A Novel Shared Segment Protection Method for Guaranteed Recovery Time

János Tapolcai*, Pin-Han Ho†, Dominique Verchere‡ and Tibor Cinkler*

*Dept. of Telecommunications and Media Informatics, Budapest University of Technology and Economics
Magyar tudósok krt. 2, Budapest, H-1117, Email: {tapolcai, cinkler}@tmit.bme.hu
†Department of Electrical and Computer Engineering, University of Waterloo, Canada, N2L3G1, E-mail: pinhan@hp3.uwaterloo.ca
‡Alcatel, Route de Nozay, 91460 Marcoussis, France, Email: Dominique.Verchere@alcatel.com

Abstract—Shared Segment Protection (SSP), compared to Shared Path Protection (SPP) or Shared Link Protection (SLP), provides an optimal protection configuration, since SSP can increase the number of connections sharing the same protection segments and can reduce the restoration time in case of single link failure.

This paper provides a thorough study on SSP under the GMPLS-based recovery framework, where an effective survivable routing algorithm for SSP is proposed, called Shared Segment Protection (SSP) algorithm. The main advantage of the SSP algorithm is to reduce the high computation complexity in solving the ILP formulation first introduced in [1]. With an efficient iterative approach the design space is significantly reduced by excluding all the links that result intolerably long routes.

The tradeoff between the price (i.e., cost representing the amount of resources, and the blocking probability) and the restoration time is extensively studied by simulations on three networks with highly dynamic traffic. It is demonstrated that the SSP algorithm can be a powerful solution in the GMPLS-based recovery with a stringent delay upper bound for achieving high availability and restorability of the transport services. The comparison among the three protection types further verifies that SSP can yield significant advantages over SPP and SLP.

I. INTRODUCTION

Recently Internet Engineering Task Force (IETF) has made a lot of contributions to extend its Generalized Multi-Protocol Label Switching (GMPLS) protocol suite to satisfy the requirements of Automated Switching Optical Network architecture (ASON) specified by ITU-T. On the one hand GMPLS framework specifies all the protocol capabilities in terms of signaling such as Resource reSerVation Protocol with Traffic-Engineering extensions (RSVP-TE), routing such as Open Shortest Path First with Traffic Engineering extensions (OSPF-TE) and link management such as Link Management Protocol (LMP). On the other hand, ITU-T ASON recommendations specify the architecture and requirements for the Automatic Switched Transport Network as applicable to carrier transport networks.

The label switching concept introduced in the Multi-Protocol Label Switching architecture (MPLS) is generalized to all switching technologies in GMPLS architecture. A label can be defined as a synthesis of connection-oriented and connectionless technologies. GMPLS extends the label switching capabilities to network elements with a non-packet based forwarding engine, from packet switching capable (PSC) technologies to fiber switching capable (FSC) technologies including Layer 2 switching capable (e.g. Ethernet, ATM, Frame Relay), time division switching capable (e.g. SDH/SONET PDH, G.709), lambda switching capable (e.g. wavelength (CWDM, DWDM), waveband) and Fiber switching capable (e.g. ports). Within the GMPLS architecture, the connections between the ingress and egress nodes are referred to Label Switched Paths. The traffic flow/fing along a label-switched path (LSP) is defined by the label applied at the ingress node.

GMPLS has been recognized as the most promising control plane framework for the next generation carrier networks enabling a uniform and simpler management function for heterogeneous networks. One of key requirements of network service providers for designing a transport network is to ensure survivability in the case of a network failure. Network outages are caused by a variety of events that can lead the network status into unpredictable states. The survivability refers to the ability to maintain the consistent service level agreement in the case of a network failure. The survivability can rely on the uses of different recovery mechanisms (i.e. protection and restoration) implemented in the transport network. GMPLS-based recovery is a suite of failure protection and restoration mechanisms defined under the GMPLS framework, which is expected to provide complete solutions for achieving QoS-aware protection and restoration in a Data-centric heterogeneous network. The GMPLS-based recovery mechanisms are typically categorized in functions of the recovery moment. Recovery mechanisms are typically categorized based on the time – relative to the time of failure – at which path computation, signaling, resource selection and allocation are performed. This is illustrated schematically in Figure 1. The term “Dynamic re-routing” is used when the working connection (or equivalently working LSP) is dynamically recoverable using (non pre-planned) ingress-egress re-routing. Pre-planned re-routing without extra-traffic is used when protecting LSPs are provisioned at the control plane level only (a.k.a. soft-provisioned). Finally, the term “Protection” is used if the protecting connections are either fully provisioned.
the success rate for establishing connections with shared segment protection and average cost in terms of network resources (i.e. bandwidth requirements), based on which a comparison among Shared Segment Protection (SSP), Shared Link Protection (SLP) and Shared Path Protection (SPP) under different restoration time constraints is made.

