\delta^i) \) on the arc \( \lambda \).

4. PLR-based backup path computation heuristic (PLRH)

For simplicity, we consider here that the capacity \( C \) of each arc \( \lambda \) is divided in two disjoint pools (protection pool and primary pool), as in Fig. 2. In this manner, the task which computes the backup LSPs can be independent from that determining the primary LSPs.

4.1. PLRH principles

The PLRH allows an efficient approximation of the protection costs on the links with the advertisement of a small quantity of aggregated protection bandwidth information. It is based on the two following principles:

- An arc \( \lambda \) can be used to establish a new backup LSP \( b \) requiring a quantity of bandwidth \( bw(b) \) if and only if the protection costs of the risks protected by such LSP (on \( \lambda \)) are lower or equal to \( BC_r - bw(b) \).

- Some values of protection cost on an arc can be very low. Aggregate and approximate these values by their maximum which can decrease the quantity of protection information to be advertised in the network with slight or without the deterioration of the bandwidth sharing.

To show how the PLRH exploits the two above principles, let us consider an example. In Fig. 3, the protection pool of an arc \( \lambda \) is illustrated. This protection pool has a capacity \( BC_\lambda \) of 100 units. It was used to allocate the bandwidth for backup paths traversing the arc \( \lambda \) and protecting against failures of seven risks: \( \text{node}_1 \), \( \text{node}_2 \), \( \text{link}_1 \), \( \text{link}_2 \), \( \text{link}_3 \), \( \text{link}_4 \) and \( \text{sr}_{\text{rlg}1} \). The protection costs associated to these risks are as follows: \( \delta^i_{\text{node}_1} = 100 \), \( \delta^i_{\text{node}_2} = 60 \), \( \delta^i_{\text{link}_1} = 75 \), \( \delta^i_{\text{link}_2} = 40 \), \( \delta^i_{\text{link}_3} = 5 \), \( \delta^i_{\text{sr}_{\text{rlg}1}} = 80 \), \( \delta^i_{\text{link}_4} = 80 \).

Algorithm 1

Simplified algorithm run by the outgoing node \( o \) of an arc \( \lambda \).

Please cite this article in press as: M.Y. Saidi et al., PLR-based heuristic for backup path computation in MPLS networks, Comput. Netw. (2009), doi:10.1016/j.comnet.2009.01.009
When the maximal quantity of bandwidth \( \text{max}_{bw} \) \(^2\) that a LSP can claim is known (in Fig. 3, \( \text{max}_{bw} \) is equal to 30), the application of Principle 1 permits to deduce that all the risks whose protection costs are lower or equal to the threshold \( T_s \) (\( T_s = BC_2 - \text{max}_{bw} \)) can be ignored (approximated by zero) when a new backup LSP is computed. In fact, the selection of the arc \( \lambda \) to be in a new backup LSP \( b \) of bandwidth \( bw(b) \) cannot lead to the violation of bandwidth constraints upon a failure of a risk \( r \) if the protection cost \( \delta_r \) is lower or equal to the threshold \( T_s \). This results from the following inequalities:

\[
\begin{align*}
\delta_r & \leq T_s = BC_2 - \text{max}_{bw} \\
 bw(b) - \text{max}_{bw} & \leq 0 \quad \Rightarrow \delta_r + bw(b) \leq BC_2
\end{align*}
\]

Algorithm 2

Simplified algorithm run by each node receiving a \( x_i \)-vector

\[
\text{begin algorithm } \quad \text{for all } [\text{Arc } \rightarrow \text{Bandwidth}: \text{costs}] \quad \text{do} \quad \text{getRiskId()} \quad \text{getCost();} \\
\text{end if} \quad \text{for all } [\text{Arc } \rightarrow \text{Bandwidth}: \text{costs}] \quad \text{do} \quad \text{setRiskId()} \\
\text{end for} \quad \text{end algorithm}
\]

\(^2\) When \( \text{max}_{bw} \) is not known, we can set its value on a link \( \lambda \) to BC, (worst case). Moreover, we note that we can process a protection request claiming a quantity of bandwidth which is higher than \( \text{max}_{bw} \) by splitting it in two or more requests of bandwidth lower than \( \text{max}_{bw} \).

Obviously, the elimination of the protection costs, which are lower or equal to the threshold \( T_s \) (\( T_s = BC_2 - \text{max}_{bw} = 70 \)) from the information to be advertised in the network, does not alter the decision of excluding (or including) the arc \( \lambda \) in a next backup LSP computation. Typically, in Fig. 3, the outgoing node \( o_1 \) to the arc \( \lambda \), which is responsible\(^3\) of the advertisement of the protection costs \( \{\delta_r\} \), on the arc \( \lambda \), approximates the protection costs of \( node_2, link_2 \) and \( link_3 \) by zero. As a result, these risks \( (node_2, link_2 \) and \( link_3) \) and their corresponding protection costs on the arc \( \lambda \) are not advertised in the network.

