MPLS RECOVERY MECHANISMS FOR IP-OVER-WDM NETWORKS

Didier Colle, Pim Van Heuven, Chris Develder, Steven Van den Berghe, Ilse Lievens,

Mario Pickavet, Piet Demeester

Ghent University - IMEC, Department of Information Technology

Sint-Pietersnieuwstraat 41, 9000 Gent (Belgium)

tel. no. +32 9 267 35 93

fax. no. +32 9 267 35 99

e-mail {dcolle, pim.vanheuven, chris.develder, steven,vandenberghe, ilse.lievens, mario.pickavet, demeester}@intec.rug.ac.be

Abstract. Due to the fast increase of Internet traffic and the enormous bandwidth potential of all-optical transport networks based on Wavelength Division Multiplexing, an IP-over-WDM network scenario is likely to be widespread in future communication networks. At the same time, IP networks are becoming more and more mission-critical. Hence, it is of paramount importance for IP-over-WDM networks to be able to recover quickly from frequently occurring network failures. This paper explains how Multi-Protocol Label Switching (both electrical and optical) recovery mechanisms can be important to reach that goal. Moreover, a novel MPLS recovery mechanism called Fast Topology-Driven Constrained-Based Rerouting is presented. Different MPLS recovery mechanisms are compared to each other. Special attention hereby goes to the additional capacity that is required to recover from frequently occurring failures.

Keywords: MPLS, IP-over-WDM, recovery, capacity dimensioning.

1. Introduction

Given the explosive growth of traffic on the Internet and the fastly growing number of Internet users [1], the Internet Protocol (IP) is becoming the dominant client of telecommunication networks. Most types of end-user communication today are making use of the ubiquitous TCP/IP protocol and many new services and applications being offered are also based on IP protocols. At the same time, IP networks are becoming more and more mission critical. Hence the support of service differentiation, the introduction of quality of service (QoS) in the Internet [2] and the ability to survive from network failures [3] are important research issues.

To accommodate the enormous growth of IP traffic, backbone networks should be able to transport larger and larger bandwidths. To cope with this need for evermore capacity, operators are introducing Wavelength Division Multiplexing (WDM) into their backbone network [4], [5], [6]. The first step is to upgrade point to point links by using multiple channels (wavelengths) on one fiber in order to overcome fiber exhaust or to share the amplifier cost between more channels, thereby lowering the cost per information unit. The next step is implementing network functionality directly in the optical layer by using optical crossconnects and add-drop multiplexers, leading to a so-called Optical Transport Network (OTN). This allows to reduce the amount of high-speed electronic processing equipment in the nodes, since transit traffic can be processed optically. Moreover, it creates opportunities to increase the network capacity and to simplify the network management [7]. This way, the OTN will allow cross-connecting at 10 Gbit/s granularities with line rates around 1 Tbit/s.

Due to the ability of IP to be the common revenue-generating convergence client layer and OTN as the bandwidth-rich transport layer, the IP-over-OTN network scenario is envisaged to be widespread in future communication networks. Based on the intensity of the interworking between the IP layer and the OTN layer, a distinction must be made between the overlay scenario and the integrated scenario. In an *overlay* IP-over-OTN network [8], a clear separation exists between the two layers with respect to network functionalities such as routing, resource management, recovery mechanisms, framing and monitoring, and others. The interaction between the layers is purely based on a clear client-server relationship, IP being the client and OTN being the server. In an *integrated* IP-over-OTN network [9], a tight interworking between the IP-layer and the OTN-layer is realized and the network functionalities are spread out over both layers.

Both the overlay and integrated IP-over-OTN scenarios will be deployed in the future. On one hand, for incumbent network operators IP will most probably not be the only client layer, since connectivity to its

preexisting legacy networks (based on SONET or SDH for example) will remain necessary in the near future. This situation will probably lead to a client-independent OTN to allow a smooth migration from the legacy network to an overlay IP-over-OTN network. On the other hand, for new network operators with IP as the only client-layer, a far-reaching integration between the IP-layer and the OTN-layer might be beneficial. In this case an integrated IP-over-OTN network will probably be preferred.

1.1 Electrical Multi-Protocol Label Switching

In an IP-over-OTN network scenario, Multi-Protocol Label Switching (MPLS) can play an important role. Historically, (Electrical) Multi-Protocol Label Switching (EMPLS) was a convergence of a number of 'IP switching' schemes, i.e., techniques that use Asynchronous Transfer Mode (ATM) hardware to speed up the forwarding of IP packets. In order to standardize the different IP switching implementations, an MPLS working group has arisen in the Internet Engineering Task Force (IETF) in 1997 [10]. This working group has since then been working on a common technology for IP switching that is independent of the underlying transport layer.

In a regular IP-network, an IP-packet is stored and forwarded in each router it passes along, based on a longest-prefix match process. In IP/MPLS networks [11], this cumbersome table-lookup is replaced by a simple and fast *label* switching technique similar to ATM: a received label on an incoming interface is translated and sent out on the appropriate outgoing interface. This label is usually included in a so-called shim header, which is attached to the IP packet. Due to this additional header, MPLS can run on top of every data-link layer technology.

In IP/MPLS networks, the flexible and scalable control plane of IP is preserved, but extended with MPLS functionality. In order to distribute the labels across the network, between the Label Switched Routers (LSRs), the Label Distribution Protocol (LDP) was developed. This signaling protocol allows to set up unidirectional paths (so-called label switched paths, LSPs) through the IP/MPLS network. Hence MPLS is a rather connection-oriented protocol (in contrast with regular IP). This has important consequences with respect to traffic engineering in the IP-layer: in IP networks enhanced with MPLS one can set up explicit routes (ERs) through the network to optimize the usage of available network resources, or to create distinct paths for different QoS classes.

However, MPLS has some additional features. First of all, setting up an LSP does not require an immediate allocation of resources to this LSP, as long as no traffic is transported along the LSP. Secondly, in MPLS it is possible to introduce a routing hierarchy by applying label *stacking* by attaching multiple shim headers to one IP packet (the switching is only based on the label on top of the stack). When a packet enters a higher level in the

routing hierarchy an additional label is pushed on top of the stack. When the packet leaves this higher level again the top most label is popped (removed) from the stack. Thirdly, if two or more LSPs to the same destination meet each other in an intermediate LSR and follow the same downstream path to the destination, it is possible to *(label) merge* these LSPs into a single LSP. This implies that in a network consisting of N LSRs, N multipoint-to-point LSPs are sufficient to connect all LSRs instead of N^2 point-to-point LSPs. This merging capability highly improves the scalability of MPLS networks. As will be explained in section 2, these MPLS features open up new possibilities for fast recovery mechanisms in the IP layer.

1.2 Optical Multi-Protocol Label Switching

Recently, the possible extension of the MPLS-paradigm towards the optical layer is receiving much research interest [12], [13]. In this so-called Optical MPLS (OMPLS) case, the control plane of the optical cross-connects is based on MPLS and each OMPLS-label corresponds to a wavelength on a link. Due to the physical meaning of a label in OMPLS (being a wavelength) some powerful characteristics of EMPLS can not be extended towards the optical layer. For instance, when an optical LSP is set up, then this LSP will immediately occupy a wavelength. This is in clear contrast with an electrical LSP, which may be established without occupying any resources. Furthermore, label stacking and label merging may have no equivalent in OMPLS.

Nonetheless, the extension of the MPLS-concept to the optical layer offers a number of potential advantages. In case of the overlay IP-over-OTN network scenario, OMPLS provides a framework for optical bandwidth management and dynamic provisioning of optical channels. Moreover, recent advances in EMPLS can often be reused without major adaptations in OMPLS. In the remainder of the paper, this overlay scenario will be called EMPLS/OMPLS. In the integrated IP-over-OTN network scenario, an additional advantage of MPLS arises: a single MPLS control plane can control both the IP- and OTN-layer. This enables a very close interworking between the two layers and a common platform for network management and operations control. In the remainder of the paper, this integrated scenario is denoted as (E/O)MPLS.

It must be noted that research on OMPLS is still in a preliminary phase. Many open and unresolved issues remain that will need further investigations in the coming years. For instance, the question of how the lack of label stacking and label merging possibilities can be coped with in OMPLS, is still for further study.