This paper is organized as follows. In Section II, the problem of Shared Segment Protection along with the associated modeling techniques are defined. In Section III, the heuristic approach to efficiently compute the ILP formulation for proposing the optimal SSP solutions including the constraints on restoration time is presented. The original SSP ILP [1] is presented in Appendix A. In Section IV, the performance of SSP algorithm is described, and the results are compared with that of SLP and SPP. Section V concludes this paper.

II. PROBLEM DEFINITION

The segment protection configuration is used to provide recovery over a portion of a connection. The objective of
In order to estimate the amount of spare capacity, that can not be shared with other connections and should be allocated along protection routes a matrix of size $|A| \times |A|$ called spare provision matrix is introduced [4]. It is denoted as $S$ and its entries are denoted as $s_{e,a}$, where $e = 1 \ldots |A|$, $a = 1 \ldots |A|$. Each entry $s_{e,a}$ represents the amount of non-sharable spare capacity along arc $e$ for the protection path segment if the corresponding working segment is involved in the $a$-th arc (see also Figure 3). The condition for routing $W$ with a bandwidth $b$ along arc $a$ is $f_a \geq b$, while the feasible condition for the $k$-th protection path segment to pass through arc $e$ is $f_e + v_e \geq b + \max_{a \in W_k} s_{e,a}$ for $\forall k$ [1].

Then the total recovery time ($t_r$) can be modeled as follows:

$$
t_r = t_1 + t_2 + t_3 + t_4$$

where $f$ is the number of TDM LSPs interrupted by the link failure, and for simplicity reasons it is limited to a maximum of 100 LSPs in this study.

Variable $n$ denotes the number of OXC nodes from the upstream node adjacent (including it) of the failure to the branch node of the recovery segment.

The link failure detection is performed at the Line Card interface of the SDH/GMPLS Cross-Connect that triggers the failure node states, it is supposed fixed in this model.

The failure indication signal processing time at the adjacent node interface is denoted by $d_1$. It is set to 10 $\mu$s per TDM LSP. Similarly to [5] for the $f$ TDM LSPs, the total time required to initiate the failure notify message is $f \cdot d_1$.

The time required to forward the failure notify message at each intermediate node controller between the upstream adjacent node of the link failure until the branch node is denoted by $d_2$. It is set to 1 $\text{ms}$ per node.

The length of the $i$-th link of the segment after the branch node and before the merge node is denoted by $l_i$ and the propagation delay on the optical links denoted by $p$ and it is set to 5 $\mu$s per km.

We assume that the link failure indication messages of the affected TDM LSPs are forwarded sequentially at each node interface. Therefore, the evaluation of the upper bound on the link failure notification time must consider the summation of the three terms $f \cdot d_1 + n \cdot d_2 + \sum_{i=1}^{n-1} l_i \cdot p$.

The terms $|P_k|$ is the number of links on the $k$-th protecting segment of the failed $k$-th working segment i.e. the number of links on the $k$-th protecting segment between the recovery branch node and the recovery merge node.

The time required to process the signaling message that triggers the cross-connection of the switching fabric of the OXC at each intermediate node from the branch node to the merge node is denoted by $d_3$. This time is set to 10 ms per node.

Finally $c_1 \cdot |P_k| + c_2$ is the total time to configure, test, and setup the switching fabric of each OXC node (switching time) along the protecting segment route. $c_2$ is the time to complete the cross-connection of the merge node. $c_1$ is assumed to be 10ms and $c_2$ is 20ms.