When the value \( \text{max}_{bw} \) is high (or ignored by the nodes of the network), the quantity of bandwidth information advertised for each arc of the network topology can be high and unacceptable. To avoid the flooding of the network while maintaining bandwidth sharing high, PLRH limits the size of the protection bandwidth information that is advertised for each arc \( \lambda \) to a vector (called \( x_i \)-vector) composed of \( x_i \) elements. Each \( x_i \)-vector component includes a couple of protection cost and its associated risk. Besides, the costs conveyed in the \( x_i \)-vector of an arc \( \lambda \) correspond to the \( x_i \) highest values of protection cost. In this manner, each node receiving a \( x_i \)-vector of an arc \( \lambda \) deduces the \( x_i \) highest protection costs (and their corresponding risks) on the arc \( \lambda \) and approximates all the rest of protection costs by the \( (x_i) \)th highest protection cost (principle 2 of PLRH) on the arc \( \lambda \). For instance, if we consider that \( x_i \) is equal to 2 in Fig. 3, the outgoing node \( o_1 \) to the arc \( \lambda \) will send the following \( x_i \)-vector: \([\text{node}_1, 100], (\text{generic}\_\text{risk}, 80)\) (where \( \text{generic}\_\text{risk} \) refers to any risk different from those conveyed in the \( x_i \)-vector).

4.2. PLRH algorithm description

With the combination of the PLRH principles 1 and 2, we construct the algorithms Alg. 1 and Alg. 2 which specify the steps of the protection cost advertisement and collection. Thus, Alg. 1 describes the procedure of protection cost advertisement which limits the transmitted bandwidth information for each arc \( \lambda \), to a \( x_i \)-vector. This vector

---

\(^3\) The end nodes of an arc \( \lambda \) know all the risks (and their associated protection costs on that arc \( \lambda \)) using \( \lambda \) for protection. This information can be obtained, without overhead, when the backup LSPs are signaled.
contains, at most, the \( x_i \) highest protection costs which are greater than the threshold \( T_{s_i} \). Concerning Alg. 2, it specifies the procedures used to approximate the protection costs from the received \( x_i \)-vector information.

In order to increase the bandwidth sharing and to reduce the size of messages transmitting the \( x_i \)-vectors, each node running Alg. 1, eliminates from its protection cost table all the entries corresponding to the risks which are included in others. In Fig. 3 for instance, the SRLG risk \( srlg_1 \) is made of two link risks: \( link_1 \) and \( link_3 \). As a result, these two risks (\( link_1 \) and \( link_3 \)) and their corresponding protection costs are deleted from all the protection cost tables before any advertisement of the \( x_i \)-vectors. After this step, the outgoing node \( o_i \) to each arc \( \lambda \) builds a list \( cost \), containing all the risks and their protection costs on \( \lambda \). This list is then sorted according to the decreasing values of the protection cost. For the example in Fig. 3, the sorted list \( cost \), is as follows:

\[
\{ (node_1, 100), (srlg_1, 80), (link_2, 80), (node_2, 60), (link_4, 40) \}
\]

At each change in the sorted list \( cost \), node \( o_i \) runs the following instructions (cf. Alg. 1): \( o_i \) extracts from the list \( cost \), the first \( x_i \) couples conveying the protection costs which are greater than the threshold \( T_s \) (\( x_i \) and \( T_s \) are parameters of Alg. 1). Then, \( o_i \) checks if there is a change in the value of the \( x_i \)-vector. If so, it advertises the new \( x_i \)-vector, else it does nothing.

When a node (different from the end nodes of \( \lambda \)) receives a \( x_i \)-vector corresponding to an arc \( \lambda \), it runs the routine shown in Alg. 2 to approximate the protection costs on the arc \( \lambda \).

According to the threshold value \( (T_s) \) and to the \((x_i + 1)\)th highest protection cost (denoted by \( x_{i,plus1} \)-cost) on an arc \( \lambda \), the PLRH decision of excluding the arc \( \lambda \) in the next backup LSP computation can be “sure and correct” or “possibly wrong”. Two areas are defined to measure the correctness degree of the protection cost approximation used by PLRH: doublyd area \( x_{i,plus1} \)-cost \( > T_s \) and sure area \( x_{i,plus1} \)-cost \( \leq T_s \).

4.2.1. Doubtful area \( x_{i,plus1} \)-cost \( > T_s \)

When the \((x_i + 1)\)th highest protection cost on an arc \( \lambda \) is in the doubtful area, the advertisement of a \( x_i \)-vector of a maximal size \( x_i \) can be insufficient to decide without mistake if the arc \( \lambda \) can be selected to be in a new backup LSP.

