Traffic Engineering over Hop-Constrained Node Survivable Networks

Luís Gouveia, *CIO and DEIO, Faculdade de Ciências da Universidade de Lisboa (legouveia@fc.ul.pt)*
Pedro F. Patrício, *Departamento de Matemática, Universidade da Beira Interior (pedrofp@mat.ubi.pt)*
Amaro de Sousa, *Instituto de Telecomunicações, Universidade de Aveiro (asou@av.it.pt)*

February 2007

Keywords: Traffic Engineering, Hop-Constraints, Survivability

1. Introduction

Traffic engineering on a given network is the task of determining how traffic commodities must be routed in order to maintain an optimal performance. In this work, we aim to optimize routing by minimizing the number of routing hops, which has a positive impact on the delay provided by the network to traffic commodities, while ensuring some desired survivability guarantees.

In [1], we have considered the network design of MPLS over WDM networks. In that work, we have considered a given undirected network $G = (V, E)$ where V is the set of WDM nodes and E is the set of physical connections (fibers) between WDM nodes, and also a set $S \subset V$ of traffic MPLS nodes (named Label Switched Routers or LSRs) that are origins/destinations of commodities each of which with demand $t_{pq}, p, q \in S(p < q)$. The design task was the determination of the number and routes of the WDM lightpaths and the location of core LSRs. Each WDM lightpath had an associated cost which was proportional to the length of its minimum length path on G. Each core LSR potential location had an associated given cost. The aim of the network design task was to minimize the total network cost, while guaranteeing the existence of D node disjoint hop-constrained paths for every commodity, accommodating a given demand matrix $T = [t_{pq}], p, q \in S(p < q)$. The hop-constrained paths account for a maximum number of hops that the design solution must guarantee. The D node disjoint paths, together with the capacity assigned to each path, account for the degree of survivability that the design solution must guarantee.

There are two reasons to apply traffic engineering on pre-dimensioned networks. One is that the commodity paths given by a minimum cost network design solution might not be the optimal ones (the design model defines a maximum number of hops for each path but in the design solution these values might be improved). Second, the commodity demand values used on the design task are usually estimations that might be different from the traffic that is to be supported by the network when it is put in operation.

Therefore, let a network design solution be represented by a network $N = (X, U)$, where the node set X is the set of all LSRs (the traffic and the core LSRs) and edge set U is the set of pairs of LSRs with lightpaths connecting them (b_e represents the total capacity of the lightpaths on edge $e \in U$). This solution supports all commodities of the given estimated demand matrix T for certain values of H (the maximum number of hops between any $p, q \in S, p < q$), D and β (a percentage coefficient associated with the survivability scheme that will be further explained below). Now, consider a new demand matrix $R = [r_{pq}]$ which is different from the estimated T. In this work, we generate each r_{pq} value randomly with a uniform distribution between $(1 - \delta)t_{pq}$ and $(1 + \delta)t_{pq}$, considering $\delta = 0.05, 0.1, 0.15$ or 0.2 to accommodate different error degrees in the estimation of the initial traffic matrix.

We consider a traffic engineering problem of routing the new demands (r_{pq}) over the dimensioned network $N = (X, U)$ complying with the installed bandwidth (b_e) on each edge of U and guaranteeing D node disjoint hop-constrained paths for every commodity. An optimal routing is the one that minimizes i) the average number of hops or ii) the largest number of hops, between all pairs of nodes $p, q \in S(p < q)$.

Either considering the objective i) or ii), we assume there is a maximum allowable number of hops for each commodity, denoted by $H^* = H$. This value has been given as an input of the previous problem [1]. For all $p, q \in S, p < q$, we can generically model D hop constrained paths as follows:

$$\left\{(i,j) : f_{ij}^{pq} > 0 \text{ contains } D\ (p,q) - H^* - paths\right\}, p,q \in S$$

where f_{ij}^{pq} represents the number of paths from p to q traversing edge $\{i,j\}$ in the direction from i to j, and a $(p,q) - H^* - path$ is a sequence of at most H^* arcs $\{(i_1,j_1),...,(i_k,j_k)\}$ such that $i_1 = p, j_k = q$ and $j_s = i_{s+1}$ for $s = 1,,k-1$. Following [2], we can describe these generic constraints using a single set of variables for each commodity, w_{ij}^{hpq}, representing the number of $(p,q) - H^* - paths$ traversing edge $\{i,j\}$ in the direction from i to j, in the h^{th} position. Hence, for all $p,q \in S, p < q$, the model representing D hop-constrained paths from p to q can now be defined as:

