Yugoslav Journal of Operations Research
26 (2016), Number 3, 343–360
DOI: 10.2298/YJOR150730001P

RESILIENT OVERLAY DESIGN IN DWDM SYSTEMS

Cecilia PARODI
Facultad de Ingeniería, Universidad de la República–Uruguay
cparodi@fing.edu.uy

Franco ROBLEDO
Facultad de Ingeniería, Universidad de la República–Uruguay
frobledo@fing.edu.uy

Pablo ROMERO
Facultad de Ingeniería, Universidad de la República–Uruguay
promero@fing.edu.uy

Carlos E. TESTURI
Facultad de Ingeniería, Universidad de la República–Uruguay
ctesturi@fing.edu.uy

Received: July 2015 / Accepted: November 2015

Abstract: The goal of this work is to design a minimum cost resilient overlay network, where a data network is on top of a transport network. Two major challenges are addressed. On one hand, a single failure in the transport network causes multiple simultaneous failures; on the other, the multicommodity flow must respect integrality. An integer programming formulation is presented to design an overlay, meeting the previous constraints. We prove that the problem belongs to the class *NP*-Hard. Then, a decomposition approach is introduced, where the problem is solved in two steps by means of relaxations of the original formulation. Experiments carried out with real-life instances, coming from the Uruguayan telecommunication operator, show that the approach is competitive with respect to previous metaheuristics, to know, Tabu-Search (TS) and Variable Neighborhood Search (VNS). A modest percentage of cost-reduction is achieved in some instances, which means millionaire savings in practice.

Keywords: Network Survivability, Network Optimization, Overlay.

MSC: 68M10, 90B10, 90C10.

1. INTRODUCTION

The increasing importance of the telecommunications services pushed most companies to deploy optical fiber networks. Since the volume requirements of the telephony service were low, the design process was guided by cost and availability [1].

At its early stages, the Internet service was implemented making use of the existing transport infrastructure —Internert Protocol (IP) nodes connected through existing Synchronous Digital Hierarchy (SDH) or Synchronous Optical Networking (SONET) protocols. Since SDH/SONET networks are redundant, the network rarely suffered of unplanned topology changes. However, the exponential growth of the Internet volume demanded for networks of higher capacity. This demand caused the development of Dense Wavelength Division Multiplexing (DWDM) technology. Instead of using several optical fibers to connect SDH/SONET nodes, DWDM allows multiplexing many connections over one single cable of optical fiber using different wavelengths. DWDM gained popularity in telecommunications companies since it allowed them to expand the capacity of their networks without laying more fiber. Nowadays, DWDM is the dominant network technology in high-capacity optical backbone networks.

The main drawback of these technologies is the existence of multiple layers (overlays). Multiple layers imply higher cost in operation and maintenance. As a response to the previously described issues, the industry added several features to the traditional IP routers. The most relevant are: Multi-Protocol Label Switching (MPLS), traffic engineering extensions for dynamic routing protocols (e.g. OSPF-TE, ISIS-TE) and fast reroute algorithms (FRR). This new technology bundle is known as IP/MPLS. Since IP/MPLS allows recovering from a failure in about 50ms (a benchmark comparable to SDH/SONET), it opens a competitive alternative against traditional protection mechanisms, allowing savings coming from the elimination of the intermediate transport layer. Since IP/MPLS allows the elimination of an intermediate layer by managing Internet traffic natively making possible a much easier and cheaper operation for Virtual Private Network (VPN) services, it is gaining relative importance every day.

The related literature offers several mathematical formulations of multilayer network design [2,3,4,5]. Those papers do not reveal a design with strong robustness requirements.

The goal of this work is to design a minimum-cost resilient IP/MPLS network to be directly deployed over an existing DWDM infrastructure. The major cause of concern is that a single point-of-failure in the physical layer implies multiple failures in the overlay (i.e., data layer). Therefore, a different routing strategy (i.e., different muticommodity flow) should be considered for each possible single point-of-failure. This is our main challenge, in strong contrast with previous works in the field.

This paper is organized in the following manner. Section 2 briefly describes representative works in the area. Section 4 formally presents the problem with a mathematical programming formulation. The decision variables involve the topology together with link capacities and traffic routing. Section 5 shows that the problem belongs to the class of *NP*-Hard computational problems. As a consequence, a two-step solution is introduced in Section 6, following "Divide and Conquer". Numerical results in a case study of the Uruguayan operator are presented in Section 7. These results suggest millionaire savings with respect to previous heuristic solutions. Concluding remarks and trends for future work are outlined in Section 8.

2. BACKGROUND

Due to the growing of Internet demand, traffic and capacity are recurrent elements on most network design problems, particularly on multi-layer ones. However, what defines a multi-layer problem is the explicit existence of an overlay network, wherein demands are to be routed [6].

Alevras et al. develop an early representative model for overlay networks [7]. Their work effectively states the existence of two network levels, one responsible for connecting points physically, the other responsible for providing services to customers. Instead of a physical network whose links support many logical connections simultaneously, in this work a supply network was introduced, whose edges represent different types of link technologies to connect nodes, such as microwave, leased lines and others. Each technology has capacity bounds and respective costs. The model responds to a typical application of a cellular operator. The authors detail a MIP formulation, and linear relaxations to find solutions.

Although revolutionary at its time and convenient for its application, this model is not appropriate for modelling current high-speed Internet networks. Nowadays, the number of logical links is higher than the number of conduits, so a physical failure tears down several logical links simultaneously. Besides, IP/MPLS traffic must follow a single path between points instead of diverse and concurrent paths, and a failure is not dynamically protected in this model. A failure in a supply element would disconnect active calls flowing across it, and active clients should have to call again to re-establish them. This is probably the reason to impose traffic diversity. Current applications demand better protection mechanisms.