1.3 Outline of the paper

The remainder of the paper is structured as follows. In section 2, the importance of resilience in IP-over-WDM is highlighted and an overview of MPLS recovery mechanisms is given. The advantages and disadvantages of these mechanisms are discussed in section 3. In sections 4, 5 and 6, we concentrate on an in-depth study of the capacity that each of these MPLS recovery mechanisms needs. Based on the detailed problem descriptions (section 4), the capacities are dimensioned for a particular case study (section 5) and for a wide variety of problem instances (section 6). This allows to assess the typical capacity requirements for the different MPLS recovery mechanisms. The main conclusions of our study are summarized in section 7.

2. MPLS recovery mechanisms

2.1 Resilience in IP-over-WDM networks

Recovery in optical networks is currently under commercialization. For example, the popular self-healing ring concepts in SDH can easily be extended to the optical layer. However, in our IP-over-OTN scenario, optical layer recovery schemes cannot recover from a failing router for example. Even more, a node failure in the OTN may isolate an IP router and all traffic transiting that IP router will be lost, as illustrated in Figure 1.

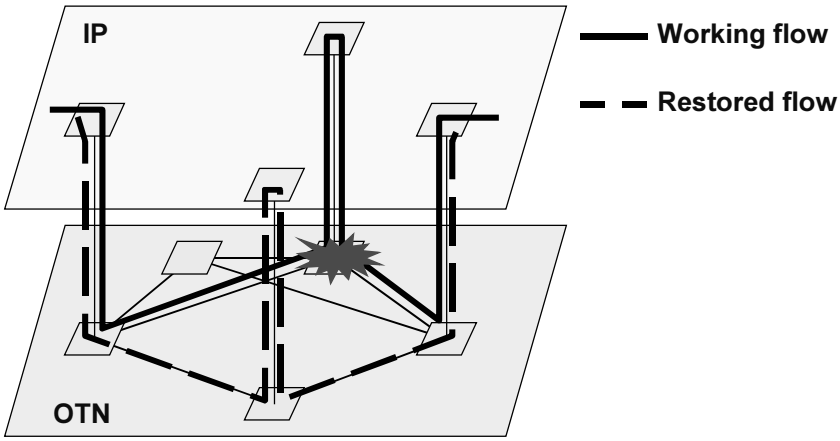

Figure 1: IP-layer recovery required for node failures

The recovery of IP traffic transiting the now isolated router can only be realized by rerouting in the IP layer. Fortunately, this is exactly what will happen after a while, due to the dynamic routing protocols running in the IP layer. An example of such a dynamic routing protocol is OSPF (Open Shortest Path First). OSPF is a so-called *link-state* routing protocol. All routers running a link-state routing protocol periodically advertise their adjacent

links, by flooding link-state packets over the whole network. Each router collects these received link-state packets in the link-state database. In this way, each router has enough information in its link-state database to reconstruct the topology of the network. By running the Dijkstra shortest path algorithm on this topology, a router constructs its routing table (defining for each destination along which outgoing interface packets have to be forwarded). Topology changes (due to failures for example) will result in other link-state packets being flooded over the network, in order to update the link-state database of all routers in the network. Once these link-state database updates are incorporated in the routing tables (by running again the Dijkstra algorithm), the traffic is rerouted along the new routes. Although this process is very robust, after the occurrence of a failure the routing information in different routers may be inconsistent for a while (e.g., temporarily resulting in loops), before all routing tables are stabilized again. The typical amount of time, required by link-state routing protocols is in the order of seconds or even tens of seconds.

Since IP networks become more and more mission critical, it is important to restore traffic as fast as possible. Therefore, the restoration capabilities of traditional dynamic routing protocols may be too slow. Even more, streaming applications (e.g., voice over IP) simply do not tolerate such high restoration times: they typically require a fraction of a second as restoration time. Summarizing, there is a need for recovery in the IP layer, since OTN recovery does not suffice (as shown in Figure 1), and the inherent IP-restoration capabilities may be unacceptable slow. Therefore, faster recovery techniques would be very welcome. The following sub-sections 2.2 and 2.3 will give an overview of such techniques based on MPLS and section 3 discusses the (dis)advantages of these techniques.

2.2 Overview of MPLS recovery mechanisms

This subsection gives an overview of proposals for recovery in MPLS networks. All these proposed techniques are intended for EMPLS networks. Their applicability for OMPLS networks will be discussed in the next section.

2.2.1 1:1 end-to-end protection

As mentioned in section 1.1, one of the important features of EMPLS is that the setup of an LSP does not require to immediately allocate resources to this LSP. This is used in 1:1 end-to-end protection [15], [16], by establishing a backup LSP from ingress LSR to egress LSR, which is physically disjoint from the primary (or

working) LSP. As long as the primary LSP is not failing, the ingress LSR forwards packets along this LSP. When the primary LSP is failing, the ingress LSR switches over to the backup LSP.

2.2.2 1:1 local protection

The same principle can be used for a backup LSP, which spans only a single link or a single node, in order to protect this link or node [16], [14]. The only additional requirement for this mechanism is that the primary and the backup LSP can be merged in the downstream node, where they meet each other again. Figure 2 shows an example of 1:1 link protection. The upstream node of the link is called Protection Switch LSR (PSL) and the downstream node Protection Merge LSR (PML) [16]. The backup LSP is pre-established, originates in the PSL and is merged with the working LSP in the PML. The PSL will switch over from working LSP to backup LSP when the link fails. Note that in order to protect a complete working LSP against single link failures, a backup LSP must be setup over every link of the primary path.

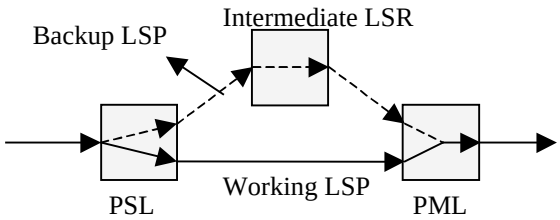

Figure 2: 1:1 link protection

2.2.3 Alternative path

1:1 Local Protection has the advantage (compared to 1:1 End-to-end Protection) that the protection switching is performed near the failure (immediately upstream), but it suffers from the fact that it requires many backup LSPs in order to protect a complete working LSP. This problem is solved by the Alternative Path technique [14], as shown in Figure 3. A backup LSP is provided in the opposite direction and concatenated to a physically disjoint LSP. As in 1:1 Local Protection, the LSR immediately upstream of the failure, switches the traffic from the working LSP to the backup LSP. This is very similar to the loop-back operation in self-healing rings: traffic is loop-backed towards the source LSR and is then forwarded along the physically disjoint route towards the destination LSR, where both working and backup LSP are merged into a single LSP.

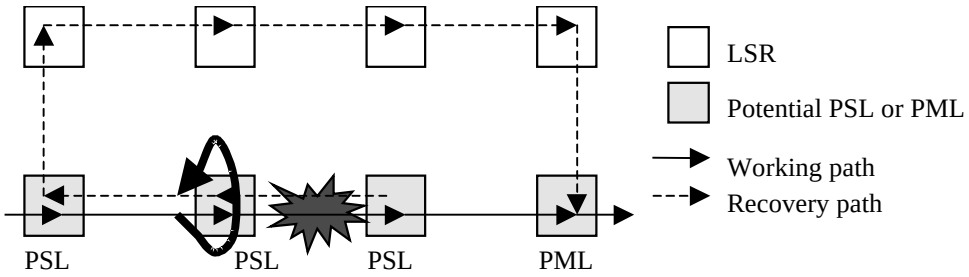

Figure 3: Alternative Path technique

2.2.4 Rerouting

Rerouting [17] differs fundamentally from the previous techniques, since it is based on the real-time calculation and establishment of the recovery LSP. After a routing table update (due to a failure) an LSR checks if there are LSPs which needs to be rerouted: this is the case when the routing table indicates another outgoing interface for the destination of the LSP than the interface along which the LSP is actually routed. If the LSR finds such LSPs, it will then modify the route through the use of the Label Distribution Protocol (LDP). The working LSP can be torn down, but this is not a necessity. Thus, we could say that the Rerouting technique represents path restoration in MPLS networks.

A typical drawback of path restoration is that the recovery time can be quite long. Indeed, the route calculation in Rerouting is often based on the link-state database inherently available in the LSR, when running OSPF[1]. Of course, for time-critical traffic more intelligent approaches could be developed (e.g., a set of pre-calculated routes, the use of a failure indication signal containing detailed information).