Equation (1) can be re-ordered as:
\[ t_r = (t_1 + f \cdot d_1 + c_2) + \\
+ \left( n \cdot d_2 + \sum_{i=1}^{n-1} l_i \cdot p \right) + \left( |P_k| \cdot d_3 + |P_k| \cdot c_1 + 2 \cdot \sum_{i=1}^{|P_k|} l_i \cdot p \right) \]

(2)

The expression \( t_1 + f \cdot d_1 + c_2 \) is assumed to be less than \( t_c \), where \( t_c=21\text{ms} \). Two link overhead parameter are defined as \( \zeta^w_i = d_2 + l_i \cdot p \) and \( \zeta^p = d_3 + 2 \cdot l_i \cdot p + c_1 \) for each link \( i \), and since \( n \leq |W_k| \), equation (2) can be further rewritten as:

\[ t_r < t_c + \sum_{i=1}^{|W_k|} \zeta^w_i + \sum_{i=1}^{|P_k|} \zeta^p \]

(3)

A global parameter \( \zeta_{max} \) is the maximum allowable restoration time for the corresponding LSP defined in the service level agreement, and is the maximum restoration time of all the self-healing units.

III. THE SSP ALGORITHM

A. An Extended ILP formulation

An overview on the ILP formulation for implementing ILP for an optimal (or least-cost) SSP solution is given in Appendix, which has been reported in [1]. The solution yields the least-cost working and protection path segments, and the allocation of the branch and merge node-pair of each protection domain.

In order to address the constraint on the restoration time for a connection, the size of each protection domain for the connection must be upper-bounded. Therefore, the following constraint is appended:

\[ \sum_{v \in A_w} \zeta^w_v \cdot x^k_v + \sum_{v \in A_p} \zeta^p_v \cdot y^k_v \leq \zeta_{max} \quad \text{for } 1 \leq k \leq k_{max}, \]

(4)

where \( \zeta^w_v \) and \( \zeta^p_v \) were introduced in Section II-A. Equation (4) brings \( k_{max} \) additional constraints to the ILP formulation.

Note that the optimal SLP can be derive by adding Equation (5) as additional constraints to the ILP formulation of SSP:

\[ \sum_{v \in A_w} x^k_v \leq 1 \quad \text{for } 1 \leq k \leq k_{max} \]

(5)

B. A Heuristic of Improving the runtime of solving ILP

It is clear that the computation time in solving the ILP of [1] could be intolerably long and strongly depend on the maximum number of segments (parameter \( k_{max} \) in the formulation) to be considered on the network. Thus, selecting a proper value of \( k_{max} \) can significantly improve the computation efficiency. Furthermore, during routing the working and protection path segments across a specific source-destination pair, some links in the network are very unlikely or definitely unfeasible to be taken, which can be simply excluded in the searching process for achieving better computation efficiency with small probability of losing quality. The above two observations motivate us to develop the proposed heuristic - SSP algorithm, that is committed to speed up solving the ILP formulation.

Figure 4 shows an illustration of the main phases of SSP algorithm. First, a pre-calculation mechanism is devised to calculate a feasible (or a good approximation on the) solution, which gives an upper-bound on the total cost. Second, based on the pre-calculated solution it estimates the value of \( k_{max} \) and excludes those arcs relatively unlikely to be taken by the working and protection segments. The graphs for solving W and Q are denoted as \( A_w \) and \( A_p \), respectively. Finally, the ILP SSP is launched on the reduced residual graphs with proper \( k_{max} \) parameter.

Figure 5 shows the flowchart on the first phase of the SSP algorithm, where an upper bound on the cost of the solution (denoted as \( c^{ub} \)) is derived by any arbitrary heuristic algorithm for SSP. In this paper we derive an arbitrary SSP solution using the following approach: the ILP problem for SPP [7] is solved with an additional constraint on restoration time:

\[ \sum_{v \in A_w} \zeta^w_v \cdot x_v + \sum_{v \in A_p} \zeta^p_v \cdot y_v \leq \zeta_{max} \]

(6)

where \( x_v \) and \( y_v \) are binary flow indicators of the working and protection path, respectively. If the working path intersects with the protection path, the nodes of intersection are simply treated as branch and merge nodes, by which a feasible solution is identified.