More recent works design a transport SDH/SONET network deployed over a DWDM optical network [2,8]. Raack et al. present a two-layer model with physical and logical levels [8]. The model integrates constraints to guarantee that certain demands can be protected. The objective is to minimize the total installation costs (optical fiber, node costs and logical link capacities). The model belongs to the class of *NP*-Hard problems [9]. It fits SDH/SONET technology, but could be improved by adding a ring entity, since the existence of two disjoint paths for a demand does not guarantee that they can be implemented under an automatic protection switching (APS) scheme, unless both match a sequence of logical rings all along their way. Another drawback of this model is the need to predetermine physical paths for logical links (i.e., predetermine light paths).

Other works precompute routes and assume the existence of an active/standby protection scheme on the logical network or the capability to dynamically change the route followed by the light path over the physical layer [3,4]. The simplest way for protecting logical links upon the physical layer is using APS mechanisms (i.e., statically protect logical links as a mean to protect demands). However, such protection would inherit inefficiencies. In order to dynamically reconfigure point-to-point light paths in the DWDM network coordinated with the IP/MPLS layer, an integration of both logical and physical control plan is necessary. For our purpose this is not possible since important portions of the physical layer are leased to third-party companies.

3. PROBLEM DESCRIPTION

A data (logical) network is built as an overlay of an existing transport network, where each node of the data network has a corresponding node on the transport network. Traffic demands among nodes are given in the data network, and they have to be routed in it, leading to the installation of data network capacities. Data network capacities define demands for the underling layer, leading to the installation of capacities in the transportation layer. Therefore, a demand among two nodes is routed through a path of links (virtual links) in the data network. Each of these links corresponds to a path of links in the transport network (physical links).

An example of the layered scheme is shown in Figure 1. Data network nodes A, B, F, and G have associated transport network nodes a, b, f, and g, respectively. At the data network, communication is demanded between nodes A and G, and between nodes F and B. Whereas the $A - G$ request is handled at the transport network trough the tunnel $a - d - g$, the $F - B$ demand is handled trough the tunnel $f - g - d - b$. Failure of link $d - g$ requires to relocate both tunnels.

During this construction, there are costs involved that should be minimized; these costs are dependent on the distance between nodes in the transport network and the capacity assigned to each selected link in the data network. Besides, in the transport network different types of technologies must be considered.

The data network found as a solution should have minimum cost and must be able to satisfy all the traffic requirements. Another important objective is that it must be resilient to simple failures in the transport network, meaning that when one link of the transport network fails, the traffic still has to be routed.

The concept of designing the data network includes basically two goals. First, the choice of those potential links that provides feasibility at the minimum cost. The traffic requirements should be met in the scenario where no transport link fails, and also in every faulty scenario where each transport link fails. Second, the capacities to these selected links should be assigned, which correspond to the

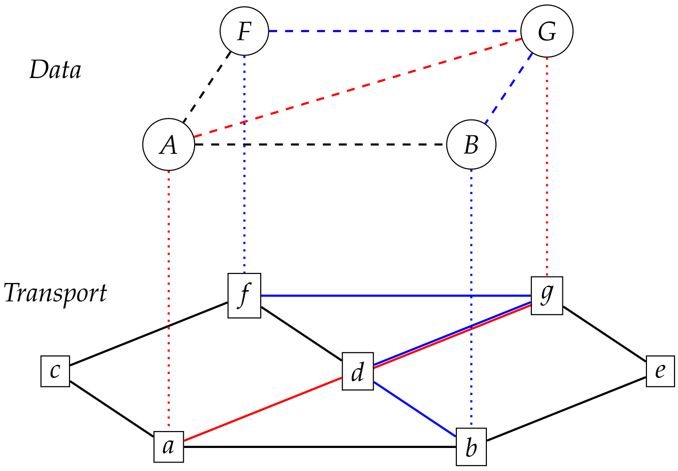


Figure 1: Example of data and transport networks architecture. Data network demands $A - G$ (red) and $F - B$ (blue) are handled trough the transport network tunnels $a - d - g$ and $f - g - d - b$, respectively.

supported technologies of the transport network. After that, for each data link established, the best path which this link should follow in the transport network must be given. Moreover, the question of how to route traffic requirements in every scenario of transport link failure and in the working scenario (the one where there are no failures) must be answered.

In order to solve this problem, all the transport network topology (nodes, links, distances between nodes) along with the possible technologies to be used should be known. Also, all the potential links between nodes of the data network and the traffic requirements must be given as an input. Finally, the statistical behavior of users traffic is necessary.

4. RESILIENT OVERLAY DESIGN MODEL

The data network is represented as an undirected graph $G_D = (V_D, E_D)$, where V_D is the set of nodes and E_D the set of potential data links. Each data link $e \in E_D$ should be dimensioned with a capacity $b_e \in \hat{B} = \{\hat{b}_0, \hat{b}_1, \dots, \hat{b}_{\bar{B}}\}$ that is technologically supported by the transport infrastructure. If a link does not become a part of the solution then its capacity is zero and stated by $\hat{b}_0$. All the traffic requirements between data nodes should be fulfilled. The traffic is denoted by $M = \{(\underline{M}_{kl}, \overline{M}_{kl}) | k, l \in V_D\}$ where $\underline{M}_{kl}$ corresponds to *committed* traffic, while $\overline{M}_{kl}$ represents *excess* traffic. Committed traffic must be exchanged all the time with minimum delay, even under any single link failure in the transport layer. On the contrary, excess traffic is typically the Internet traffic and is only available a fraction of time. This is modelled by a piecewise linear function z_Q that assigns the minimum required capacity of the links that carry that traffic demand [10].