2.3 Fast Topology-Driven Constrained-Based Rerouting

We proposed another recovery technique in [18]: the so-called Fast Topology-Driven Constrained-Based Rerouting (FTCR). This technique provides rather fast protection switching, without the need to establish a backup LSP for each individual working LSP. The idea is that the LSR, immediately upstream from the failure, will detect the failure very fast and that its link-state database contains on overview of the topology. At the time that this LSR detects the failure, its link-state database will still contain the topology prior to the failure. And thus, this LSR can simply calculate the recovery path (from itself towards the egress LSR), based on a modified version of the link-state database, by removing the failing equipment from the link-state database. Once the

[1] Note that MPLS does not replace but extends the IP control-plane functionality.

recovery path is known, it can be established immediately by using CR-LDP (Constraint Routed - Label Distribution Protocol): the Explicit Route Label Request (ER-LR) messages being exchanged contain every node to be transited by the LSP. This can be done while link-state packets are still being flooded over the network, because CR-LDP does not require information from the IP routing tables in the different network nodes. Note that only the part of the LSP downstream from that LSR is rerouted and the upstream part of the LSP remains unchanged.

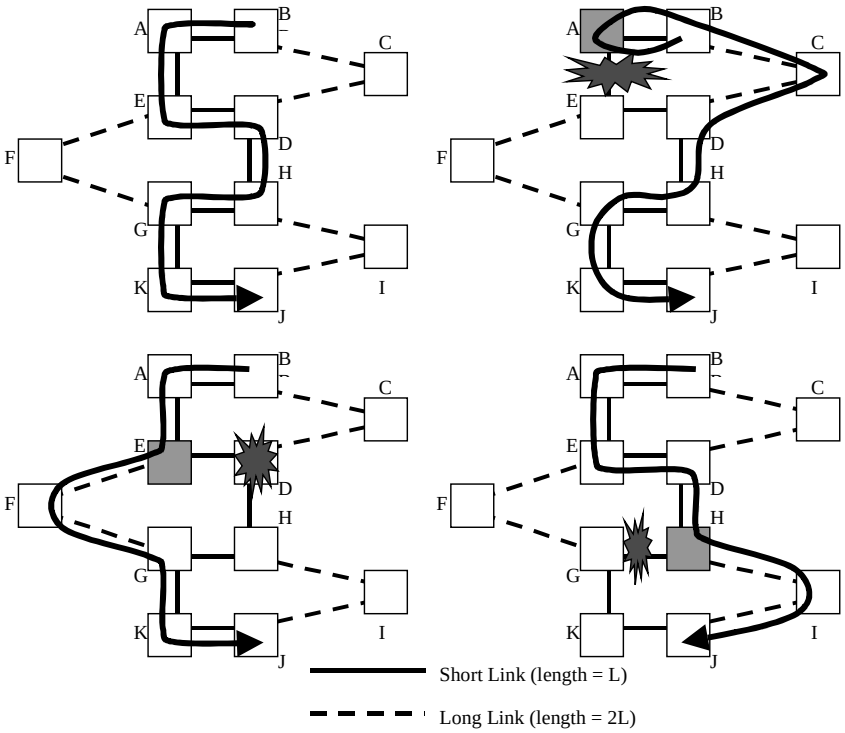

Figure 4: FTCR technique illustrated by an example (link weights proportional to distance)

Figure 4 illustrates this technique on a sample 11-nodes network. The top left drawing shows the shortest path routing for the connection B-J under study, in the case that there is no failure in the network: B-A-E-D-H-G-K-J. The case that the link A-E is failing is presented in the top right drawing. When node A detects that link A-E is failing, it will remove link A-E from its link state database and recalculate the shortest path from itself towards the destination. The downstream part of the LSP is modified with this calculated route (by setting up an explicitly routed LSP with CR-LDP, along the path A-B-C-D-H-G-K-J) and the upstream part, being B-A, remains unchanged. This particular case is special in the sense that the traffic is routed back over the source, which is of course not that capacity efficient. Note also that the route of the part of the LSP downstream of node D after rerouting is the same as before rerouting. The bottom left picture illustrates the case in which node D is failing. Node E (which is for this LSP the immediately upstream LSR from the failure) will detect that node D is

failing and thus it will modify the route of the part of the LSP downstream from itself: the new route downstream from node E is E-F-G-K-J. A last example is shown in the bottom right figure: link H-G fails, resulting in the part of the LSP downstream from node H being rerouted along H-I-J. This case illustrates that it is also possible that the rerouted part of the LSP enters the destination node through another interface than before rerouting.

3. Pros and cons of different MPLS recovery mechanisms

In section 2 we concluded that fast recovery techniques in MPLS would be very welcome, since the inherent IP-restoration capabilities are rather slow. None of the MPLS recovery techniques (except Rerouting) has to wait for the stabilization of the dynamic IP-routing protocols, which implies a significant reduction of the *restoration times*. Table 1 gives an overview of the parts affecting the restoration time for each recovery scheme. 1:1 Local Protection and Alternative Path will be the fastest techniques and Rerouting the slowest. FTCR and 1:1 End-to-end Protection are situated somewhere in between. Not only the restoration time is important, but also the *delay* perceived by a packet *to travel through the network*. If we consider that this delay increases in function of the path length, then Rerouting will have the shortest delay along the recovery route and the Alternative Path scheme the longest. FTCR will perform at least as good as 1:1 Local Protection. These two timing criteria are important in choosing the right recovery scheme with respect to the QoS requirements.

Recovery Mechanism	*Failure Detection*	*Failure Propagation*	*Route Calculation*	*Routing Protocol Stabilization*	*Switch-over or Real-time setup*
1:1 End-to-end Prot.	Yes	Yes	No	No	Switch
1:1 Local Prot.	Yes	No	No	No	Switch
Alternative Path	Yes	No	No	No	Switch
Rerouting	Yes	Yes	Yes**	Yes**	Setup
FTCR	Yes	No	Yes*	No	Setup

Table 1: parameters influencing restoration times (* = pre-calculation possible; ** = also other possibilities)

Another issue is the *failure coverage*. Since the 1:1 Local and End-to-end Protection and Alternative Path schemes are based on pre-established backup LSPs, they cannot survive a double failure affecting both working and backup LSP. If Rerouting is adapting itself to topology updates in the link-state databases, then it can survive any failure not splitting the network in two or more disconnected sub-networks. Also FTCR, which is also based on the link-state database prior to the failure, is rather flexible. However, it can fail if two failures occur at almost the same time. Then, only the immediate downstream link or node is removed from the link-state database. This implies that the calculation of the new shortest path can potentially be routed over the other failure (which was not yet advertised to the link-state database). Therefore, the current proposal is only suitable

for single failures. However, it is more flexible than schemes based on pre-established LSPs, when the time between two single failures is long enough to be incorporated in all link-state databases.

Although these schemes are rather promising, there are still some issues to be solved. First of all, an important question is which information to include in the *Failure Indication Signal (FIS)* [17], [16] sent upstream to notify the LSR responsible for rerouting that a failure has occurred. Although it may not be that important for a scheme based on a pre-established backup LSP, it may speed up Rerouting significantly because it no longer has to wait for stabilization of the dynamic routing protocols when this FIS contains information identifying the failing equipment. In the case that the FIS identifies the egress LSR as being failed, then it can avoid that traffic is forwarded towards a black hole (even for schemes based on backup LSPs). A related question concerns the *failure propagation from the OTN layer to the EMPLS layer*. Since all LSRs are periodically sending Liveness messages to their neighbors, a failure is only detected by the lack of Liveness messages receipts, which does not specify whether the upstream link or rather the LSR (sending the Liveness messages) is failing. A failure indication signal from OTN to EMPLS would be very useful to indicate a link failure. However, such a signal from OTN to EMPLS layer needs to be included in the standardization of the interface between both layers (only an integrated (E/O)MPLS does not need such a modification of this interface standard, since the single control plane has a full view on the network, covering both layers), Such a signal may give more accurate information concerning the failure and it may speed up the failure detection process significantly.