In Fig. 3 for instance, the \((x_i + 1)\)th highest protection cost on the arc \( \lambda \) is in the doubtful area if \( x_i \leq 2 \). Thus, the advertisement of a \( x_i \)-vector containing at most the two highest protection costs on \( \lambda \) can be insufficient. Typically, with \( x_i = 2 \), the outgoing node \( o_i \) to the arc \( \lambda \) transmits a \( x_i \)-vector including (at most) the two highest protection costs which are greater than the threshold \( T_s = 70 \). This \( x_i \)-vector is deduced from the sorted list \( cost \), and corresponds to: \{ (node_1, 100), (generic_risk, 80) \} (Alg. 1). We note that the last couple of the \( x_i \)-vector sent by the node \( o_i \) contains a special risk, called \( generic \), to indicate that the \((x_i + 1)\)th highest protection cost on the arc \( \lambda \) is in the doubtful area.

When a node \( plr \) receives the \( x_i \)-vector transmitted by \( o_i \), it updates its protection cost table by approximating the protection costs on the arc \( \lambda \) as follows (Alg. 2):

\[
\begin{align*}
\delta_{node_1}^\lambda &= 100 \\
\forall r \in \{ \text{r is a risk} \land r \neq node_1 \} : \delta_r^\lambda &= 80
\end{align*}
\]

As we see here, all the risks whose protection costs on the arc \( \lambda \) are not transmitted with the \( x_i \)-vector, are aggregated and approximated by the \((x_i)\)th highest protection cost on this arc. As this last cost is higher than the threshold, any computation of a new backup LSP \( b \), protecting against a risk which is not conveyed in the transmitted \( x_i \)-vector of \( \lambda \) and claiming a quantity of bandwidth \( bw(b) \), excluding by mistake the arc \( \lambda \). However, any other computation will include or exclude the arc \( \lambda \) without mistake.

In Fig. 3 for instance, after the reception by node \( plr \) of this \( x_i \)-vector, if \( x_i \)-vector = [(node_1, 100), (generic_risk, 80)] \), node \( plr \) excludes by mistake the arc \( \lambda \) in its next computation of a backup LSP if this last one claims a quantity of bandwidth greater than 20 \((20 - BC_s - 80)\) and protects against failure risks which do not belong to \{node_1, srlg_1, link_4\}; otherwise, the arc \( \lambda \) is included or excluded without mistake.

Example. Consider the example depicted in Fig. 3 and assume that:

- The bandwidth quantities desired by new backup LSPs are uniformly distributed in the interval [1, 30].
- Any backup LSP is dedicated to the protection of only one failure risk.
- The risk to be protected by a new backup LSP is randomly (uniformly) chosen among the set of failure risks \( FR \).
- The last \( x_i \)-vector advertised by the node \( o_i \) for the arc \( \lambda \) is [(node_1, 100), (generic_risk, 80)].

The probability to reject the arc \( \lambda \) by mistake in a next backup LSP computation is determined as follows:

\[
P_{rej/m}(\lambda) = \frac{30 - (100 - 80)}{30 - 1 + 1} \times \frac{card(FR) - card(node_1, srlg_1, link_4)}{[card(FR) - 3]/3 - card(FR)}.
\]

For the minimal number of risks where \( card(FR) = 7 \), we have \( P_{rej/m}(\lambda) = 19\%. \) For the maximal number of risks where \( card(FR) = \infty \), we have \( P_{rej/m}(\lambda) = 33.33\% \).

4.2.2. Sure area \( x_{i,plus1} \)-cost \( \leq T_s \)

When the \((x_i + 1)\)th highest protection cost on an arc \( \lambda \) is in the sure area, the advertisement of a \( x_i \)-vector of a maximal size \( x_i \) is sufficient to decide without mistake if the arc \( \lambda \) can be selected (or not) to be in a new backup LSP (Principle 1 of PLRH).

In Fig. 3 for instance, the \((x_i + 1)\)th highest protection cost on the arc \( \lambda \) is in the sure area when \( x_i > 2 \). Thus, the advertisement of a \( x_i \)-vector containing at least the three highest protection costs on \( \lambda \) is sufficient. Typically, with the choosing of \( x_i \) equal to 3, the outgoing node \( o_i \) to the arc \( \lambda \) transmits a \( x_i \)-vector including (at most) the three highest protection costs which are greater than the threshold \( T_s \). This \( x_i \)-vector is deduced from the sorted list \( cost \), (Alg. 1) and corresponds to: \{ (node_1, 100), (srlg_1, 80), (link_4, 80) \}.
When a node \( p_{lr} \) receives the \( x_{\cdot}, \text{vector} \) transmitted by \( v_o \), it updates its protection cost table by approximating the protection costs on the arc \( x \) as follows (Alg. 2):

\[
\delta_{\text{node}, k} = 100 \land \delta_{\text{srli}, k} = 80 \land \delta_{\text{link}, k} = 80
\]

Contrarily to the case \( x_{\cdot}, \text{plus}_1 \cdot \text{cost} > T_{S_s} \), where the protection costs of the risks, which are not conveyed in the advertised \( x_{\cdot}, \text{vector} \), are approximated by the \( x \cdot \text{hth} \) highest protection cost on the arc \( x \), in the sure area these protection costs are aggregated and approximated by zero. As explained in the previous section, this approximation does not alter the decision of including or excluding the arc \( x \) in the next computation of a backup LSP \( b \).

