Designing Paths with Minimum Number of Hops: Comparing Disaggregated and Aggregated Formulations

Luís Gouveia[*] Pedro Patrício[⋆] Amaro de Sousa[⋄]

[*] *CIO and DEIO, Faculdade de Ciências da Universidade de Lisboa*
Bloco C6, Piso 4, 1749-016 Lisboa, Portugal;

[⋆] *CIO and Departamento de Matemática, Universidade da Beira Interior*
Rua Marquês d'Ávila e Bolama, 6201-001 Covilhã, Portugal

[⋄] *Instituto de Telecomunicações, Universidade de Aveiro*
Campus Universitário de Santiago, 3810-193 Aveiro, Portugal

Abstract

Given an undirected network with link capacities and a set of commodities with known demands, this paper addresses the problem of determining D (with $D = 2, 3$) hop-constrained node disjoint paths for each commodity while minimizing the average or the maximum number of hops of all relevant paths. These paths are defined according to two survivability mechanisms: Path Diversity and Path Protection. The addressed problems arise in the context of a traffic engineering task over pre-dimensioned networks where some changes occur on the estimated traffic demands.
We present two classes of ILP models, disaggregated and aggregated, for both problems, establish the relationship between their linear programming relaxations and compare their effectiveness through computational experiments. Contrary to what one would expect, in practice there is no gain in using the disaggregated models.

Keywords*: Hop Constraints, Survivability, Traffic Engineering*

1 Introduction

Given an undirected network $N = (X, U)$ with link capacities (b_e) and a set of commodities with known demands, this paper addresses the problem of routing demands (r_{pq}) for each commodity (p, q) over the network N, under the survivability mechanism (Path Diversity or Path Protection) one wishes to implement, through D hop-constrained node disjoint paths for every commodity and complying with the installed bandwidth on each edge $e \in U$. The two types of survivability mechanisms − Path Diversity and Path Protection − which differ in the way one wishes to protect total demand: the first is suitable to enhance demand protection but total protection is not a requirement (as each demand is equally split over D paths); the latter mechanism is preferred when total demand protection is needed (any $D - 1$ out of D paths have enough capacity to support each total demand). An optimal routing is the one that minimizes either i) the average number of hops or ii) the maximum number of hops, of all relevant paths between any pair of nodes $p, q \in S \subset X$. For each commodity, the relevant paths are the best (that is, those with fewer number of hops) Δ paths (with $1 \leq \Delta \leq D$). Due to the way total demand is protected, under Path Diversity we have $\Delta = D$ and, under Path Protection, we have $\Delta = D - 1$.

Usually, traffic engineering problems deal mainly with the minimization of network link loads as a means to improve the capacity of the network to recover from failures. In the models studied in our work, the survivability requirements are guaranteed by the solution and optimizing the hop count is the main objective. The recent Demand-wise Shared Protection mechanism [4] is an example of a more efficient mechanism guaranteeing that if a single path fails, there is a required percentage of demand that is protected. More recently, protection mechanisms have been proposed for modern packet switched networks [5, 6] which are straightforward adaptations of the optical $1+1$ dedicated and $1:n$ shared mechanisms.

In a previous paper [3], we have considered aggregated models for the problems addressed here. Aggregated models use a single set of variables for all paths of a commodity. In this paper we also study disaggregated models (that have one set of variables for each path in each commodity). The objective function of the proposed traffic engineering problems is easy to model with the variables of the disaggregated models. However, the number of variables in a disaggregated model increases with the value of D and this might lead to substantially different CPU times for solving the models when needing to analyze the solutions given by different scenarios (for instance, comparing solutions with $D=2$ and $D=3$) which might be a disadvantage when compared with the models in [3].

Our study will focus on these models for the different cases, the theoretical relationships between the two classes of models as well as an empiric comparison of their effectiveness to solve the proposed traffic engineering problems. Since, in general, "disaggregation" is a useful tool for tightening the linear programming relaxation of integer linear programming models, our expectation is that our study will provide insight for tightening the models previously presented in [3]. Unfortunately, our results will show that for the problems under study there is no gain in using the disaggregated models. The reason for this is not only due to the increase on the number of variables of the disaggregated models but is also due to the fact that the linear programming bound of the disaggregated models is not better (in practice) than the linear programming bound of the aggregated models. In fact, for the min-max case we will even show that the linear programming relaxation of the originally proposed disaggregated model does not dominate the linear programming relaxation of the aggregated model.