The transport network is a two-node-connected graph $G_T = (V_T, E_T)$, where V_T is the set of nodes and E_T is the set of links. The traffic in the transport network is completely static. When a path in this network is associated with a link in the data network, then this path is fixed and it is the same for every fault scenario. The telecommunication structure is physically installed at *network stations*. In these stations there are transport nodes as well as data nodes. It will be assumed that in each network station there is at most one data node. The assumption made is that in these network stations there is an equipment of unlimited capacity that may use any of the available technologies. Unlike the data links, transport links are supposed to have unlimited capacity. If a transport link fails it affects every connection of the data network that uses that link. In each network station there is exactly one transport node. Given a data node, we can determine which transport node belongs to the same network station. Let us consider the function $tns : V_D \to V_T$ such that $tns(i) = v$ returns the transport node v located in the same network station as demand node i. In order to know which data links were affected by a transport link failure, it is necessary to know the paths that the data links follow in the transport network. The transport network routing is allowed in all possible paths in the graph G_T. We denote *flow configuration* to every subset of paths for a given pair of transportation nodes. Let $r : E_T \to \mathbb{R}$ be the *distance function* that assigns to each transport link its length, and $T : \hat{B} \to \mathbb{R}$ the *cost function* that associates a technology with its price per unit distance. With this, the cost is calculated as $r(e) \times T(b_e)$.

Following a node-link representation, the indexes are over the sets of nodes and links are represented as pairs of nodes, the variables are

- $x_{ij}^{kl} = 1$ if link $(i, j) \in E_D$ is used to route the traffic demand between nodes $k, l \in V_D$ and 0 otherwise,
- $x_{ij}^{kle} = 1$ if link $(i, j) \in E_D$ is used to route the traffic demand between nodes $k, l \in V_D$ when $e \in E_T$ fails and 0 otherwise,
- $y_{uv}^{ij} = 1$ if link $(u, v) \in E_T$ is used to route data flow between nodes $i, j \in V_D$ and 0 otherwise,
- $w_t^{ij} = 1$ if link $(i, j) \in E_D$ has capacity $\hat{b}_t \in \hat{B}$ and 0 otherwise, and
- $\chi^{ij} = 1$ if there is a link between the nodes $(i, j) \in E_D$ and 0 otherwise.

We are in conditions to introduce the Resilient Overlay Design Problem (RODP)

$$\min \sum_{i,j \in V_D} \left(\sum_{(u,v) \in E_T} y_{uv}^{ij} \cdot r(u, v) \right) \cdot T\Big(\sum_{\hat{b}_t \in \hat{B}} w_t^{ij} \cdot \hat{b}_t\Big) \tag{1}$$

$$\chi^{ij} = \sum_{\hat{b}_t \in \hat{B}} w_t^{ij}, \quad \forall (i,j) \in E_D, \tag{2}$$

$$\sum_{v:(tns(i),v)\in E_T} y^{ij}_{tns(i),v} \geq \chi^{ij}, \quad \forall i,j\in V_D, \tag{3}$$

$$\sum_{u:(u,tns(j))\in E_T} y^{ij}_{u,tns(j)} \geq \chi^{ij}, \quad \forall i,j\in V_D, \tag{4}$$

$$\sum_{u:(w,u)\in E_T} y^{ij}_{wu} - \sum_{v:(v,w)\in E_T} y^{ij}_{vw} \geq \chi^{ij} - 1, \quad \forall i,j\in V_D,\ \forall w\in V_T\backslash\{tns(i),tns(j)\}, \tag{5}$$

$$\sum_{u:(w,u)\in E_T} y^{ij}_{wu} - \sum_{v:(v,w)\in E_T} y^{ij}_{vw} \leq 1 - \chi^{ij}, \quad \forall i,j\in V_D,\ \forall w\in V_T\backslash\{tns(i),tns(j)\}, \tag{6}$$

$$y^{ij}_{uv} \leq \chi^{ij}, \quad \forall i,j\in V_D, \forall u,v\in V_T, \tag{7}$$

$$\sum_{k,l\in V_D} x^{kle}_{ij} \cdot \underline{M}_{kl} + z_Q(\sum_{k,l\in V_D} x^{kle}_{ij} \cdot \overline{M}_{kl}) \leq \sum_{\hat{b}_t\in\hat{B}} w^{ij}_t \cdot \hat{b}_t, \quad \forall e\in E_T, \forall (i,j)\in E_D, \tag{8}$$

$$\sum_{q:(i,q)\in E_D} x^{ije}_{iq} \geq 1, \quad \forall e\in E_T,\ \forall i,j\in V_D,\ \underline{M}_{ij}\geq 0, \tag{9}$$

$$\sum_{q:(q,j)\in E_D} x^{ije}_{qj} \geq 1, \quad \forall e\in E_T,\ \forall i,j\in V_D,\ \underline{M}_{ij}\geq 0, \tag{10}$$

$$\sum_{q:(p,q)\in E_D} x^{ije}_{pq} - \sum_{q:(q,p)\in E_D} x^{ije}_{qp} = 0, \quad \forall e\in E_T,\ \forall i,j\in V_D,\ \forall p\in(V_D\backslash\{i,j\}),\ \underline{M}_{ij}\geq 0, \tag{11}$$

$$x^{kle}_{ij} \leq 1 - y^{ij}_{uv}, \quad \text{with } e=(u,v),\ \forall e\in E_T,\ \forall k,l\in V_D,\ \forall (i,j)\in E_D, \tag{12}$$

$$\sum_{k,l\in V_D} x^{kl}_{ij} \cdot \underline{M}_{kl} + z_Q(\sum_{k,l\in V_D} x^{kl}_{ij} \cdot \overline{M}_{kl}) \leq \sum_{\hat{b}_t\in\hat{B}} w^{ij}_t \cdot \hat{b}_t, \quad \forall (i,j)\in E_D, \tag{13}$$