Indeed, if the new backup LSP \( b \) protects against the failure of a risk belonging to the set \( \{ \text{node}_1, \text{srli}_1, \text{link}_1 \} \), node \( p_{lr} \) can deduce easily the exact value of the highest protection cost on \( x \) of the risks protected by \( b \). This highest protection cost corresponds to 100 units if \( b \) protects against the failure of \(
\delta_{\text{node}, k} = 100 \land \delta_{\text{srli}, k} = 80 \land \delta_{\text{link}, k} = 80
\)

As a result, node \( p_{lr} \) decides without mistake if the arc \( x \) can be selected to be in \( b \) or not (formula (4) in Section 2). When the new backup LSP \( b \) is planned to protect against the failure risks which do not belong to the set \( \{ \text{node}_1, \text{srli}_1, \text{link}_1 \} \), node \( p_{lr} \) selects the arc \( x \), without risk of mistake, when it computes the backup LSP \( b \) (cf. Principle 1 of PLRH in Section 4.1).

At this point, we deduce that with the advertisement of partial information about the protection costs (\( x_{\cdot}, \text{vectors} \) of a limited size), the decision of including or excluding an arc in a backup LSP computation can be done with high degree of correctness. It results that PLRH scales well since the transmission frequency and the size of messages advertising the protection bandwidth information can be reduced significantly without the deterioration of the bandwidth sharing possibilities. Indeed, the size decreases since at most the \( x \cdot \text{hth} \) highest protection costs (and their corresponding risks) are transmitted in the network, for each arc \( x \).

Moreover, the \( x_{\cdot}, \text{vector} \) transmitted for an arc \( x \) does not change at each establishment of a new backup LSP passing through the arc \( x \). This decreases also the frequency of advertisements since it is not necessary to flood a \( x_{\cdot}, \text{vector} \) as long as it is unchanged.

Finally, we note that we can obtain same performances as with complete information knowledge when the parameters \( \{ x \cdot \} \) are infinite. In such case, only the protection costs which are greater than the threshold on an arc are advertised. Another approach enhancing the quality of the protection cost approximation used in PLRH can consists to regulate the value of \( x \cdot \) according to the load of the arc \( x \) (i.e. a high value of \( x \cdot \) is assigned to the arc \( x \) when the number of protection costs on \( x \) which are greater than the threshold \( T_{S_s} \) is high).

4.3. IGP-TE extensions to support PLRH

Another important advantage of PLRH is its easiness of deployment in MPLS networks. Concretely, slight extensions to the IGP-TE protocols are sufficient to permit the distributed computation of backup LSPs protecting against the three types of risk: link, node and SRLG.

Hence, to advertise the protection bandwidth information (i.e. the \( x_{\cdot}, \text{vectors} \)), we propose to use and extend the TE parameters defined in [7] (for OSPF-TE) and [8] (for ISIS-TE). A new sub-TLV field (transmitted within the LSA field for OSPF and within the LSPDU field for ISIS) is then defined and associated to each arc. This field transports for each arc \( x \) its corresponding \( x_{\cdot}, \text{vector} \).

5. Analysis and simulation results

5.1. Simulation model

We evaluated the performance of our proposed heuristic PLRH by comparing it to the Kini’s flooding-based algorithm (FBA) and to the Kini’s heuristic (IKH) described in Section 3. Our choice is motivated by the following reasons:

- With FBA, all the protection costs are advertised in the network. In this way, the backup LSPs are computed in an efficient manner, without any bandwidth waste (due to approximations). Although, this algorithm is not practical (i.e. it floods the network), we used it in our simulation to measure the quality of the approximation provided by our heuristic. We note that FBA corresponds to a particular case of PLRH (null thresholds and infinite size of \( x_{\cdot}, \text{vectors} \) on all the network arcs).
- With IKH, the number and size of messages advertising the protection costs are decreased significantly. This heuristic uses only the maximal cost on a link for backup LSP computation and as a result, a (large) amount of bandwidth could be wasted on links. Contrarily to FBA, IKH is practical and it was used in several backup LSP computation algorithms (like in [11,13]).

In our simulation, we chose to divide the total available bandwidth of each arc into two pools (protection pool and primary pool). In this way, our measurements are not distorted by external parameters due to the task of primary LSP computation.

At each arrival of a request asking for a backup LSP computation, PLRH and IKH use their approximation model to compute the protection costs. Then, they check for validity of the inequality (4) and decide if the arc can be selected or not in the next backup LSP computation.