Up till now, we have been focussing on EMPLS, for the discussion of all these recovery techniques. However, it is also important to know how these schemes would perform in an *OMPLS network*. The schemes based on pre-established backup LSPs are harder to implement in or less interesting for an OMPLS network, since the setup of the backup LSP also implies the usage of a wavelength along its route and the backup LSP cannot simply be merged with the working LSP. This merging problem could be solved only in the egress LSR, when dropping both backup and working LSP to the electrical layer (however, this would require two tributary ports). Another solution could be to send a FIS not only upstream but also downstream, providing the possibility to perform a protection switch in both the up- and downstream LSR (where both LSPs have to be merged). However, one has to be careful when deploying such a solution, because up- and downstream switches must remain synchronized at all times!

Since FTCR and Rerouting set up a recovery LSP only when the failure occurs, they do not suffer from these capacity and merging[2] problems. Even more, since FTCR was designed in first instance for single failures and it restores traffic rather fast (since it does not have to wait for the stabilization of the dynamic routing protocols), it is very suitable for OMPLS networks: it is a rather fast and capacity efficient alternative!

A last issue deals with the *required resources*. Rerouting will require the lowest amount of additional resources for survivability. As will be proven in the rest of this paper, we expect that FTCR will be situated somewhere in between Rerouting and 1:1 Local Protection, since the rerouting path begins close to the failure (local character, cf. 1:1 Local Protection) and ends at the destination node (end-to-end character, cf. Rerouting). Furthermore, it is intuitively clear that Alternative Path requires more than 1:1 End-to-end Protection, and that 1:1 End-to-end protection requires more resources than Rerouting.

4. Capacity requirements of recovery mechanisms

In this section 4 and the following sections 5 and 6, we want to assess how much additional capacity is required in an (electrical or optical) MPLS network if we want the traffic to be survivable in case of frequently occurring failures. More in particular, we want to estimate the capacity requirements of the Fast Topology-Driven Constraint-Based Rerouting mechanism by comparing it with 1:1 Local Protection and Rerouting. These techniques were chosen, because FTCR essentially combines the capacity requirements of both techniques.

Section 4 concentrates on the problem description and the assumptions we made. The network and traffic model is described in subsection 4.1 and the applied capacity model is discussed in subsection 4.2. In subsection 4.3 it is investigated which failure types should be taken into account for recovery, leading to four main failure scenarios. The choice of the routing and rerouting paths for the different recovery mechanisms is motivated in subsection 4.4.

4.1 Network and traffic model

The capacity requirements of the different recovery mechanisms will be compared for a wide variety of MPLS network topologies, consisting of network nodes that are interconnected by some bi-directional lines. The topology of the MPLS-network can be represented as a directed graph $G = (V,E)$. V is the node set of the graph,

[2] It may be required to first tear-down the primary O-LSP.

E is its link set. The head node and tail node of the directed link e are denoted i_e and j_e, respectively. Since all lines are assumed to be bi-directional, for every link e from i_e to j_e there is a corresponding link e' in the opposite direction from j_e to i_e.

Since IP traffic can be highly asymmetrical and due to the uni-directional nature of LSPs in MPLS, we will model the traffic requirements as a non-symmetrical demand matrix $[d_{ij}],\ i,j \quad V$ ($d_{ij} = 0$ if $i = j$), i.e., d_{ij} can be different from d_{ji}. This demand matrix will typically be imposed by service level agreements (SLAs) between the network operator and its customers.

4.2 Capacity model

The capacity of link e is denoted as u_e. To cover the most general situation, we assume that the capacities of the MPLS-links can be different for both directions (uni-directional facilities). This implies that u_e can be different from $u_{e'}$. Despite the discrete character of link capacities in reality, we will assume in our model that a link capacity u_e can take any positive value. This choice is motivated by the fact that a discrete capacity model is not necessary to assess the capacity requirements of different recovery mechanisms (since we are not optimizing the choice of (re)routing paths to improve the filling of capacity modules) and would only obscure our comparison. We assumed a linear capacity model, i.e., the cost c_e of a certain capacity u_e on a link e is proportional to the capacity u_e:

$$c_e = a_e \times u_e$$

where the coefficient a_e represents the (link-dependent) cost of one unit of capacity on link e. In reality, this cost a_e may differ from link to link (e.g., depending on the distance covered by the link) and can incorporate both installation and maintenance costs. We also assumed that the same cost per unit of capacity was assigned to anti-parallel links e and e': thus $a_e = a_{e'}$.

Note that node capacities are not explicitly taken into account in our model. The reason is that linear node costs (cost proportional to throughput capacity) can easily be incorporated in the link coefficients a_e. The total cost c_{tot} of the capacitated network can then be expressed as the sum of the link capacity costs:

$$c_{tot} = \sum_{e \in E} a_e \times u_e$$

4.3 Important failure types

To understand which failure types must be taken into account in our study, it is imperative to make a distinction between the integrated and overlay network scenarios. In case of the integrated (E/O)MPLS scenario, the topology viewed by MPLS corresponds to the physical network topology. On the physical layer, single line failures (e.g., cable cut) and single node failures (e.g., equipment defect) are usually the failure types with highest probability. In case of the overlay EMPLS/OMPLS scenario, the situation is somewhat more complex. The topology viewed by OMPLS is again the physical network topology. The topology viewed by EMPLS however is a logical topology, where a link in the EMPLS-layer corresponds to an LSP-connection in the OMPLS-layer. In such a logical topology single line failures and single node failures should again be taken into account. For instance an EMPLS-LSR defect corresponds to a single node failure, an EMPLS interface defect corresponds to a single line failure. Moreover also multiple failures can occur in the logical topology, because a single transport node failure or a single transport line failure can affect multiple optical LSP-connections, corresponding to multiple failures in the EMPLS-layer. However, if we assume that these single transport failures are recovered in the OMPLS-layer itself (except for traffic terminated in a failing node, of course), then the dominant failure types in the EMPLS-layer are again single line and single node failures.

The arguments above show that in each network layer (EMPLS, OMPLS or integrated (E/O)MPLS) single line failures and single node failures are the most important failure types. Hence, we considered four different scenarios in our experiments:

I. **Single line failure.** The probability of a line failure is much higher than the probability of a node failure. In case of failure detection, the recovery mechanism assumes the failure to be a single line failure. Additional capacity must be sufficient to recover all traffic affected by any single line failure.

II. **Single node failure.** The probability of a line failure is much lower than the probability of a node failure. In case of failure detection, the recovery mechanism assumes the failure to be a single node failure. Additional capacity must be sufficient to recover all transit traffic affected by any single node failure.

III. **Single line or node failure, failure type known.** The probabilities of line and node failures are in the same order of magnitude. In case of failure detection (only appropriate for 1:1 and FTCR case), we assume that the recovery mechanism knows the failure type (line failure or node failure) and acts

accordingly. Additional capacity must be sufficient to recover all traffic affected by any single line failure and must be sufficient to recover all transit traffic affected by any single node failure.

IV. **Single line or node failure, failure type unknown.** The probabilities of line and node failures are in the same order of magnitude. In case of failure detection, we assume that the recovery mechanism does not know the failure type. Hence the rerouting path is based on the worst case assumption that a node failure has occurred. Additional capacity must be sufficient to recover all traffic affected by any single line failure and must be sufficient to recover all transit traffic affected by any single node failure[3].

4.4 Routing and rerouting paths

The overall objective of our study is to dimension the capacity of an MPLS-network so that the following constraints are fulfilled:

1. During normal operation (i.e., faultless conditions), all demands $[d_{ij}]$ should be transported through the network.
2. In case of a single failure (see failure scenarios I - IV above), all affected traffic should be rerouted by the considered recovery mechanism (except for traffic terminated in a failing node and the case described in footnote 3). The routing of the traffic that is not affected by the failure, remains unaltered.

Taking into account the capacity model described above, the cheapest way to transfer the traffic through the network corresponds to a routing through the graph G along the 'shortest' path[4] with link weights w_e proportional to the capacity unit cost a_e: $w_e = a_e$. Hence we will route all traffic during normal operation along its shortest path. In case of a failure, the rerouting paths are also determined by calculating the shortest path through the graph G with link weights a_e:

---

[3] Note that in case of scenario IV, it is possible that even a single line failure leads to traffic loss. If the last line along a LSP fails, then the 1:1 Local Protection and FTCR recovery mechanisms will assume that the terminating node has failed and take no recovery actions for that LSP. Since Rerouting follows the changes in the topology database, it will find a new shortest path after stabilization of the IP-routing protocol, leading to identical results for Rerouting in scenario III and IV.