$$\sum_{q:(i,q)\in E_D} x^{ij}_{iq} \geq 1, \quad \forall i,j\in V_D,\ \underline{M}_{ij}\geq 0, \tag{14}$$

$$\sum_{q:(q,j)\in E_D} x^{ij}_{qj} \geq 1, \quad \forall i,j\in V_D,\ \underline{M}_{ij}\geq 0, \tag{15}$$

$$\sum_{q:(p,q)\in E_D} x^{ij}_{pq} - \sum_{q:(q,p)\in E_D} x^{ij}_{qp} = 0, \quad \forall i,j\in V_D,\ \forall p\in(V_D\backslash\{i,j\}),\ \underline{M}_{ij}\geq 0, \tag{16}$$

$$w^{ij}_t,\ x^{kle}_{ij},\ x^{kl}_{ij},\ y^{ij}_{uv},\ \chi^{ij} \in \{0,1\}. \tag{17}$$

The meaning of each sentence is

(1) The goal is to minimize the total cost of the network. For every pair of data nodes, if there is a link between them, then it has a non zero cost per length associated to it. In the calculation of that cost, only the lengths of the transport links that are used to send data flow are considered. This is represented by the term $\sum_{(u,v)\in E_T} y^{ij}_{uv} \cdot r(u,v)$, where (i,j) is the link considered.

The other term, $T(\sum_{\hat{b}_t \in \hat{B}} w_t^{ij} \cdot \hat{b}_t)$ is the cost of the technology selected for the link (i, j). The global cost is obtained adding all these individual costs (non-linear binary expressions).

- Constraint (2) states that if a technology was chosen for the link $(i, j) \in E_D$, the value of χ^{ij} is set to 1; otherwise, it is set to 0.
- Constraints (3), (4), (5), and (6) construct the paths in the transport network associated to each data link. Expressions (3) and (4) imply that if a non-zero capacity was chosen to dimension data link $(i, j) \in E_D$, then at least one transport link must start from $tns(i)$ and at least one transport link must arrive at $tns(j)$. They are the terminals of the path. Expressions (5) and (6) refer to intermediate points.
- Constraint (7) is an integrity constraint. If no capacity was assigned to data link $(i, j) \in E_D$ then no transport path should be found.
- Constraint (8) states that the capacity of a data link must be able to support the whole committed traffic carried by it, plus a certain percentage of Internet traffic z_Q, in every fault scenario. The function z_Q is completely established in [10], and it is piecewise linear and it was embedded in the data, so this constraint is linear.
- Constraints (9), (10), and (11) correspond to the routing in the data network in every fault scenario $e \in E_T$. They mean that $\forall i, j \in V_D$ such that $\underline{M}_{ij} > 0$, there must be at least one path between i and j that connects them.
- Constraint (12) establishes the relation between data links and transport links. It means that if a transport link $e \in E_T$ fails, then all the data links whose transport flow is routed through e cannot be used in fault scenario e.
- Constraint (13) states that the capacity of a data link must be able to support the whole committed traffic carried by it, plus a certain percentage of the excess or peak traffic, in the nominal scenario. As in Constraint (8), this one is also linear.
- Constraints (14), (15) and (16) correspond to the routing in the data network in the nominal scenario. There must be at least one path between i and j for every $i, j \in V_D$ such that $\underline{M}_{ij} > 0$.
- Constraint (17) establishes the domain of the variables.

We remark that the previous model accepts an equivalent linearized version, here termed as *Complete Version*. In [10], an exact Brand & Cut resolution is introduced to the Complete version. We invite the reader to consult [10] for further details. Here, the integer programming formulation is studied in two steps, where the routing is first selected, and the network dimensioning is performed afterwards (cf. section 6).

5. COMPUTATIONAL COMPLEXITY

In this section we will establish the computational complexity of the RODP. We will proceed by reduction to the Steiner Two-Edge-Survivable Network Problem (STESNP)

Instance 5.1 (STESNP). *Undirected graph $G = (V, E)$, non-Steiner nodes $T \subseteq V$ and non-negative costs c_e for each link $e \in E$.*

Problem 5.2. *Find the minimum-cost two-edge connected subgraph $H \subseteq G$ that covers T.*

Proposition 5.3. *STESNP belongs to the class of NP-Hard Computational Problems*

Proof. See [11]. □

Lemma 5.4. *In a feasible solution for RODP, both graphs $G_T = (V_T, E_T)$ and $G_D = (V_D, E_D)$ are two-edge connected*

Proof. If there is a bridge $e \in E_T$, consider the components $H_1 \cup H_2 = G_T - e$. Choose data nodes $i, j \in V_D$ with $M_{ij} > 0$ such that $tns(i) \in H_1$ and $tns(j) \in H_2$ (if there are no such nodes, the problem can be subdivided in graphs H_1 and H_2). Let P_{ij} the tunnel from i to j in G_D. There exists $u = (x, y) \in P_{ij}$ such that $(tns(x), tns(y)) = e$. Therefore, the traffic M_{ij} cannot be carried over G_T, and G_T must be two-edge connected.