5.1.1. Topologies, SRLGs and traffic matrix generation

Our simulations use two well known network topologies. The first one [17,18], which is a USA network (it is the Long Haul network topology modified to include SRLGs), is depicted in Fig. 4a and is composed of 28 nodes, 45 bidirectional links and 22 SRLGs. It is a network topology of a large size where the average degree of nodes is equal to 3.21. To take SRLG failures into account, we added to the topology a large number of SRLGs (22 SRLGs). We note that these SRLGs are generated so that (1) they verify

Please cite this article in press as: M.Y. Saidi et al., PLR-based heuristic for backup path computation in MPLS networks, Comput. Netw. (2009) xxx-xxx
the geographical neighbourhood property\(^4\) (see Fig. 1) and (2) the protection against any failure risk remains physically possible. The second network topology, corresponding to an European network [18, 19] (it is the COST 239 network topology) that is depicted in Fig. 4b, is composed of 11 nodes and 26 bidirectional links. It is a network topology of a small size where the average degree of nodes is high and equal to 4.73.

To study the performances of PLRH in networks without SRLGs, we do not add any SRLG to the European network. Another network topology with different characteristics (50 nodes, 87 bidirectional links and 25 SRLGs) is used in our simulations. It can be found in [20].

The traffic matrix is generated randomly and consists of requests arriving one by one and asking for quantities of bandwidth uniformly distributed between 1 and 10 units. The head-end and tail-end routers of each primary LSP are chosen randomly among the network routers.

5.1.2. Primary and backup path computations

To focus only on the performance impact of the compared methods on the backup path computation, we have separated the task of primary path computation from that computing the backup paths (i.e. the task computing the primary LSPs is independent from that computing the backup LSPs). Thus, we divided the capacity of each unidirectional link in two disjoint pools: primary pool and protection pool. The primary pool is used to allocate the bandwidth for the primary LSPs whereas the protection pool is used for backup LSP bandwidth allocations.

In our simulations, we considered that the primary pool capacities are sufficient to satisfy all the establishment requests of primary LSPs. In this manner, the same primary LSPs, which are computed according to the shortest path first algorithm (SPF with unitary weights), are used to compare PLRH to FBA and IKH.

All the protection pool capacities of the network links in Fig. 4 are equal to 100 units except the bold links in Fig. 4a which have a capacity of 300 units. The backup LSPs are computed according to the constrained shortest path first algorithm (CSPF with unitary weights). Concretely, the LSPs correspond to shortest paths verifying the bandwidth constraints and bypassing the protected risks.

5.1.3. PLRH variants

Four variants of PLRH are used in our simulations. The first one PLRH\(_{90,90}\) uses a high threshold (\(T_S = 90\) on light links and \(T_S = 290\) on bold links) and \(x_{v, vectors}\) of an infinite size (\(\forall v: x_v = \infty\)) on all the arcs. The second variant PLRH\(_{2,0}\) uses \(x_{v, vectors}\) of a maximal size equal to 2 (\(\forall v: x_v = 2\)) but it does not employ the threshold (\(\forall v: T_S = 0\)). The third variant PLRH\(_{5,0}\) uses \(x_{v, vectors}\) of a maximal size equal to 5 (\(\forall v: x_v = 5\)) and a null threshold (\(\forall v: T_S = 0\)). Finally, the last variant PLRH\(_{5,90}\) uses a high threshold (\(T_S = 90\) on light links and \(T_S = 290\) on bold links) and \(x_{v, vectors}\) of a maximal size equal to 5 (\(\forall v: x_v = 5\)). We note that the variants PLRH\(_{2,0}\) and PLRH\(_{5,0}\) are useful when we do not have any information about the maximum quantity of bandwidth that a LSP can claim. Otherwise, the two other variants are more practical since they decrease the frequency of \(x_{v, vector}\) advertisements.

5.2. Comparison metrics

Four metrics are used for the comparison of PLRH to FBA and IKH: ratio of rejected backup LSPs (RRL), average protection bandwidth parameter changes (APR), protection bandwidth utilization (PBU) and highest protection cost average (HCA).

The first metric measures the ratio of backup LSPs that are rejected because of the lack of protection bandwidth. This metric is computed as the ratio between the number of backup LSP requests that are rejected and the total number of backup LSP requests.

The second metric measures the (average) rate of changes of the bandwidth protection parameters, used to approximate the different protection costs. It is computed as the ratio between the number of changes in the bandwidth protection parameters and the number of satisfied backup LSP computation requests. For PLRH, each change in the \(x_{v, vectors}\) increases APR whereas only the changes of the highest protection costs on arcs are counted with

---

\( ^4\) Since a SRLG is composed of logical links that share a common physical component, the end nodes of the links forming a SRLG are often adjacent.
IKH. We note that higher the APC value is, larger the advertisement frequency of messages conveying the protection bandwidth information is.