At step (1) the algorithm tries to find a feasible solution and a valid upper bound on the total cost. If the working path is one hop the searching process is terminated in step (3) because the derived solution is treated as the optimal or near-
If no feasible solution of SPP is derived in (1), a heuristic approach is developed to define $c_{ub}$ described as follows. In (4) of the flowchart, we solve the SPP again with the restoration time constraint relaxed (i.e., Equation (6) is eliminated from the ILP of [7]). If succeed, we go to (5) and define $c_{ub}$ according to our rule of thumb that is $c_{ub} = 2 \cdot c_{pre}'$, where $c_{pre}'$ is the cost of the relaxed solution. In this case $k_{max}$ is set as the number of hops in the derived working path. If solving the SPP with the relaxed restoration time constraint still fails, we go to (6) and calculate the shortest path, and in (7) we define $c_{ub} = 3 \cdot c_{sp}$, where $c_{sp}$ is the cost of the derived shortest path. We estimate $k_{max}$ similarly as the number of hops in the shortest path. Obviously, if the shortest-path algorithm fails to find any solution, we go to (8) to halt the algorithm.

Figure 6 shows the second phase of the algorithm, where the graphs $A_w$ and $A_p$ are derived as follows. Let us define a detour factor of each arc assigned to the connection and network state, denoted as $\xi_{i,j}^w$ and can be calculated with the following formula:

$$\xi_{i,j}^w = sp(G_w,s,i) + c_{i,j} + sp(G_w,j,d)$$  \hspace{1cm} (7)

where $sp(G_w,s,i)$ represents the cost of the shortest path in $G_w$ between $s$ and $i$; $c_{i,j}$ represents the cost of arc $(i,j)$, and $sp(G_w,j,d)$ is the cost of the shortest path between $j$ and $d$ (see also Figure 7).

The detour factor shows the minimal detour compared with that of the shortest path if the working path passes through arc $(i,j)$. Let us define $c_{lbp}$ as the lower bound on the cost of protection paths. $c_{lbp}$ is set to 0 at the beginning of the phase. Let us define $\xi_{i,j}^w$, which gives an upper bound on the detour of the optimal solution. In the first step $\xi_{i,j}^w$ can be set to $c_{ub} - sp(G_w,s,d) - c_{lbp}$ in the case that the pre-calculated solution was feasible. As a consequence, all the arcs with $\xi_{i,j}^w > \xi_{i,j}^w$ can be removed from $A_w$. An other necessary condition is the General Shared Protection (GSP) feasibility condition. In GSP feasibility condition it is checked if the failure of the link can be restored. It can be restored if after the failure there is a feasible protection path.

Another necessary condition is the General Shared Protection (GSP) feasibility condition on each arcs, which is taken to determine if the failure of arc $a$ can be restored by the capacity (including the amount of free and sharable spare capacity) along arc $e$. It can be restored if after the failure there is a feasible protection path.

III.1 Definition GSP test of arc $a$ is true if there is a path $P$ between $s$ and $d$, such that $\forall e \in P \quad f_e + v_e - s_{e,a} \leq b.$
It is easy to verify that the working route has only arcs where GSP test holds. As a result, in (11) of the flowchart, \(A_w\) contains all the arcs with \(c_{w, e}^p \leq \xi_{\text{max}}\), \(b \leq f_e\), and those which satisfy the GSP test.

After \(|A_w|\) is reduced, the lower bound on the cost of protection path can be derived. First a lower bound on the cost of each arc taken by protection path is calculated. Obviously, the working path will take at least one arc of \(A_w\). Since \(A_w \subseteq A\), a better upper bound on the sharable spare capacity on link \(e\), \(m_e\), can be derived as shown on (12) of the flowchart:

\[
m_e = \min_{a \in A_w} v_e - s_{e,a}
\]

with the information of \(m_e\), \(A_p\) can be defined as the graph containing all the arcs with \(b \leq v_e + m_e\). Thus, a lower bound on the cost of each arc taken by the protection path can be determined as:

\[
e_{e}^p = \max \left\{ \frac{b - v_e + m_e}{b}, 0 \right\}, \quad c_e
\]