Now, we will show that G_D must be two-connected as well. Recall that $V_D = V_T$. If there is a bridge $e_d = (u, v) \in E_D$ then $W_1 \cup W_2 = G_D - e_d$. Pick data nodes $i \in W_1, j \in W_2$ such that $M_{ij} > 0$, and let P_{ij} the tunnel from i to j in G_D under the nominal scenario, and $P_{tns(u),tns(v)}$ the routing of $e_d \in G_T$. Choose any $e_T \in P_{tns(u),tns(v)}$. In the failure scenario e_T, the graph G_D is separated into the components W_1 and W_2. Thus, under the failure scenario e_T the tunnel P_{ij} is not feasible, and G_D must be two-edge connected as well. □

Theorem 5.5. *RODP belongs to the class of NP-Hard computational problems.*

Proof. We show that RODP is at least as hard as STESNP. Consider an arbitrary instance (G, T, c_e) of STESNP. We build an instance for RODP such that both problems are equivalent. Choose $G_T = G_D = G$, $M_{ij} = 1$ iff $i, j \in T$ and $D_{i,j} = 0$ otherwise, $\hat{B} = \{b_0, b_1\}$ with $b_0 = 0$ and $b_1 > 2\sum_{i,j \in T} m_{ij}$, $T(b_{ij}) = 1$ iff $i, j \in T$, and $T(b_{i,j}) = 0$ otherwise, and distance function $r_e = c_e$. If we are able to solve this instance for RODP then we meet a solution for STESNP as well, and RODP belongs to the class of *NP*-Hard computational problems. □

The reader can find alternative proofs in the thesis of Parodi [10] and Risso [12].

6. DECOMPOSITION APPROACH

A two-step approach is used, since an exact resolution is prohibitive for large instances. Observe that for each data link, its corresponding path in the transport network is unique. We exploit this fact and solve these paths for every data link in the first step. Once these paths are chosen, the dimensioning of the data links and routing in the data network is solved in a second step. The decomposition gives a relaxation of the original model.

Even though the cost is never lower than the one offered by the original model, the objective function is linear in both steps.

In this section we assume that the traffic demands are represented as $\{(\underline{M}_d, \overline{M}_d) | d \in E_D\}$, where the set of nodes is the same as in the data network and the links represent the nodes among which there is traffic. For this model, we chose a link representation, sub-indices have been redefined in terms of links instead of node pairs; therefore the model variables are

- $x^{de} = 1$ if link $e \in E_D$ is used to send traffic between the nodes of $d \in E_D, \underline{M}_d > 0$ in the nominal scenario and 0 otherwise,
- $x_f^{de} = 1$ if link $e \in E_D$ is used to send traffic between the nodes of $d \in E_D, \underline{M}_d > 0$, in the fault scenario $f \in E_T$ and 0 otherwise,
- $y_f^e = 1$ if link $f \in E_T$ is used to send data for $e \in E_D$ and 0 otherwise,
- $w_t^e = 1$ if link $e \in E_D$ uses capacity $\hat{b}_t \in \hat{B}$ and 0 otherwise, and
- $\chi^e = 1$ if link $e \in E_D$ has a technology assigned and 0 otherwise.

Additionaly, consider the auxiliary variables

- $\gamma^{ep} = 1$ if $p \in V_D$ is used in the routing of $e \in E_D$ in the nominal scenario and 0 otherwise,
- $\gamma_f^{ep} = 1$ if $p \in V_D$ is used in the routing of $e \in E_D$ in fault scenario $f \in E_T$ and 0 otherwise, and
- $\delta_q^e = 1$ if $q \in V_T$ is used in the routing of $e \in E_D$ and 0 otherwise.

The relationships among links and nodes are abbreviated: (i) $e \sim v$, means the link e is incident to the node v, and (ii) e_1 depicts the source vertex, and e_2 the sink vertex of link e, respectively.

6.1. Step 1

In the first step the goal is to find the best transport path for each data link. In order to do that, the length of those paths is minimized, which results in a minimization of the total cost of the final solution. The routing is modelled as constraints that control the cardinality of links incident to nodes.

$$\min \sum_{e\in E_D} \sum_{f\in E_T} r_T(f) \cdot y_f^e \tag{18}$$

s.t. :

$$\sum_{f\in E_T: f\sim tns(e_1)} y_f^e = 1, \forall e\in E_D, \tag{19}$$

$$\sum_{f\in E_T: f\sim tns(e_2)} y_f^e = 1, \forall e\in E_D, \tag{20}$$

$$\sum_{f\in E_T: f\sim q} y_f^e = 2\delta_q^e, \quad \forall e\in E_D\ \forall q\in V_T\setminus\{tns(e_1), tns(e_2)\}, \tag{21}$$

$$y_f^e,\ \delta_q^e \in \{0,1\}. \tag{22}$$

6.2. Step 2

The solution provided in Step 1 serves as an input for Step 2. In this step the variable y_f^e is fixed with the value provided as the result of Step 1, $\overline{y}_f^e$. This is the reason why the objective function is linear; the only variables that appear are w_t^e. Once we have the capacity of each data link and the corresponding paths on the data network, a complete solution is being built. These paths should be found for every fault scenario in addition to the paths in the nominal scenario.

$$\min \sum_{e\in E_D} \sum_{f\in E_T} \sum_{\hat{b}_t\in\hat{B}} r_T(f) \cdot T(\hat{b}_t) \cdot \overline{y}_f^e \cdot w_t^e \tag{23}$$

s.t. :

$$\chi^e = \sum_{\hat{b}_t\in\hat{B}} w_t^e \quad \forall e\in E_D, \tag{24}$$

$$\sum_{d\in E_D, \underline{M}_d>0} x_f^{de} \cdot \underline{M}_d + z_Q\Big(\sum_{d\in E_D, \overline{M}_d>0} x_f^{de} \cdot \overline{M}_d\Big) \leq \sum_{\hat{b}_t\in\hat{B}} w_t^e \cdot \hat{b}_t, \quad \forall f\in E_T,\ \forall e\in E_D, \tag{25}$$

$$\sum_{e\in E_D: e\sim d_1} x_f^{de} = 1, \quad \forall f\in E_T,\ \forall d\in E_D, \underline{M}_d>0, \tag{26}$$

$$\sum_{e\in E_D: e\sim d_2} x_f^{de} = 1, \quad \forall f\in E_T,\ \forall d\in E_D, \underline{M}_d>0, \tag{27}$$