$$\begin{aligned}
&\sum_{j\in X} w_{pj}^{1pq} = D \qquad \text{for all } p,q \in S \\
&\sum_{j\in X} w_{ij}^{2pq} = w_{pi}^{1pq} \qquad \text{for all } i \neq p; p,q \in S \\
&\sum_{j\in X} w_{ij}^{h+1,pq} - \sum_{m\in X} w_{mi}^{hpq} = 0 \qquad \text{for all } i \neq p; h = 2,...,H^*-1; p,q \in S \\
&\sum_{j\in X} w_{jq}^{H^*pq} = D \qquad \text{for all } p,q \in S \\
&w_{ij}^{hpq}, w_{ji}^{hpq} \in \{0,1,...,D\} \qquad \text{for all } e = \{i,j\} \in U; h = 1,...,H^*; p,q \in S \\
&w_{qq}^{hpq} \in \{0,1,...,D\} \qquad \text{for all } h = 2,...,H^*; p,q \in S
\end{aligned} \tag{1}$$

This model contains "loop" variables $w_{qq}^{hpq}(h = 2,...,H^*)$ to represent situations when some of the D $(p,q)-H^*-paths$ contain fewer than H^* arcs (that is, $w_{jq}^{hpq} = 1$ for some $j \in V\setminus\{q\}$ and $1 \le h \le H^*-1$).

We divide our study according to two survivability schemes (see [1]): Path Diversity and Path Protection. The first scheme is appropriate when the network operator aims to enhance demand protection but total protection is not a requirement; the latter scheme is preferred when total demand protection is needed.

2. Survivability Schemes and Optimization Criterias

In the Path Diversity scheme, the demand r_{pq} of each commodity $p,q \in S(p < q)$ is equally split by the D node disjoint paths. This scheme ensures that a percentage of the total demand, $(D-1)/D \times 100\%$, is guaranteed if a single network element fails (the most likely failure case). When $D = 2$, each of both paths between p and q support half of r_{pq} ($\beta = 1/2$) and, thus, 50% of the total demand is always protected. When $D = 3$, each path supports one third of r_{pq} ($\beta = 1/3$) and, so, 66% of the total demand is protected.
Under this survivability scheme, all the D node disjoint paths carry traffic between any pair of nodes $p,q \in S(p < q)$ which implies that all the $D \times |S| \times (|S|-1)/2$ paths are subject to the optimization criteria, either considering i) or ii).
Minimizing the average number of hops is equivalent to maximizing the sum of the loop variables: a path from p to q with fewer arcs makes more loops at node q:

$$Minimize \sum_{p,q \in S} \sum_{\{i,j\} \in U} \sum_{h=1,\ldots,H^*} w_{ij}^{hpq} \iff Maximize \sum_{p,q \in S} \sum_{h=2,\ldots,H^*} w_{qq}^{hpq}$$

subject to:

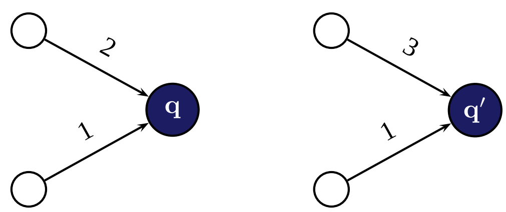
(1)

$$\sum_{i \in X} \sum_{h=1,\ldots,H^*} w_{ij}^{hpq} \leq 1 \qquad \text{for all } j \in X \backslash \{p,q\}; p,q \in S \qquad \textbf{(2)}$$

$$\sum_{p,q \in S} \beta r_{pq} \left(\sum_{h=1,\ldots,H^*} w_{ij}^{hpq} + \sum_{h=1,\ldots,H^*} w_{ji}^{hpq} \right) \leq b_e \qquad \text{for all } e = \{i,j\} \in U \qquad \textbf{(3)}$$

This model contains the previously described flow conservation constraints **(1)**. The following group of constraints **(2)** guarantee that the D paths between any pair of nodes $p, q \in S(p < q)$ are node disjoint, while constraints **(3)** prescribe that the traffic matrix routing must comply with the installed capacity on each edge of U.

To define an appropriate objective function for the largest number of hops minimization criteria, for each commodity we need a way to identify how many hops are contained in the worst case path among the D paths supporting that commodity. We do it using the loop variables. To better understand the proposed formulation for this criteria, consider example 1 below: two commodities (with origins p and p', and respective destinations q and q'), $D = 2$, $H^* = 4$ and where the values associated with the arcs represent the position in which they are being traversed:



Example 1

For the first commodity we have $w_{qq}^{2pq} = 1$ and $w_{qq}^{3pq} = w_{qq}^{4pq} = 2$, and, for the second, we have $w_{q'q'}^{2p'q'} = w_{q'q'}^{3p'q'} = 1$ and $w_{q'q'}^{4p'q'} = 2$. Consider binary variables V^{hpq} indicating whether 2 loops are performed at node q in position h from p to q, for all $h = 2, 3, 4$. If $V^{hpq} = 1$, then we know that the "worst" path (the path with the largest number of hops) between p and q reaches node q after $h - 1$ hops at most.
In this example, we want to have $V^{3pq} = V^{4pq} = 1, V^{2pq} = 0$ for the first commodity and, for the second commodity, $V^{4p'q'} = 1$ and $V^{2p'q'} = V^{3p'q'} = 0$. Note that for each commodity the sum of the corresponding variables, $\sum_{h=2,\ldots,H^*} V^{hpq}$, can be interpreted as indicating "how early" the worst path between p and q reaches node q. Furthermore, if we bound below quantities $\sum_{h=2,\ldots,H^*} V^{hpq}$ by an integer variable V and we maximize V, then we are forcing the worst paths to reach q as "early" as possible, thus minimizing the largest number of hops of the worst path from p to q. The integer variable V can be interpreted as how early the worst of all paths arrives at its destination node (in the example, $V = 1$). Moreover, note that V^{hpq} can only be 1 if $w_{qq}^{hpq} = 2$ and, so, we must consider the binding constraints $2.V^{hpq} \leq w_{qq}^{hpq}$, for all $h = 2, 3, 4$.
Generalizing, for either D = 2 or D = 3, we define:

$$V^{hpq} = \begin{cases} 1, & \text{if } D \text{ loops are performed at node } q \text{ in position } h\text{, from } p \text{ to } q \\ 0, & \text{otherwise} \end{cases} \quad , p,q \in S; h = 2, ..., H^*$$

The problem that minimizes the largest number of hops can, now, be defined as:

$Maximize\, V$

subject to:

(1), (2), (3)

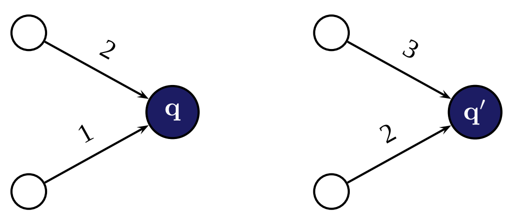
$D.V^{hpq} \leq w_{qq}^{hpq}$ for all $p, q \in S; h = 2, ..., H^*$

$$\sum_{h=2,\dots,H^*} V^{hpq} \geq V \quad \text{for all } p, q \in S$$

$V^{hpq} \in \{0, 1\}$ for all $p, q \in S; h = 2, ..., H^*$

$V \geq 0$ and integer

The Path Protection survivability scheme ensures that the D node disjoint paths between any pair of nodes $p, q \in S(p < q)$ support a capacity such that if one path fails, the remaining ones must accommodate the total demand.
For $D = 2$, each path is able to support the total demand ($\beta = 1$).In this case, the best path (the one with fewer number of hops) is used as the service path (the commodity demand is support by this path) and the other path is used as a protection (backup) path (it is used whenever the service path fails).
For $D = 3$, each path is able to support half the demand ($\beta = 1/2$). In this case, the two best paths (the ones with fewer number of hops) are used as service paths (the total demand is equally split by the two service paths) and the other path is used as a protection (backup) path (it is used whenever one of the service paths fails).
Note that the protection paths do not support traffic under normal network operation. They support traffic when a failure occurs and only during the period until the failure is fixed. Therefore, we consider the minimization of the number of hops only on the service paths.
Considering the average number of hops minimization criteria, since for $D = 2$ we have one service path per commodity, we want to minimize the average number of hops of the "best" path supporting each commodity, that is, we just want to consider the path with the least number of hops between the two paths supporting each commodity. For $D = 3$, we have two service paths and thus we consider the two best paths of each commodity for the average minimization purpose.
In order to motivate the next model consider, for $D = 2$, the following example with two commodities, $H^* = 4$ and where the values associated with the arcs represent the position in which they are being traversed:

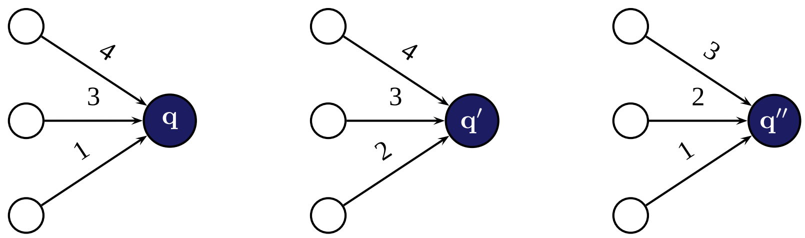


Example 2

In this example, for the first commodity we have $w_{qq}^{2pq} = 1$ and $w_{qq}^{3pq} = w_{qq}^{4pq} = 2$, and, for the second one, $w_{q'q'}^{2p'q'} = 0, w_{q'q'}^{3p'q'} = 1$ and $w_{q'q'}^{4p'q'} = 2$. For each commodity, we want to know "when" the best path reaches its destination node, regardless of how the worst path performs. That is, if the first loop that is made at node q is in position 2, then we must be able to use this information. Thus, consider binary variables V^{hpq} as follows:

$$V^{hpq} = \begin{cases} 1, & \text{if at least a loop is performed at node } q \text{ in position } h \text{, from } p \text{ to } q \\ 0, & \text{otherwise} \end{cases} \quad , p, q \in S; h = 2, ..., H^*$$