[4] If for a certain demand d_{ij} more than one shortest path exists, one of these paths is arbitrarily chosen and all traffic is routed along this single path. We refer to section 6.4 for a comparison of single-path routing with multi-path routing.

- In case of **FTCR**, the rerouting path is determined between the LSR immediately upstream of the failure and the egress LSR. This rerouting path is calculated by searching the shortest path between both nodes through a modified graph G'. In the case of a node failure (scenario II, III or IV), G' is obtained, by removing the failing node (and all its adjacent links) from the graph G. In the case of a line failure, deriving G' depends on the failure scenario. In scenario I and III both directed links (corresponding to the failing line) are removed from G; in scenario IV the immediately downstream node is removed from the graph G. The part of the path upstream of the failure remains unchanged.
- In case of **1:1 Local Protection**, the appropriate pre-calculated bypass route is chosen. For line failures in scenario I and III, these bypass routes are obtained, by searching the shortest path between the head and tail nodes of the affected links, in a modified graph G'. This graph G' is obtained, by removing the affected links from G. In the case of other failures, G' is derived from G, by removing the failing node (node failure in scenario II, III or IV) or the tail node of an affected link (line failure in scenario IV). The bypass route is calculated as the shortest path through G', between the two nodes which were in G adjacent to the removed node. Note also that bypass routes between the same nodes but in opposite direction may transit different nodes.
- In case of **Rerouting**, a graph G' is derived from the graph G, by removing the affected links (in case of a line failure in scenario I, III or IV) or node (in case of a node failure in scenario II, III or IV). The rerouting path is obtained by searching the shortest path through G' between ingress and egress nodes.

5. Case study

To start our study, we compared the capacity requirements of the three recovery mechanisms on a realistic network for the failure scenario IV (both line and node failures are considered and no special failure identification feature is required). The topology under study is based on the Espire network (as shown on [19]): the topology consists of 44 nodes, connected by 57 lines. The link weights were assigned proportional to the roughly estimated distance.

10 demand matrices were generated randomly. Each (integer) demand $d_{ij},\ i,j \quad V$ was assumed to have a uniform distribution between 0 and D_{ij}. This maximum value D_{ij} is proportional (scaling factor 1000) to the product of node weights (v_i*$v_j,\ i,j \quad V$) of the source and destination node. The node weights $v_i,\ i \quad V$ (varying between 1 and 5) were used to make larger cities more important than smaller ones.

Figure 5 shows the total cost of the spare capacity expressed relative to the cost of the working capacity for failure scenario IV. The figure shows average values (as the height of the bars) and the standard deviation (line markers).

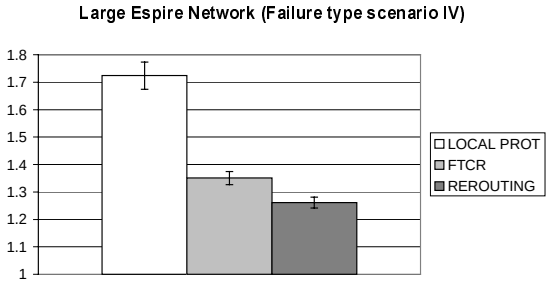

Figure 5: additional cost comparison for Large Espire Network

These results show that the capacity cost is more than doubled compared to a network without any survivability. As one may expect, we can see that 1:1 Local Protection is the most expensive one and Rerouting the cheapest one. The FTCR mechanism is situated somewhere in between. This can intuitively be explained as follows. 1:1 Local Protection has to reroute all affected traffic locally, which is of course worse than Rerouting, which has a global scope and thus potentially spreads out the rerouted traffic over the network. FTCR has upstream the local nature of 1:1 Local Protection but downstream the global nature of Rerouting, which explains that the cost of FTCR lies somewhere between the cost of 1:1 Local Protection and Rerouting. As mentioned in footnote 3, 1:1 Local Protection and FTCR may not be able to restore all traffic for line failures in failure type scenario IV. As a result some traffic will get lost, which implies that no spare capacity is required for this lost traffic. This is not the case for failure type scenario III, where 1:1 Local Protection and FTCR are able to recover all traffic affected by a line failure, resulting in more spare resources and a higher cost for these schemes.

6. Simulation results

The previous section discusses the results for a single case study. We should be careful to draw some general conclusions from this particular case study. Therefore, this section is comparing the three recovery mechanisms while changing several parameters (other topology, other scaling factor for demand generation, other link weights, …)

6.1 Comparison of recovery mechanisms for different failure types

The goal of this subsection is to compare the recovery mechanisms under different traffic conditions and on different topologies. This comparison is also made for different failure types (being discussed in section 6.1)

The network topologies being considered (described in Table 2) are based on some realistic networks, which can be found on the Internet [19], [20]. The link weights are again estimated to be proportional with the length of the links. For each topology, once again a set of 10 traffic matrices is randomly generated as described in section 5. We also considered a heavy traffic load (scaling factor 1000) and a light traffic load (scaling factor 0.2). The case of the lightly loaded network leads to many zero demands and hence a sparse demand matrix.

Network Name[5]	*# Nodes*	*# Lines*	*Nodal degree*
Large(1) Qwest Network	14	25	3.57
Large(2) Qwest Network	14	27	3.86
Small Qwest Network	11	17	3.09
Large Espire Network	44	57	2.59
Small Espire Network	30	36	2.40

Table 2: topology definitions

Figure 6, Figure 7, Figure 8, Figure 9 and Figure 10 make the comparison for each of these networks. As concluded in section 5, we find that Rerouting is the cheapest solution, 1:1 Local Protection the most expensive one and FTCR is situated somewhere in between. There is only one exception to this rule: failure type scenario IV may result in FTCR (and sometimes also 1:1 Local Protection) becoming cheaper than Rerouting. This can be explained by the fact that traffic is also lost for line failures when deploying 1:1 Local Protection or FTCR[6].

[5] All presented topologies are based on the Qwest and Espire networks, as illustrated on the URLs [20] and [19], respectively. The capacities depicted in these networks were used to obtain our topologies: these capacities do not have any impact on our further simulations. The Large(1) Qwest network is restricted to only OC-48 and OC-12 links on the picture, the Large(2) Qwest network to OC-48, OC-12 and OC-3 and the Small Qwest network to OC-48 only. The Large Espire network contains both OC-3 and DC-3 links and the Small Espire network only OC-3 links.

[6] As discussed in section 5, lost traffic will result in less spare resources and thus in a reduced cost.

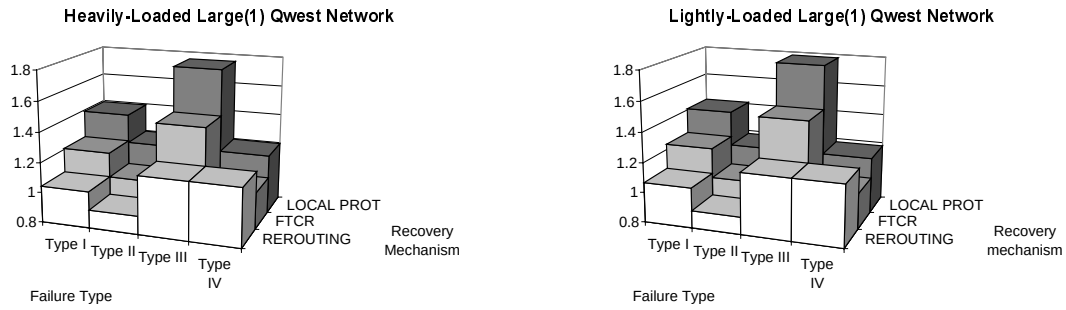

Figure 6: Large(1) Qwest Network (additional cost for survivability relative to network w/o survivability)