$$\sum_{e\in E_D:e\sim p} x_f^{de} = 2\gamma_f^{ep}, \quad \forall f\in E_T,\ \forall d\in E_D, \underline{M}_d>0, \forall p\in(V_D\setminus\{d_1,d_2\}). \tag{28}$$

$$x_f^{de} \leq 1 - \overline{y}_f^{e}, \quad \forall f\in E_T, \forall e\in E_D, \forall d\in E_D, \underline{M}_d>0, \tag{29}$$

$$\sum_{d\in E_D, \underline{M}_d>0} x^{de}\cdot \underline{M}_d + z_Q\Big(\sum_{d\in E_D, \overline{M}_d>0} x^{de}\cdot \overline{M}_d\Big) \leq \sum_{\hat{b}_t\in\hat{B}} w_t^e\cdot \hat{b}_t, \quad \forall e\in E_D, \tag{30}$$

$$\sum_{e\in E_D:e\sim d_1} x^{de} = 1, \quad \forall d\in E_D, \underline{M}_d>0, \tag{31}$$

$$\sum_{e\in E_D:e\sim d_2} x^{de} = 1, \quad \forall d\in E_D, \underline{M}_d>0, \tag{32}$$

$$\sum_{e\in E_D:e\sim p} x^{de} = 2\gamma^{ep}, \quad \forall d\in E_D, \underline{M}_d>0, \forall p\in(V_D\setminus\{d_1,d_2\}), \tag{33}$$

$$x_f^{de}, x^{de}, w_t^e, \chi^e, \gamma^{ep}, \gamma_f^{ep} \in \{0,1\}. \tag{34}$$

The meaning of these constraints is precisely the same as in the basic model.

6.3. *Building a feasible solution*

Step 2 is executed independently for every fault scenario, as well as for the nominal scenario. A solution is built using all scenarios. The paths in the data network for each fault scenario, as well as the nominal one, are precisely the ones returned in each execution of the model. The capacities b_e should support the flow in every scenario. Therefore, the biggest capacity among all scenarios is selected: $b_e = \max_{f\in E_T\cup f_0}\{b_e^f\}$.
Consequently, a feasible solution is produced. This is a remarkable property of this decomposition approach, which we refer to as *Decomposed version*.

7. NUMERICAL RESULTS

Real-life instances were provided by the Uruguayan telecommunications service provider, ANTEL. The transport network is presented in Figure 2. We study the network into two regions: East and West (see Figure 3).

We additionally consider artificial test cases where the data network was modified and consists of a full-mesh topology or "half-full-mesh" topology. The test cases that include this modification are identified by `fm` or `hfm` respectively. The traffic demands were uniformly distributed at random from 0 to 30 Gbps. The number of nodes with traffic demand is half the number of data links. The adjective `full` will be added to a test case when practically all links operate close to their full capacity. Their capacities range from 1 to 10 Gbps with a step of 1 Gbps. Test cases are shown in 1 and 2. In the Decomposition approach, the number of variables and constraints are presented for each failure scenario.

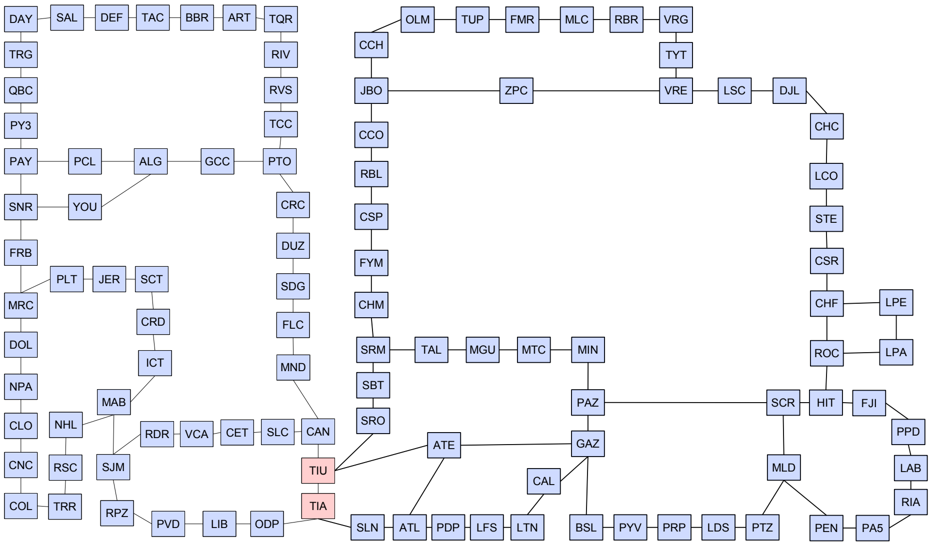
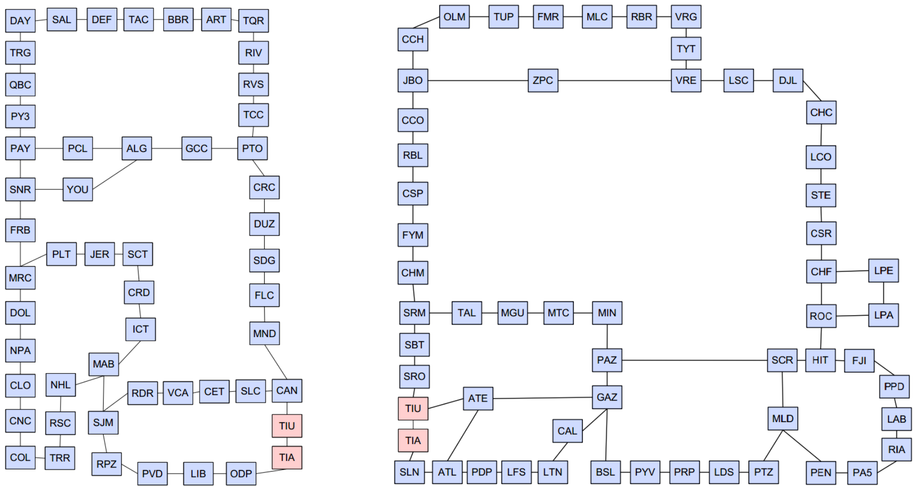


Figure 2: ANTEL transport network.