The above relation holds since \(v_e - m_e\) is a lower bound of the sharable spare capacity such that the term \(\max \left\{ b - v_e + m_e + b, 0 \right\} / b\) gives a lower bound on the ratio of the sharable capacity. With this, we can derive the detour factor of each arc in \(A_p\) (denoted as \(\xi^p_{i,j}\)), which is specific to the connection request and the current link-state:

\[
\xi^p_{i,j} = sp(G_p, s, i) + c_{i,j}^p + sp(G_p, j, d)
\]

where \(sp(G_p, s, i)\) and \(sp(G_p, j, d)\) represent the total cost of the shortest path in \(G_p\) between the source node and node \(i\) and between node \(j\) and the destination, respectively, and \(c_{i,j}^p\) is the cost of \((i, j)\) in \(G_p\). With \(sp(G_p, j, d)\) we can derive a better lower bound on \(c_{i,j}^p\). Note that the arcs with \(\xi^p_{i,j} > \xi_{\text{max}}\) are not included in \(A_p\) since the desired protection path can not feasibly pass through any of them due to its large amount of detour. Therefore, as shown in (14) of the flowchart, \(A_p\) is composed of all the arcs where \(\xi^p_{i,j} \leq \xi_{\text{max}}\) and \(b \leq f_e + m_e\). If the lower bound on the cost of protection path \(c_{i,j}^p\) has been improved, the algorithm switches back to (10) of the flowchart to further reduce the arcs of \(A_w\) and \(A_p\) in the next iteration.

With the SSP algorithm, the runtime of solving the ILP of SSP can be significantly reduced while the quality of the result is almost always guaranteed or very close to the optimal one. We will further verify the heuristic in the following section.

IV. VERIFICATION

Experiments are conducted to verify the SSP algorithm using CPLEX 8.0 on Sun Ultra 80 workstation with 2GB memory and several Linux workstations. As a comparison among Shared Link Protection (SLP), Shared Path Protection (SPP) and Segment Shared Protection (SSP), simulations are conducted on two simulated network topologies and two random topologies. The first one is the pan-European fibre-optic network defined by IST project LION and COST action 266 as [8], which has 28 nodes and 57 bi-directional links as shown on Figure 8. The second one is based on the US NSF Network [9] with 26 nodes and 43 bi-directional links as shown in Figure 9. For both networks a traffic matrix in year 2005 is estimated according to [10], which is a slightly improved model than that provided in [11]. A dynamic traffic pattern is generated according to the traffic matrix such that an Interrupted Poisson Process and Pareto inter-arrival times are integrated together with exponential holding time. Simulation
other is the average cost using the target function of the ILP.

Figure 11 shows an illustrative example, where simulation is conducted on the network N16 (with 16 nodes and 27 bi-directional links) with a heavy load (at average total network utilization of 73%). The graph with the selected $s - d$ pair is drawn on the right-bottom corner of the chart. The $y$ axis represents the cost of the connection yielded by the target function of the ILP. It is clear that as the restoration time constraint is getting more relaxed, the performance impairment (in terms of the cost) for each connection request is reduced.

As a comparison, the optimal SPP is evaluated using the ILP formulation of Section III, where the derived result for each connection is marked by triangles on the charts. Note that the cost of solving the SPP is not smaller than that of the SSP case since SPP is a special case of SSP (with $k_{max} = 1$). Shared Link Protection (SLP) is also a special case of SSP where each working segment consists of only one link.

A detailed overview on SLP and SPP can be seen in [12]. The simulation results of SLP are marked with boxes on the charts, in which the length limitation on the protection path can be addressed using Equation (6). The selected node-pairs are selected far from each other such that it illustrates the increase of the cost of SSP if we gradually sharpen the restoration time constraint. Figure 13 shows the results using a 61-node network (shown in Figure 12) with a light traffic load for comparing the three types of protection in terms of average cost and the success rate under different restoration time constraints. Results of 100 random connection requests are averaged for each data. It is shown that an average of approximately 10% reduction in the cost can be achieved with SSP over the case of SPP if the restoration time constraint is relaxed to 13 hops. The average cost in the SSP and SLP cases increase when the restoration time constraint is sharpened, as it was expected. However, the average cost drops dramatically when the restoration time constraint is loose since at this moment most of the long connections (with large cost) are blocked, and only the short connections (with small cost) can be allocated.