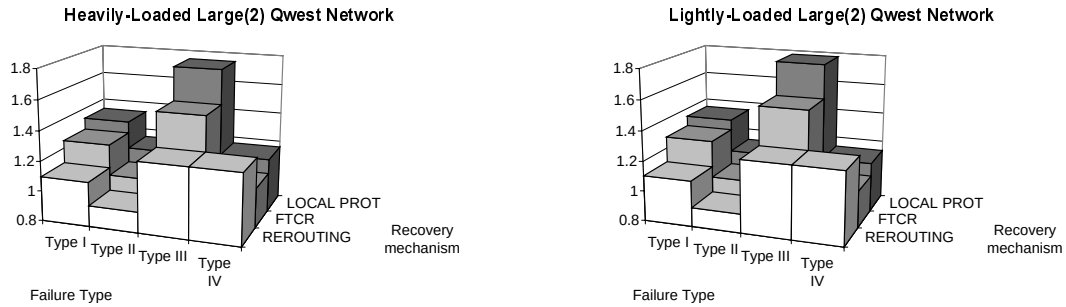

Figure 7: Large(2) Qwest Network (additional cost for survivability relative to network w/o survivability)

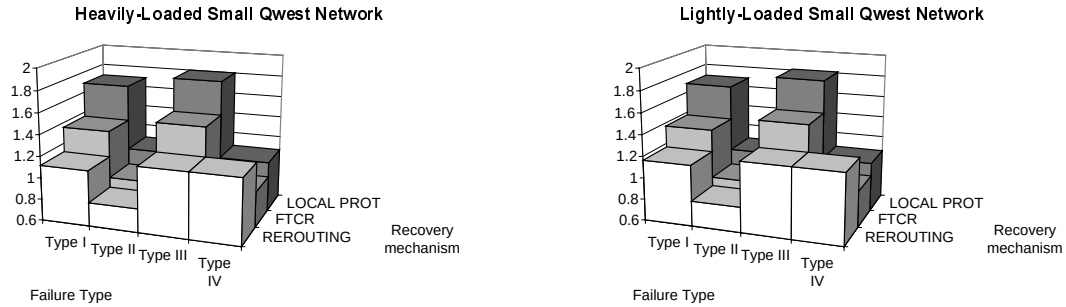

Figure 8: Small Qwest Network (additional cost for survivability relative to network w/o survivability)

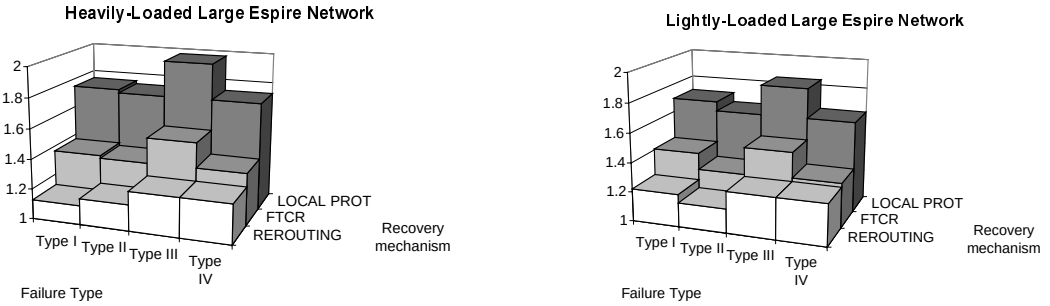

Figure 9: Large Espire Network (additional cost for survivability relative to network w/o survivability)

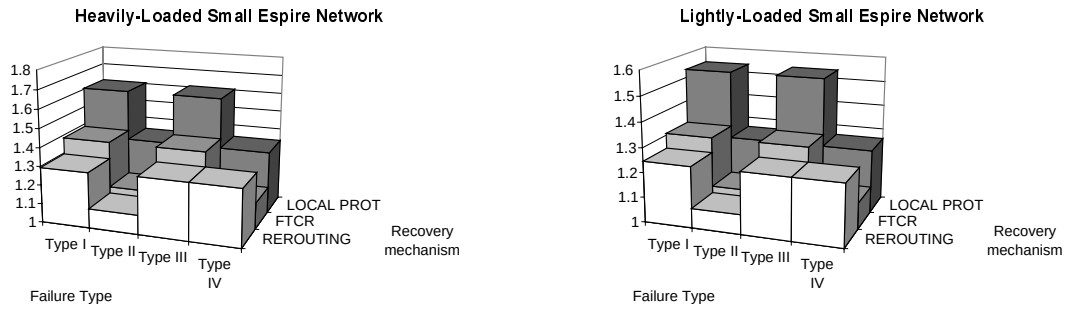

Figure 10: Small Espire Network (additional cost for survivability relative to network w/o survivability)

Although more traffic is affected by a node failure (scenario II) than by a line failure (scenario I), it is cheaper to design a network which can recover from node failures. The reason is probably that the effect of loosing traffic, due to a node failure, may be rather important. Since shortest path (instead of an optimized) routing was considered, we did not encourage artificially long routes, resulting in a smaller (compared to optimized routing) amount of transit traffic in the nodes. The most extreme case would be to route all demands over a single hop, making it impossible to recover from node failures and thus resulting in no additional spare capacity. A similar argument explains the differences between scenario III and IV. A cheaper design is achieved for scenario I compared to scenario III and for scenario II compared to scenario IV. This is of course a result of the fact that scenarios I and II are subsets of scenarios III and IV, respectively.

Some of these differences discussed above are nihil or very small in particular cases. The cost for scenarios III and IV in the case of Rerouting is identical, since Rerouting treats all failures in the same way for both scenarios. As discussed above, scenario IV is more expensive then scenario II. However, the additional cost to recover from line failures as well is not significant, but sometimes not zero. This slight difference can be explained as follows. In the case of a (undirected) line failure, both endpoints consider that the opposite side is failing. This means that although a subset of rerouting paths belonging to a single failure of the endpoints is being used, the union of both subsets has to become active at the same time. Only when these rerouting paths (belonging to different subsets) do not share a directed link, there is no need for more capacity. An example is shown in Figure 11, where one LSP is routed from A to D via B and C and another LSP from D to A via C and B. Both LSPs will be affected by a failure of the line B-C. B will detect a failure, but B will not know whether C is available. Thus, B will assume that C fails, resulting in LSP A-D being rerouted via a backup LSP from B to D. C will also detect a failure and will not have enough information to know whether B is failing. This results in LSP D-A being rerouted via a backup LSP from C to A (because C assumes that B is failing). Note that both backup LSPs protect different nodes. This implies that they do not carry traffic at the same moment when considering only

single node failures. However, as shown by Figure 11, they have to carry traffic at the same moment in the case of a line failure. The right side of this figure shows an example, where both backup LSPs are routed over the same directed link, potentially requiring more capacity in the case of a line failure (compared to case of only single node failures).

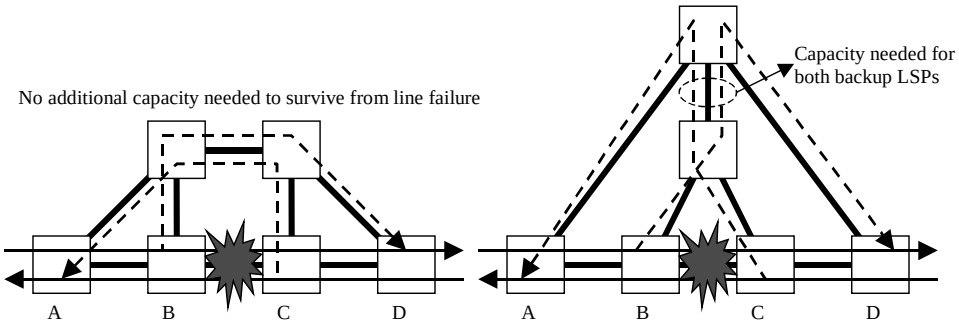

Figure 11: additional cost for line failures in scenario IV

Another conclusion that can be drawn from these results is that the amount of network traffic only has a minor influence on the results. This is again emphasized in Figure 12: for each failure type scenario and network combination, the ratio of the average values (presented in the previous figures) for a light load and a heavy load are calculated for each recovery mechanism and shown in Figure 12.

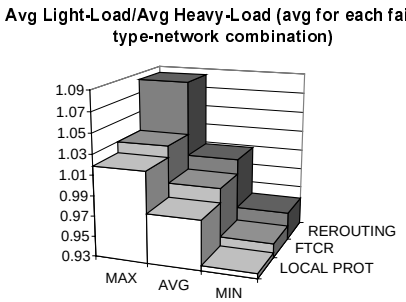

Figure 12: influence of amount of traffic loaded on the network

A last conclusion is that the differences in the cost depend on the network topology. However, no clear relation could be derived between the cost for added survivability and the number of nodes in the network or the average degree of the nodes (see Table 2 for the nodal degree).