(a) West Region. (b) East Region.

Figure 3: Decomposition of ANTEL transport network in (a) West and (b) East regions.

All the basic structures and algorithms were implemented using *C++*, together with the branch and cut solver of the *ILOG CPLEX* [13] library. The code was run on a Intel Core2 Quad 2.83GHz 12MB Cache with 4GB RAM computer. The results are shown in Table 3. For each test case for the Complete version we can find the total cost, the relative MIP gap (between the best integer objective and the objective of the best node remaining), and the elapsed time. For each test case at the Decomposed version we find total cost and the elapsed time. Additionally, the objective values of two metaheuristics applied to the same problem by other authors are shown: Variable Neighborhood Search (column VNS [14]) and Tabu Search (column TS [15]). Finally, Table 5 shows the relative gap between the costs of the Decomposed and Exact (optimal) solution, while Table 6 presents the relative gap between the costs of the Decomposed solution versus TS and VND heuristic solutions.

Table 1: Relevant data network information for different test cases.

Test Case	# data links	# data nodes	# demands	# capacities
`east_copy`	15	18	11	2
`east_copy_cap`	15	18	11	10
`east_copy_full_cap`	15	18	7	10
`east_fm`	153	18	11	2
`east_fm_cap`	153	18	11	10
`east_fm_full`	153	18	75	2
`east_fm_full_cap`	153	18	75	10
`east_hfm`	79	18	11	2
`west_copy`	47	18	26	2
`west_copy_cap`	47	18	26	10
`west_copy_full_cap`	47	18	26	10
`west_fm`	47	18	16	2

Table 2: Size of test cases at complete and decomposed versions.

Test case	Complete version		Decomposed version	
	# variables	# constraints	# variables	# constraints
`east_copy`	11895	29145	145	866
`east_copy_cap`	11850	29145	265	866
`east_copy_full_cap`	8055	20085	255	562
`east_fm`	46053	60733	1989	7214
`east_fm_cap`	–	–	3213	7214
`east_fm_full`	–	–	11781	47406
`east_fm_full_cap`	–	–	13005	47406
`east_hfm`	23657	34019	+	+
`west_copy`	29281	43168	1316	5398
`west_copy_cap`	29657	43544	1692	5398
`west_copy_full_cap`	29657	43544	1692	5398
`west_fm`	60129	75292	2754	10354

$^{-}$There is not sufficient memory space to model it.
$^{+}$The system fails to obtain a feasible solution.

Table 3: Performance for Complete and Decomposed versions, VNS and TS metaheuristics.

Test case	Complete version	Gap	Decomp. Version	VNS	TS
`east_copy`	89175	0	89175	112206	112207
`east_copy_cap`	15114	0	15114	107891	107891
`east_copy_full_cap`	28702	0	28702	111247	95185
`east_fm`	112418	32	171811	128384	114536
`east_fm_cap`	-	-	26304	106854	112207
`east_fm_full`	-	-	512385	1715966	1662190
`east_fm_full_cap`	-	-	463313	1375590	1598260
`east_hfm`	413000	69.6	+	+	+
`west_copy`	882468	58	605957	977649	977650
`west_copy_cap`	513074	95.5	188020	816818	940048
`west_copy_full_cap`	NF	NF	277386	774090	940048
`west_fm`	413509	42.96	335044	354282	460764

Table 4: CPU Time (in seconds) for Complete, Decomposed versions, VNS and TS metaheuristics.

Test case	Complete version	Decomp. Version	VNS	TS
`east_copy`	2	1	5	90
`east_copy_cap`	28	1	7	77
`east_copy_full_cap`	43	1	240	140
`east_fm`	1051980	853	110	51
`east_fm_cap`	-	13	136	49
`east_fm_full`	-	99113	174	152
`east_fm_full_cap`	-	899	536	468
`east_hfm`	90240	+	-	-
`west_copy`	11371	70471	45	78
`west_copy_cap`	48407	1699	33	112
`west_copy_full_cap`	806580	16528	34	40
`west_fm`	501060	1081	823	67

Some remarks are mandatory with respect to the results. Let us start with the Complete version.

- Outperforms both metaheuristics in all test cases with prefix `east_copy`.
- Even though the running time for `east_fm` was longer (more than 12 days), we find better results than both metaheuristics. Nevertheless, the optimization process stops when the MIP gap is 32%. The reader can appreciate from Table 4 that our approach requires higher CPU times than the previous metaheuristics, unless an exact solution is found.
- In test cases with prefix `east_fm` we were not able to find an integer solution in short time due to their complexity.
- It produces a solution for test case `east_hfm`, even though both metaheuristics could not return feasible solutions. This test case was stopped when finding the first integer solution that took already a lot of time (more than one day).

- Test cases `west_copy` and `west_copy_cap` improved the results obtained from the VNS and TS and were executed up to gaps of 58% and 95.5% respectively.
- Test case `west_copy_full_cap` run for approximately 9 days and 8 hours and could not find any integer solution.
- Test case `west_fm` was running for almost 6 days and it outperformed TS. It is worth to remark that it run up to a gap of 42.96%. A better solution than VNS should also be reached.

Let us make now similar remarks for the Decomposed version. Unless stated otherwise, the optimal solution was found for each failure scenario. In those cases were this could not be done, the MIP gap is clarified, and it is the same for every failure scenario in that test case.