It can also be observed that SSP can yield much higher success rate for those connection requests under a tight restoration time constraint than that with SPP at the expense of taking moderately extra cost, as shown in Figure 13. SLP yields the highest average cost in all cases with a close success rate with SSP, and is seen less competitive with the other two types of protection. We further extend the simulation study on ERNet and NARNet with the same traffic pattern and simulation environment. The simulation results of NARNet with a high traffic load are shown in Figure 14, where the major advantage demonstrated in using SSP is that the success rate outperforms that of the SLP case by 2 times or more at the expense of taking a little bit higher cost (as shown in Figure 14(b)). For SLP, the overall performance is far outperformed by the other two cases although it can guarantee the shortest restoration time. On Figure 16 the results of the ILP SSP and the results of the SSP algorithm are close to each other in the whole range of the restoration time limit. It demonstrates that the performance of SSP is not significantly degraded by reducing the design space at the pre-calculation step. To summarize the simulation results, SSP gave a good compromise in terms of cost versus restoration time, while SLP requires an average of 10-20% additional capacity allocation compared to SSP or SPP.

V. CONCLUSIONS

This paper studies optimal configuration for setting the shared segment protections (SSP) on the working LSPs to be established, in which an ILP is formulated such that the branch-merge node-pairs in each Working - Protecting
Fig. 13. Performance impairment by addressing the restoration time constraint using the 61-node network at light load (19%).

Fig. 14. Performance impairment by addressing the restoration time constraint with NARNet at high load.

Fig. 15. Runtime in solving the SPP, SLP with pre-calculation, SLP, SSP with pre-calculation, and SSP on N16 and ERNet.

Fig. 16. Average cost and success rate of SSP-optimal and SSP with pre-calculation step. For SLP there is an average of 3.3 gap.

Segment and the corresponding least-cost Working - Protecting segment-pair for a LSP request can be jointly determined in a single step. To improve the high computation complexity induced by solving the ILP, a novel heuristic called SSP algorithm is proposed, aiming to initiate a graceful compromise between the optimality and the computation time by employing a novel iterative arc reduction pre-calculation mechanism. Different from the ILP in [1], the demonstrated formulation considers the restoration time constraints, which meets the practical requirement of the successful optical transport network restoration. Extensive simulation efforts were conducted on NARNet and ERNet based on the estimated traffic pattern in year 2005 to compare the Shared Link Protection (SLP), Shared Path Protection (SPP) and Shared Segment Protection (SSP) in terms of average cost and success rate of setting up connections. We observe that SSP can initiate a graceful compromise between average cost and network availability.
under a wide range of restoration time constraints.

ACKNOWLEDGEMENTS

We would like to express our thanks to Dimitri Papadimitriou for the review of the recovery time model.

REFERENCES


APPENDIX

A. The ILP formulation for SSP

This section overviews the linear formulation for the segment shared protection problem presented in [1]. The main idea is to define a path $Q$, called mass protection path, which defines the route of each protection segment as well as the branch and merge nodes of working path (W). Similar to Suurballe’s [13] algorithm, $Q$ is composed of the reversed links along the working path and all the backup segments. A simple example is shown in Figure 17, where $Q$ is composed of (s-a-b-c-e-d). The first protection domain is formed by the working and protection segments (s-c-b) and (s-a-b), respectively; while the second is formed by (c-b-d) and (c-e-d), respectively. We allow the overlapping between the working segments of two neighbor protection domains in order to explore the largest design space so as to guarantee the optimality of the derived solution. Note that $Q$ may contain loops to reflect the fact that spare capacity sharing can happen between two protection segments of different protection domains. Variable $k_{max}$ is defined as a parameter of the ILP and represents maximum number of protection domains that can be possibly handled in the problem (it is $\leq n - 1$).

Three residual graphs are defined for solving this problem:

- $G_w(V, A_w)$ is composed of links with $b \leq f_a$ for $a \in A_w$ (in the following formulas $x$ is a binary, $\bar{x}$ is a real and $x^k$ are $k_{max}$ binary variables assigned to arcs of $G_w$), and describes the working segments.

- $G_p(V, A_p)$ is composed of all the links $b \leq f_e + \nu_e$ for $e \in A_p$ (in the following formulas $y^k$ is $k_{max}$ binary and $\nu^e$ is real variable assigned to the arcs of $G_p$) and describes the protection segments. We need this graph to record the spare link-state because working and protection paths take different suites of link-state with shared protection.