6.2 Influence of topology

As mentioned in section 6.1 the relative cost for additional survivability depends on the topology of the network under study. However, we did not find any meaningful relation between both. Important in this context is the

connectivity of the network topology, but also the link weights used in the shortest path routing. In this section we present the study of the influence of both parameters.

To study the influence of the connectivity, we generated random topologies by varying the probability to include a line in the topology. All topologies were assured to be at least biconnected and contain 26 nodes. For each probability to include a line 10 topologies were generated. Also the link weights are generated randomly: a link being included receives a uniform link weight distribution up to 2000. All tests were performed with respect to the same demand matrix (scaling factor 1000), in order to make a fair comparison. The results for only line (type I) or only node (type II) failures are shown in Figure 13. The main conclusion that we can draw out of these plots is that the topology may impact the results (standard deviation of more than 10% for the relative cost for additional survivability). However, a clear tendency is not shown by these drawings, which confirms our statement in section 6.1. Only minor differences in average values (bars) are seen for highly meshed networks (link probability of 75% and 100%), but the standard deviation remains quite high (line markers). The lower working cost (thus for a network without survivability) for higher line probabilities, illustrates the higher meshedness/connectivity of the networks[7].

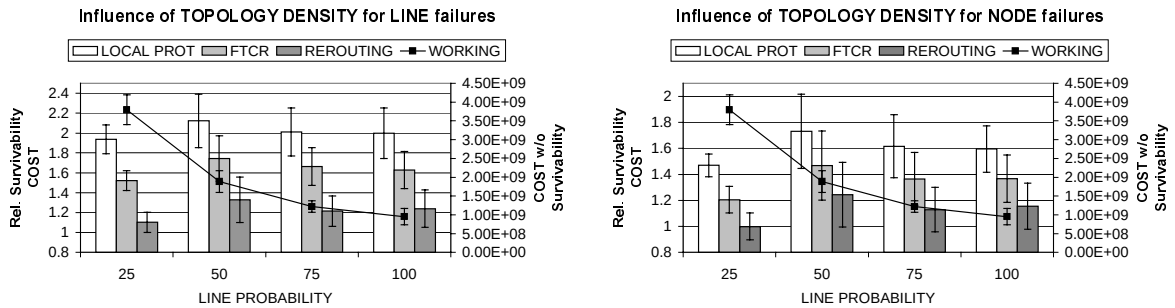

Figure 13: influence of network topology (density/connectivity)

In these experiments, random networks are generated with random link weights up to 2000. However, the link weights themselves may impact the shortest path routing. Therefore, we also assigned link weights randomly to the Large Espire Network, always under the same traffic conditions (one matrix of the experiments in section 5 was chosen). A uniform distribution up to a maximum link weight was considered for each link and for each maximum value 10 sets of link weights were generated. The results for only line (type I) or only node (type II) failures are shown in Figure 14. When comparing Figure 14 with Figure 13, then we see that the standard

[7] Note that although there is no clear tendency for the relative cost for additional survivability, the absolute value will of course follow the decreasing tendency of the cost for the working capacity.

deviation for the relative cost for additional survivability remains quite important, but that the standard deviation for the working cost (or the cost for a network without survivability) is much smaller in this experiment than the previous one. Although these plots do not show any hard relation between the maximal link weight and the relative (compared to the working cost) cost for the added survivability and the large standard deviation, one may notice a slightly growing tendency for growing link weights.

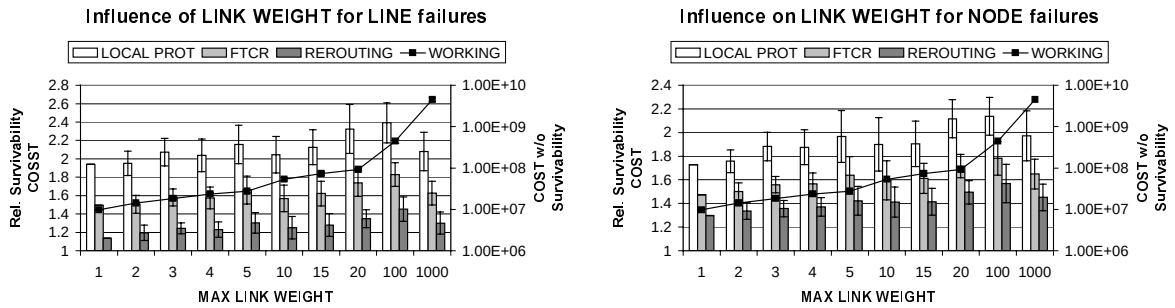

Figure 14: influence of link weights

6.3 Influence of traffic pattern

Up till now, we have been generating random traffic matrices, as described in section 5, by allowing a non-zero demand for each node pair. Such a uniform traffic pattern represents a backbone network between important cities (e.g., the Qwest and Espire network being used in this paper).

However, a hubbed/star traffic pattern (where all traffic originates and/or terminates in one site) is also important to study and therefore it is presented in this subsection. Such a traffic pattern may represent the gateway to the Internet in a MAN or a residential ISP network. But it is also an approximation of the current situation of European backbone networks, where most traffic is coming from the gateway to the USA [21].

The experiment in this subsection is based on one demand matrix $[d_{ij}]$ (with scaling factor 1000) on the Large Espire Network, obtained in section 5. For each node k in the Large Espire Network, the simulations were repeated while assuming that particular node being the gateway. Therefore, the demand matrix $[d_{ij}]$ was modified into a matrix $[d_{ij}]^{\text{bidir}}$, $[d_{ij}]^{\text{from}}$, $[d_{ij}]^{\text{to}}$, by only keeping the matrix elements d_{ik}^{bidir} and d_{kj}^{bidir}, d_{kj}^{from}, and d_{ik}^{to}, $i, j \quad V$ respectively and setting all other elements to zero. The results for only line (type I) or only node (type II) failures are shown in Figure 15. These drawings show two important results.

The additional cost for survivability relative to the cost for a network without survivability is significantly lower for supporting flows in both directions. This is mainly due to the fact that in the latter case there is a

higher potential to share spare capacity between different failures (e.g., this is easily seen for 1:1 local protection, when considering the backup LSPs from or to the common node for two adjacent lines).

The other conclusion is that only for FTCR the additional cost for survivability is significantly lower for traffic only from the gateway compared to traffic only towards the gateway (note that FTCR is typically asymmetric, which is in clear contrast with the other two schemes). The main reason is that for the case with only traffic towards the gateway, the LSRs (immediately upstream of the occurring failure) calculate only one rerouting path (towards the single destination), leading to a highly concentrated rerouting traffic pattern. This is in contrast with the case with only traffic coming from the gateway where every LSR calculates rerouting paths for several destinations, leading to a rerouting traffic pattern which is more spread out over the network. This more spread-out character enhances the reuse of spare capacity in case of other failures.

Figure 15: hubbed/star traffic pattern

6.4 Single-path versus multi-path (re)routing

Our whole MPLS study has been relying on the fact that all traffic between two LSRs was routed or rerouted along a single path. We mentioned already that IP inherently is reliable, through the deployment of dynamic routing protocols. An example of such a routing protocol is OSPF, which has the possibility to deal with multiple shortest paths (this is the Equal-Cost Multi-Path (ECMP) feature of OSPF [22]). An important advantage of OSPF is that it contains a single routing-table entry per interface with the lowest distance towards the destination (potentially more than one interface per router) and thus there is no need to set-up a path (or equivalent of a LSP) for each shortest path. Note that a router does not care about how traffic is spread in other nodes of the network. Thus, OSPF is very suitable for a multi-path.

An important question in the context of our paper is: “how much do you have to pay (in terms of cost for additional survivability), when deploying an MPLS network, compared to a traditional IP network?” For this

purpose all experiments discussed above, have been repeated for an equal cost multi-path context. A comparison is made between the MPLS case (where a single shortest path is restored by another single shortest path after the failure occurrence) and the OSPF case (where each demand is spread over **all** shortest paths and after a failure forwarded over **all remaining** shortest paths).