- Test cases with prefix `east_copy` gave the same solution as the exact version and outperformed both metaheuristics.
- Test case `east_fm` was able to find a solution not as good as the metaheuristics, but provides a faster execution time with respect to the exact method.
- Test case `east_fm_cap` run really fast and found much better solution than VNS and TS.
- Test cases `east_fm_full` and `east_fm_full_cap` improved significantly the results obtained from the VNS and TS and they were executed up to gaps of 20% and 10% respectively in each failure scenario.
- Finally, test case `west_copy_cap` improved not only the results obtained from the metaheuristics but also the one obtained with the complete version (that was not optimal); it is worth noticing the speed with which it found that solution.

It should be clarified that the test cases that are missing in the table or the ones that do not have any results could not be executed because of lack of time. Considering the amount of data nodes and links and different traffic loads, the remaining cases should be comparable in time to the most complex ones, needing roughly a month to solve each. Nevertheless, a cost reduction of 1% usually means millionaire savings.

8. CONCLUDING REMARKS

Nowadays, DWDM is the dominant network technology in high-capacity optical backbone networks. The main drawback of DWDM technology is the existence of multiple layers (called overlays in the literature).

In this paper we formally present a mathematical programming problem where the goal is to design a resilient overlay at minimum cost. There, a major cause of concern is the re-assignment of multicommodity flows for every single point of failure in the transport network since it provokes multiple failures in the data network.

Table 5: Complete version VS Decomposition.

Test case	%D-E
`east_copy`	0
`east_copy_cap`	0
`east_copy_full_cap`	$-2.4E-3$
`east_fm`	34.6
`west_copy`	−45.6
`west_copy_cap`	−172.9
`west_fm`	−23.4

Table 6: Decomposed approach VS Metaheuristics.

Test case	%D-VNS	%D-TS
`east_copy`	20.5	20.5
`east_copy_cap`	86.0	86.0
`east_copy_full_cap`	74.2	69.8
`east_fm`	−33.8	−50.0
`east_fm_cap`	75.3	76.6
`east_fm_full`	70.1	69.2
`east_fm_full_cap`	66.3	71.0
`west_copy`	38.0	38.0
`west_copy_cap`	77.0	80.0
`west_copy_full_cap`	64.2	70.5
`west_fm`	5.4	27.3

The problem has been heuristically addressed in prior works in the literature since it belongs to the class of NP-Hard computational problems. Here, a Decomposition approach is provided, where the problem is solved in two steps that involve linear objectives. Numerical results using real-life instance suggest that the proposal is competitive with respect to previous heuristics (Tabu Search and Variable Neighborhood Search). A drawback is that this decomposition approach is time consuming (some runs take several days).

It is worth to mention that an improvement of 1% in real-life instances mean millionaire savings. As future work, we would like to consider different relaxations to the model in order to obtain bounds to the cost of the optimal solution. A different approach is to consider a link-path representation, in contrast to node-link setting. Following a decomposition approach, a path may be generated when it is needed. Furthermore, the implementation could be adjusted taking advantage of the problem instances.

REFERENCES

[1] Colbourn, C.J., *Reliability Issues In Telecommunications Network Planning*, Springer, US, 1999.

[2] Orlowski, S., Koster, A., Raack, C., and Wessly, R., ”Two-layer network design by branch-and-cut featuring MIP-based heuristics”, in: *Proceedings of the Third International Network Optimization Conference (INOC 2007)* Spa, Belgium, 2007.

[3] Ruiz, M., Pedrola, O., Velasco, L., Careglio, D.,Fernández-Palacios, J., and Junyent, G., ”Survivable IP/MPLS-Over-WSON multilayer network optimization”, *Optical Communications and Networking* 3 (8) 2011, 629–640.

[4] Zhang, X., Li, K., and Bian, L., ”Towards the maximum resource sharing degree for survivable IP/MPLS over WDM mesh networks”, *Optical Switching and Networking*, 11 (2014) 177–188.

[5] Fouilhoux, P., Karasan, O.E., Mahjoub, A.R., Zkk, O., and Yaman, H., ”Survivability in hierarchical telecommunications networks”, *Networks*, 59 (1) (2012) 37–58.

[6] Okamura, H., and Seymour, P., ”Multicommodity flows in planar graphs”, *Combinatorial Theory, Series B*, 31 (1) (1981) 75–81.

[7] Alevras, D., Grtschel, M., and Wessly, R., ”A network dimensioning tool”, in: *Preprint SC 96–49, Konrad-Zuse-Zentrum fur Informationstechnik*, 1996.

[8] Raack, C., Koster, A., Orlowski, S., and Wessly, R., ”On cuinequalities for capacitated netdesign polyhedra”, *Networks*, 2010.

[9] Oellrich, M., ”Minimum cost disjoint paths under arc dependence”, *University of Technology Berlin*, (4) 2008.

[10] Parodi, C., ”Integer optimization applied to the design of robust minimum cost multi-layer networks”, Master’s thesis, Facultad de Ingeniería, Universidad de la República Uruguay, 2011.

[11] Baiou, M., ”Le problème du sous graphe Steiner 2-arête connexe: Approche polyédrale”, Ph.D. dissertation, Université de Rennes I, France, 1996.

[12] Risso, C., ”Using GRASP and GA to design resilient and cost-effective IP/MPLS networks”, Theses, University of the Republic, Uruguay, 2014.

[13] IBM, *IBM ILOG CPLEX V12.1: User’s Manual for CPLEX*, International Business Machines Corporation, Armonk, New York, US, 2009.

[14] Corez, A., ”Multi-Overlay network planning by applying a variable neighbourhood search approach”, Master’s thesis, Facultad de Ingeniería, Universidad de la República Uruguay, 2010.

[15] Despaux, F., ”Optimización de una Red de Datos IP/MPLS sobre SDH/DWDM usando Tabú Search”, 2011.