- $G_{p}'(V, A_{p}^{'})$ is composed of all the links in $A_p$ along with the links of $A_w$ in a reversed direction. The “reversed” arcs corresponding to $A_w$ in $A_{p}'$ are denoted as $(\bar{i}, \bar{j})$, and all the others (corresponding to $A_p$) are $(\bar{i}, \bar{j})$. The graph is assigned to $Q$ and handles the reverse arcs caused by the overlapping between $Q$ and $W$. (in the following formulas $y^k$ is an integer, $\bar{y}$ is a real and $y$ is a binary variable assigned to the edges of $G_{p}'$).

$x_a$ and $y_e$ are flow indicators of path $W$ and $Q$, respectively. The target function is as follows:

Minimize: \[ \sum_{a \in A_w} b \cdot c_a \cdot x_a + \sum_{e \in A_p} \left( c_e \cdot z_e + \varepsilon \cdot y^e \right) \]  \hspace{1cm} (11)

where $c_a$ is the cost per unit of working bandwidth to reserve arc $a$ and $z_e$ is a for the protection segments and represents the amount of capacity, which cannot be shared and need to be allocated for the protection paths.

The constraints are following:

We need the flow conservation constraint for the working (on variable $x$) and mass protection paths (on variable $y$), respectively. We need to formulate the above-mentioned...
properties of working and mass protection path: $x_{i,j}$ and $y'_{i,j}$ will be exclusive in terms of the SRGs they take. An arc can be taken by $y'_{i,j}$ in a reversed direction only if $x_{i,j}$ pass through it. Besides, each reversed arc can be used only once since the algorithm only allows two working segments overlapped. Thus, $Q$ is SRG disjoint from $W$ except for the reversed arcs of $W$. Note that, reversed arcs indicates the branch/merge nodes for each protection domain along $W$. A pair of variables, $\hat{x}_{a}$ and $\hat{y}'_{e}$, is assigned to each link along $W$ and $Q$, respectively, such that the first link from the source has a label of 1; and if a protection domain ends or starts at a node, the labels of the following arcs will be increased by 1. This labelling method is similar to that proposed in [14]. This is done by modified flow conservation constraints, where the following four situations are considered for all vertices (not the source or destination) taken by $W$: (a) $Q$ merges back to $W$; (b) $Q$ branches out of $W$; (c) $Q$ merges back and branches out of $W$; (d) otherwise. The amount of flow of $\hat{x}_{i,j}$ and $\hat{y}'_{i,j}$ at vertex $i$ along $W$ increases by 1 in the case of situations (a) and (b), and increases by 2 in the case of situation (c), and is unchanged otherwise. With $\hat{x}_{a}$ and $\hat{y}'_{e}$ link labels, the $k$th protection domain, respectively. By observing Figure 18 one can be easily verify that the value of $\hat{y}'_{e}$ on $Q$ of the first protection domain is 1; and in the second protection domain $\hat{y}'_{e}$ is 3; and in the $k$th protection domain $\hat{y}'_{e}$ is $2k-1$, thus $\hat{y}'_{e}=1$ only when $\hat{y}'_{e}=2k-1$. The value of $\hat{x}_{a}$ of arc $a$ taken by $W$ in the first protection domain is either 1 or 2, depending on whether or not there is overlapped arc(s) between the working segments of the first and the second protection domain; while on the links of the $k$th protection domain, we have $\hat{x}_{a}=2k-2$ on the non-overlapped links and $\hat{x}_{a}=2k-1$ on the overlapped links of the $(k-1)$th and the $k$th protection domain; we have $\hat{x}_{a}=2k$ on the overlapped links of $k$th and $(k-1)$th protection domain. Please refer to Figure 19 for an example demonstrating the definition of $x_{a}^{k}$ and $y'_{e}^{k}$.

The last constraints need to be defined are the SRG constraints setting the value of $z_{e}$ defined in the target function. It is considered using $S$ matrix, such that when link $a$ and $e$ is taken by the working and protection segments in the $k$th protection domain, respectively, the resultant amount of scaling (i.e., $z_{e}$) is at least $b-v_{e}+s_{e,a}$.

![Fig. 17. Design of mass protection path $Q$.](image1)

![Fig. 18. An example showing the variables $x$, $y'$, $\hat{x}$, and $\hat{y}'$. It is zero for the variables of the arc not shown on the figure.](image2)

![Fig. 19. An example for the values of $x_{a,b}^{k}$ and $y_{a,b}^{k}$. For any link without a mark on the figure has a cost zero.](image3)