The third metric measures the efficiency of bandwidth sharing. It is computed as the ratio between the sum of the cumulated bandwidth of backup LSPs on arcs and the sum of all the protection capacities. Formally, it is determined as follows: \( PBU = \frac{\sum_k \lambda_k \cdot B_k}{\sum \text{B}_k} \).

The last metric measures the average rate of bandwidth which is really used for protection (note that the bandwidth used for protection is equal to the highest protection cost on each arc). It is computed as the ratio between the sum of the highest protection costs on arcs and the sum of all the protection capacities (i.e. \( HCA = \sum_k \text{Max}(\delta_k^r) / \sum \text{BC}_k \)).

At each establishment of 20 primary LSPs, the four metrics RRL, APC, PBU and HCA are computed for each backup LSP computation technique (FBA, IKH and PLRH). We point out that our simulation results correspond to the average values of 1000 runs generated randomly with different seed values.

5.3. Results and analysis

Fig. 5 depicts the evolution of RRL as a function of the number of primary LSPs established in the network. As expected, the various variants of PLRH have RRL values better and lower than those of IKH. This is due to the protection bandwidth information advertised with PLRH which includes that transmitted with IKH. Indeed, the unique protection cost \( \text{Max}(\delta_k^r) \) transmitted for each arc \( r \) with IKH is always included in the first couple of the \( \chi \)-vector advertised with PLRH.

We also observe in Fig. 5a (resp. in Fig. 5b) that for the 100 (resp. 500) first primary LSPs, the RRL of PLRH and IKH are very similar and close to zero. This is due to the very small quantities of protection bandwidth, generally lower than the threshold, allocated on the network topology arcs (see Fig. 8). As a result, almost all the arcs are selected and participated to the computation of the backup LSPs protecting the 100 (resp. 500) first primary LSPs in Fig. 5a (resp. in Fig. 5b).

After the establishment of the 100 (resp. 500) first primary LSPs in Fig. 5a (in Fig. 5b), the RRL of IKH increases more quickly than those of the PLRH variants since the quality of protection cost approximation used in PLRH is better than that of IKH. Whereas IKH approximates all the protection costs on an arc \( (\lambda) \) by the highest cost \( \text{Max}(\delta_k^r) \), PLRH approximates them by a protection cost which is lower than \( \text{Max}(\delta_k^r) \).

With regard to FBA and to the four variants of PLRH used in our simulations, Fig. 5 shows that the RRL values of PLRH\(_{2.0}\) are very higher than those of PLRH\(_{(9,90)}\) and FBA. This means that the advertisement of only two protection costs per arc is not sufficient to get a RRL close to that obtained with the complete knowledge of the protection bandwidth information (PLRH\(_{(9,90)}\) and FBA). We observe also in Fig. 5a (resp. in Fig. 5b), especially for the 600 (resp. 1300) first primary LSPs, that the RRL values of PLRH\(_{5.0}\), PLRH\(_{(9,90)}\), PLRH\(_{(9,90)}\) and FBA are very similar. Thus, the transmission of the five highest protection costs (and their corresponding risks) seems sufficient to obtain a high-quality protection cost approximation. This can be explained by (1) the locality of the PLRH heuristic (cf. annex) and (2) the heterogeneity of protection costs on arcs (due to the heterogeneity of backup LSPs). In fact, the locality of the backup LSP computation guarantees that the number of high protection costs (typically, the protection costs which are higher than the threshold) is limited and depends on the neighbourhood of the risks to be protected whereas the heterogeneity of protection costs on an arc results in a significative difference between the \( k \)th highest protection costs and their corresponding protection capacities (higher the difference is, better the protection cost approximation is). In our simulation scenario for instance, we observe for the same value of \( \chi \) (\( \chi = 2 \) or \( \chi = 5 \)), the protection cost approximation quality of PLRH on the USA network topology is better that obtained on the European network topology. This comes from the average node degree of the European network topology which is higher.

Fig. 5. Ratio of rejected backup LSPs (RRL).
than that of the USA network topology (i.e. the close neighbourhood area in the European network is larger than that corresponding to the USA network).

Concerning the second metric, Fig. 6 shows that FBA has higher APC values than those of PLRH and IKH. This can be explained by the systematical advertisement of protection costs used with FBA. In fact, at each establishment of a new backup LSP, FBA advertises the new values of protection costs (of the new backup LSP’s links) whereas only changes in the \( x \) highest protection costs of the new backup LSP’s links require new protection cost advertisements with PLRH (resp. changes of the highest protection costs of the new backup LSP’s links require new protection cost advertisements with IKH).

In Fig. 6, we observe also that the APC values of PLRH(\( <90 \)) and PLRH(\( <90 \)) are generally lower than those of IKH, PLRH(\( <0 \)) and PLRH(\( <0 \)). This comes from the use, in PLRH(\( <90 \)) and PLRH(\( <90 \)), of a high threshold \( T_0 \) (\( T_0/BC_{min} \geq 0.9 \)) which eliminates the flooding of a large number of \( x \)-vectors (since the protection costs, which are greater than the threshold, do not frequently change).