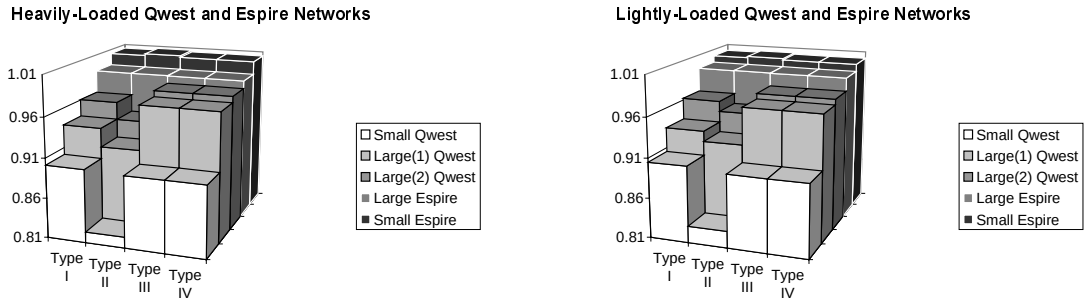

Figure 16: Qwest and Espire Networks (ratio of survivability cost: multi-/single-path)

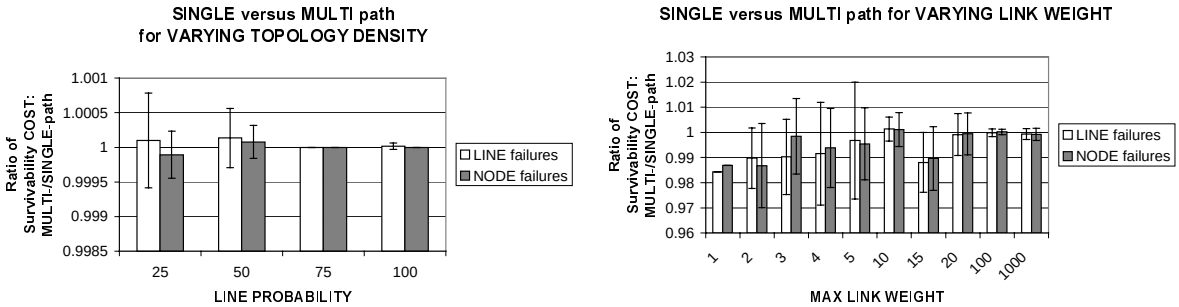

Figure 17: influence of topology (left: connectivity; right: link weights)

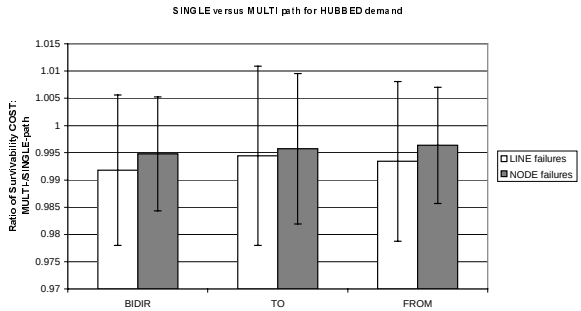

Figure 18: hubbed/star traffic pattern

Figure 16, Figure 17 and Figure 18 compare the relative cost for additional survivability for a single- and multi-path context. Note that these plots do not include any cost for working traffic, since it is the same for both single- and multi-path contexts (since the cost to use an LSP is the same for all shortest LSPs). These drawings show that generally (but not always) multi-path is slightly cheaper than single path. For large values of the link weights

(see right side of Figure 17) the ratio of the survivability cost becomes 1 and the standard deviation is significantly reduced for both line and node failures. This is of course because it becomes hard to find more than one shortest path on a network with uniform distribution up to a large maximal link weight (the lengths of the shortest path should be equal, even a difference of 1 is not allowed).

7. Conclusions

We started this paper by explaining how MPLS and more specifically MPLS recovery techniques fit in the context of an IP-over-WDM network. Afterwards, we have given an overview of the current proposal for MPLS recovery and introduced our own scheme called Fast Topology-Driven Constrained-Based Rerouting (FTCR). The main part of the paper compared the additional capacity requirements for this technique, compared to the 1:1 Local Protection and Rerouting schemes.

The most important conclusion from our study is that Rerouting is the cheapest solution, 1:1 Local Protection the most expensive and FTCR is situated somewhere in between. Although no tendency or relation could be derived, this study showed that the topology significantly influences the capacity requirements for added survivability. Multi-path routing was found to be a little cheaper than single-path routing, but the profit tends to decrease for increasing link weights. The traffic pattern was found to be important, although the amount of traffic has a minor influence on the results. The asymmetrical nature of FTCR was shown in the case of a traffic pattern where traffic only originates or terminates in a single node: FTCR was proved to be cheaper in the most realistic case where all traffic originates in a gateway or server farm.

Acknowledgement

Part of this work has been supported by the European Commission through the IST-projects LION and TEQUILA and by the Flemish Government through the IWT-project ITA/980272/INTEC and an IWT-scholarship.

References

[1] U.S. Dept. of Commerce, The Emerging Digital Economy, http://www.ecommerce.gov/emerging.htm, (1998).

[2] X. Xiao, L.M. Li, Internet QoS: a Big Picture, IEEE Network Magazine (March/April 1999), pp. 8-18

[3] Y. T'Joens, et al., Resilient Optical and SONET/SDH-based IP networks, Proc. of Workshop on Design of Reliable Communication Networks (Munich, Germany, April 2000), pp. 255-260.

[4] R. Ramaswami and K. Sivarajan, Optical Networks: a Practical Perspective, Morgan Kaufmann, (1998).

[5] B. Mukherjee, Optical Communications Networks, McGraw-Hill, (1997).

[6] K. Struyve, et al., Application, Design and Evolution of WDM in GTS's Pan-European Transport Network, IEEE Communications Magazine, Vol. 38, No. 3, (2000), pp. 114-121.

[7] P. Lagasse, et al.: Photonic Technologies in Europe, Horizon - Infowin (ACTS), Telenor AS R&D, (1998).

[8] P. Bonenfant, A. Rodriguez-Moral, Optical Data Networking, IEEE Communications Magazine, Vol. 38, No. 3, (2000), pp. 63-70.

[9] N. Ghani, S. Dixit and T.-S. Wang, On IP-over-WDM Integration, IEEE Communications Magazine, Vol. 38, No. 3, (2000), pp. 72-84.

[10] R. Callon, et al., A Framework for Multi-Protocol Label Switching, IETF Internet Draft <draft-ietf-mpls-framework-05.txt>, (1997), work in progress.

[11] G.-S. Kuo, Multi-Protocol Label Switching, special issue of IEEE Communications Magazine, Vol. 37, No. 12, (1999).

[12] D. Awduche, et al., Multi-Protocol Lambda Switching : Combining MPLS Traffic Engineering Control With Optical Crossconnects, IETF Internet Draft <draft-awduche-mpls-te-optical-01.txt>, (1999), work in progress.

[13] N. Ghani, Lambda-Labeling: a Framework for IP-over-WDM using MPLS, Optical Networks Magazine, Vol. 1, No. 2, (2000), pp. 45-58.

[14] "A Method for Setting an Alternative Label Switched Paths to Handle Fast Reroute", work in progress, internet-draft March 2000: http://infonet.aist-nara.ac.jp/member/nori-d/mlr/id/draft-haskin-mpls-fast-reroute-03.txt

[15] "A Path Protection/Restoration Mechanism for MPLS Networks", Changcheng Huang et al, work in progress, internet-draft March 2000: http://search.ietf.org/internet-drafts/draft-chang-mpls-path-protection-00.txt

[16] "Protection/Restoration of MPLS networks", Makam et al, work in progress, internet-draft October 1999: http://search.ietf.org/internet-drafts/draft-makam-mpls-protection-00.txt

[17] "Framework for MPLS based recovery", Makam et al, work in progress, internet-draft March 2000: http://search.ietf.org/internet-drafts/draft-makam-mpls-recovery-frmwrk-00.txt

[18] P. Van heuven, et al., Recovery in IP based networks using MPLS, Proc. of IEEE Workshop on IP-oriented Operations & Management (Cracow, Poland, September 2000)

[19] http://www.espire.net/service_locations/network_map.cfm

[20] http://www.qwest.com/about/inside/network/nationip.html

[21] Lieven Vanhaverbeke, "Studie van ringnetwerken voor het transport van IP-verkeer" (Dutch), Master Degree Thesis, Ghent University (July 2000)

[22] “OSPF version 2”, J. Moy, RFC1247, July 1991