In this example, for the first commodity, we must have $V^{2pq} = V^{3pq} = V^{4pq} = 1$. We can impose these values upper bounding the V^{hpq} variables by the w_{qq}^{hpq} variables, for all $h = 2, 3, 4$, and maximizing $\sum_{h=2,...,H^*} V^{hpq}$. Note that this sum can be interpreted as how early the best path from p to q arrives at node q. For the second commodity, considering similar constraints and adding $\sum_{h=2,...,H^*} V^{hp'q'}$ to the objective function, we obtain $V^{3p'q'} = V^{4p'q'} = 1$.
For $D = 3$, we are interested in the two best paths supporting each commodity. Similarly to the case $D = 2$, we will use the V^{hpq} variables to detect "when" the two best paths reach their destination node. However, the meaning of those variables will slightly differ from that of case $D = 2$. Consider the next example with three commodities, $H^* = 4$ and where the values associated with the arcs represent, again, the position in which they are being traversed:



Example 3

We follow the same idea as in $D = 2$, that is, we upper bound the V^{hpq} variables by the w_{qq}^{hpq} variables, for all $h = 2, 3, 4$, and maximize $\sum_{h=2,...,H^*} V^{hpq}$, but we now let the V^{hpq} variables assume values in $\{0, 1, 2\}$. Hence, $\sum_{h=2,...,H^*} V^{hpq}$ represents the number of loops performed at node q (by the two best paths from p to q). We can interpret each variable as a filter: that is, if, from p to q, k loops are performed at node q in position h, then $V^{hpq} = min\{k, 2\}$ (note that, for $D = 2$, we can also interpret the V^{hpq} variables as filters and in that case $V^{hpq} = min\{k, 1\}$).
The problem that minimizes the average number of hops, for either $D = 2$ or $D = 3$, can now be stated as follows:

$$Maximize \sum_{p,q \in S} \sum_{h=2,...,H^*} V^{hpq}$$

subject to:

(1), (2), (3)

$V^{hpq} \leq w_{qq}^{hpq}$ for all $p, q \in S; h = 2, ..., H^*$

$V^{hpq} \in \{0, 1, ..., D - 1\}$ for all $p, q \in S; h = 2, ..., H^*$

Considering now the largest number of hops minimization criteria under the Path Protection scheme, the situation differs from that of the Path Diversity scheme in the sense that: when $D = 2$, we want to minimize the largest number of hops of the best path (service path) of every commodity; and, when $D = 3$, we want to minimize the largest number of hops of the worst path between the best two paths of every commodity.
Consider examples 2 and 3. The idea is to set a binary variable V^{hpq} to 1 only if: the best path from p to q performs a loop at node q in position h, for $D = 2$; the worst of the two best paths from p to q performs a loop at node q in position h, for $D = 3$. If our aim is to minimize the largest number of hops, the formulation to this problem becomes as follows:

$Maximize\, V$

subject to:

(1), (2), (3)

$(D-1).V^{hpq} \leq w_{qq}^{hpq}$ for all $p, q \in S; h = 2, ..., H^*$

$$\sum_{h=2,\ldots,H^*} V^{hpq} \geq V \quad \text{for all } p, q \in S$$

$V^{hpq} \in \{0, 1\}$ for all $p, q \in S; h = 2, ..., H^*$

$V \geq 0$ and integer

For $D = 2$, $\sum_{h=2,\ldots,H^*} V^{hpq}$ represents how early the best path from p to q arrives at node q, whereas for $D = 3$ it represents how early the worst of the two best paths from p to q arrives at node q. If we want to minimize the largest number of hops, we then have to upper bound an integer variable V by $\sum_{h=2,\ldots,H^*} V^{hpq}$, for all $p, q \in S$, and maximize V.

On an extended version of this paper, we will provide more insight on the proposed formulations, considering the survivability schemes and the two optimization criteria described on this version, and we will also present our computational results.

References

[1] Gouveia, L. E.; Patrício, P.F.; de Sousa, A.F., "Hop-Constrained Node Survivable Network Design: an Application to MPLS over WDM", *submitted to the 8th INFORMS Telecommunications Conference Proceedings.*

[2] Gouveia, L. E.; Patrício, P.F.; de Sousa, A.F., "Compact models for hop-constrained node survivable network design", *Telecommunications Network Planning: Innovations in Pricing, Network Design and Management, ed. G. Anandaligam and S. Raghavan, Springer, 2006.*