When the threshold is not used (null threshold on all arcs) as in PLRH(\( <0 \)) and PLRH(\( <0 \)), the advertisement frequency of the \( x \)-vectors increases with the augmentation of the parameter \( x \). Thus, the heuristics IKH (which only advertises for each arc the maximum protection cost) has a smaller APC than that of PLRH(\( <0 \)), which has itself a smaller APC than that of PLRH(\( <0 \)).

In addition to the previous observations, we note the similarity in Fig. 6a (resp. in Fig. 6b) between the APC values of PLRH(\( <90 \)) and those of PLRH(\( <90 \)), when the number of primary LSPs is lower than 600 (resp. 1300). This means that for such network load, the \( x \)-vectors transmitted with PLRH(\( <90 \)) are nearly the same as those advertised with PLRH(\( <90 \)). Obviously, this APC similarity explains also the RRL likeness of PLRH(\( <90 \)) and PLRH(\( <90 \)) in Fig. 5a (resp. in Fig. 5b) when the number of primary LSPs is lower than 600 (resp. 1300).
In Fig. 7, the evolution of the protection bandwidth utilization as a function of the number of primary LSPs is shown. As expected, we observe that FBA and \( PLRH(1, 90) \) have a better PBU than those of \( PLRH(5, 0), PLRH(5, 90) \) and \( PLRH(5, 0), PLRH(5, 90) \) which have in their turn better PBUs than that of IKH. This observation reinforces the results depicted in Fig. 5 since the two metrics RRL and PBU are very correlated (i.e. the values of the two metrics depend strongly on the distribution of the traffic and the density of risks in the close neighbourhood of the protected risks, as explained in the annex of this article). Hence, more precise the protection cost approximation quality is, higher the protection bandwidth utilization is and smaller the ratio of rejected backup LSPs is.

The comparison of the RRL and PBU values obtained on the USA network topology with those obtained on the European network topology confirms our thinking about the dependance of the (optimal) \( x_i \) values (i.e. the distribution of the traffic and the density of risks) on the close neighbourhood of the arc \( i \) (cf. annex). Hence, for a same RRL value, the PBU value obtained on the USA network is always lower than that obtained on the European network (the neighbourhood area in the European network is more connected than that of the USA network). For instance, for a RRL which is equal to 0.3, the corresponding PBU on the USA network is equal to 1.64 whereas the corresponding PBU on the European network is equal to 7.3.

Concerning the last metric HCA, we see in Fig. 8 that the average of highest protection costs on arcs increases slightly with the augmentation of the protection cost information size advertised in the network. This means that FBA, \( PLRH(1, 90), PLRH(5, 0), PLRH(5, 90) \) use more efficiently the protection bandwidth pool than \( PLRH(2, 0) \) which exploits in its turn more effectively the protection bandwidth pool than IKH. In Fig. 8a for instance, up to 10% of protection bandwidth capacity is wasted when the backup LSPs are computed with IKH. Obviously, the low HCA values of IKH can be explained by the inefficient protection cost approximation quality that it uses.

6. Conclusion

In this article, we proposed a distributed heuristic, called PLRH, for backup LSP computation. Our heuristic allows a high-quality approximation of the protection information necessary for the backup LSP computation with the advertisement of small and size-limited vector \( (x_i, \text{vector}) \) per network arc \( i \). This \( x_i, \text{vector} \) is formed of the \( x_i > 0 \) highest protection costs which are higher than a threshold \( T_s_i \) \( (T_s_i > 0) \), and does not change at each backup LSP establishment.

PLRH has several advantages. Firstly, it is symmetrical (all the nodes use a same information for the backup LSP computation) and it balances equitably computations among the network nodes. Secondly, PLRH does not require any network transfer between the entity computing a backup LSP and the one configuring it. Indeed, these two tasks are performed by the same node (PLR). Thirdly, PLRH is scalable and it reaches a high degree of bandwidth allocation.
sharing with the advertisement of a limited quantity of protection information (i.e. with $x_{\text{vector}}$ data). Finally, our heuristic is easy to be deployed since it requires only very slight extensions to the IGP-TE protocols for its installation.

Simulation results show that PLRH decreases significantly the number of rejected backup LSPs and the frequency of advertisements when the threshold and the size of $x_{\text{vector}}$ are well chosen. It also exploits more efficiently the protection bandwidth and it increases the bandwidth sharing.

**Annex (relation between network size, locality and $x_k$ value)**

Contrarily to the value of the threshold which can be determined easily when the maximal quantity of bandwidth that a LSP can claim is known (otherwise, the administrator can associate a low value to the threshold), the optimal value of $x_k$ depends on several parameters like:

1. the traffic matrix (traffic homogeneity),
2. the protection link capacities (or link capacities when the bandwidth is not separated into two pools),
3. the computation algorithm,
4. the maximum ratio of protection requests that can be rejected,
5. the topology network, the number of risks $N_R$ (especially node and SRLG risks) to be protected ($0 < x_k \leq N_R$) and the structures of the SRLGs,
6. etc.

In this section, we will concentrate on the impact of the increase of the network size (or on the impact of the increase of the number of network risks) on the value of $x_k$. To show that our approach scales when the network size increases (or when the number of network risks increases), we try to determine an upper bound ($UB_k$) for the value of $x_k$ beyond which there is no bandwidth waste (or the bandwidth waste is lower than a given constant). We note that this upper bound must not depend linearly on the network size (or on $N_R$). In general, it does not correspond to the optimal value of $x_k$; it ensures only that independently of the network size (or independently on $N_R$), the advertisement of a $x_{\text{vector}}$ of size equal to $UB_k$ avoids the reject by mistake of the arc $\lambda$ when a new backup LSP is computed. Although there are some rare cases in theory where the value of $UB_k$ depends linearly on the network size (or on $N_R$), the locality of protection ensures that the upper bound ($UB_k$) does not depend in practice on the network size but only on the number of risks in the neighbourhood (for instance: 1-hop neighbourhood or 2-hop neighbourhood) of the arc $\lambda$.

Consider the example shown in Fig. 9 where all the arcs have the similar protection capacity. To protect against the failure of a given risk $r$, all the arcs and nodes of the network topology, except those belonging to $r$, can be used. Typically, to compute a new backup LSP $b$ protecting against the failure of the link $A-F$ in Fig. 9, we can elect any link $l$ (such as $l=A-F$) to be in the backup LSP $b$. In order to decrease the amount of resources (MPLS labels, RSVP-TE states, etc.) reserved for the backup LSPs and to ensure acceptable recovery times (to satisfy the application time constraints), the backup LSP nodes should be close to the PLR (PLR is always an end node of the protected link) and to the next hop of the PLR (facility backup protection) or to the primary destination node (one-to-one backup protection). For instance, in Fig. 9, any backup LSP formed exclusively of arcs located in the 1-hop neighbourhood area of the protected link $A-F$ is preferred to the LSP $A-B-..-R-T-..-E-F$ because the very long length of the last LSP can result in the violation of the real time constraints of the supported communications. As a result, the links whose end nodes are far from the protected link/node are not (or are rarely) selected to be in a backup LSP protecting against the failure of that link/node although they verify the bandwidth constraints. We note that the exploration of a larger neighbourhood area to establish a new backup LSP does not generally decrease the blocking probability since this last metric depends strongly on the adjacent links of the head-end router of the backup LSP that is being computed (generally, the overloaded links which result in the rejection of a new backup LSP are those located in the 1-hop or 2-hop neighbourhood area of the PLR). Therefore, the number of risks whose protection cost is high on a given arc $\lambda$ is in practice limited and depends generally on the close neighbourhood area (close nodes) of the arc $\lambda$ (see Fig. 9 where almost all the backup LSPs verifying the real time constraints and protecting against the failure of link $A-F$ are located in the 1-hop neighbourhood area).

**References**

Mohand Yazid Saidi is a teaching assistant at the University of Rennes 1. He received in 2008 his Ph.D. degree in Computer Science from the University of Rennes 1. He is currently a member of IRISA (a CNRS-University of Rennes1-INSa joint research laboratory in computing science located at Rennes). His research topics and interests include dependable networking, high speed networks, traffic engineering, multicast routing, network QoS management, network security, sensor networks and multimedia distributed applications.

Bernard Cousin is a Professor of Computer Science at the University of Rennes 1, France. He received in 1987 his Ph.D degree in computer science from the University of Paris 6. He is, currently, member of IRISA (a CNRS-University-INSa joint research laboratory in computing science located at Rennes). More specifically, he is at the head of a research group on networking. He is the co-author of a network technology book: “IPV6 théorie et pratique” (Fourth edition, O’Reilly, 2006) and has co-authored a few IETF drafts in the areas of Explicit Multicasting and Secure DNS. His research interests include dependable networking, high speed networks, traffic engineering, multicast routing, network QoS management, network security, sensor networks and multimedia distributed applications.

Jean-Louis joined France Telecom eight years ago, and is currently working as Senior Architect in domotic networks and IP/MPLS networks. He is working on short-term design and deployment activities and on longer term research and development projects. He is actively contributing to the IETF, where he has been editing and co-authoring several Internet Drafts and RFCs. Jean-Louis is a frequent speaker in international conferences. His interests are Traffic Engineering, Fast Rerouting, Multi-Layer Routing as well as Multicast Transport. Jean-Louis holds an engineering degree from the Ecole Nationale Supérieure des Télécommunications de Bretagne, France.

Please cite this article in press as: M.Y. Saidi et al., PLR-based heuristic for backup path computation in MPLS networks, Comput. Netw. (2009), doi:10.1016/j.comnet.2009.01.009