In the models discussed below, we need to distinguish the direction in which every edge is traversed. Thus, we will also consider an arc set A which contains "arcs" (i,j) and (j,i) for each edge $\{i,j\}$ in U.

2 The Disaggregated and Aggregated Models

In the following, we define a (p,q)-*H-path* as a path with at most H arcs (hops) and we define a path as a sequence $\{(i_1,j_1),...,(i_k,j_k)\}$ of arcs such that $i_1=p$, $j_k=q$ and $j_s=i_{s+1}$ for $s=1,...,k-1$. Following [1], for each commodity consider D sets of binary hop-indexed variables, $z_{ij}^{h,pq,d}$ ($\{i,j\}\in U$; $h=1,...,H$; $p,q\in S$; $d=1,...,D$), indicating whether the d^{th} (p,q)-*H-path* traverses edge $\{i,j\}$ in the direction from i to j, in the h^{th} position. Note that some of the D (p,q)-*H-paths* may contain fewer than H arcs (that is, $z_{jq}^{h,pq,d}=1$ for some d, $j\in X\setminus\{q\}$ and $1\le h\le H-1$), hence, we also consider the so-called "loop" flow variables $z_{qq}^{h,pq,d}$ ($h=2,...,H$) to represent such situations. The constraints for a disaggregated model are then as follows:

$$\sum_{j:(p,j)\in A} z_{pj}^{1,pq,d}=1,\ p,q\in S; d=1,...,D$$
$$\sum_{j:(i,j)\in A} z_{ij}^{h+1,pq,d}-\sum_{j:(j,i)\in A} z_{ji}^{h,pq,d}=0,\ i\ne p; h=1,...,H-1; p,q\in S; d=1,...,D \tag{2.1}$$
$$\sum_{j:(j,q)\in A} z_{jq}^{H,pq,d}=1,\ p,q\in S; d=1,...,D$$

$$\sum_{i:(i,j)\in A}\ \sum_{h=1,...,H}\ \sum_{d=1,...,D} z_{ij}^{h,pq,d}\le 1,\ j\in X\backslash\{p,q\}; p,q\in S \tag{2.2}$$

$$\sum_{p,q\in S}\beta r_{pq}\left(\sum_{h=1,...,H}\ \sum_{d=1,...,D} z_{ij}^{h,pq,d}+\sum_{h=1,...,H}\ \sum_{d=1,...,D} z_{ji}^{h,pq,d}\right)\le b_e,\ e=\{i,j\}\in U \tag{2.3}$$

$$z_{ij}^{h,pq,d}, z_{ji}^{h,pq,d} \in \{0,1\},\ e=\{i,j\} \in U; h=1,...,H; p,q \in S; d=1,..,D$$
$$z_{qq}^{h,pq,d} \in \{0,1\},\ h=2,...,H; p,q \in S; d=1,...,D. \qquad \textbf{(2.4)}$$

As noted in [2], these constraints guarantee that: i) for each commodity (p,q), $p,q \in S$, there exist D node disjoint (p,q)-*H-paths* and ii) there exists enough capacity on each undirected link $\{i,j\}$ to support the required traffic demand according to the survivability mechanisms (Path Diversity or Path Protection, which are specified by the β coefficient: for the first mechanism, since $\Delta = D$, β must be equal to $1/D$ and, for the latter mechanism, $\beta = 1/(D-1)$ since in this case $\Delta = D-1$). For every commodity, we can include the following constraints:

$$\sum_{(i,j)\in A} \sum_{h=1,...,H} z_{ij}^{h,pq,d+1} \geq \sum_{(i,j)\in A} \sum_{h=1,...,H} z_{ij}^{h,pq,d},\ p,q \in S; d=1,...,D-1. \qquad \textbf{(2.5)}$$

These constraints rank the paths in the usual way (that is, the d^{th} path has not more hops that the $d+1^{th}$ path). This ranking helps us to write the objective functions for the cases we will study. In some cases, these inequalities are not necessary since the objective function will guarantee that the required paths satisfy the given ranking (these cases will be pointed out below).

The objective function for the min-average case is straightforward since we want to minimize the number of arcs used by the best Δ paths for every commodity. Thus, we obtain the following general model:

D-Ave-Arc $\quad Min \sum_{d=1,...,\Delta} \sum_{p,q\in S} \sum_{(i,j)\in A} \sum_{h=1,...,H} z_{ij}^{h,pq,d}$

subject to **(2.1)**, **(2.2)**, **(2.3)**, **(2.4)**.

In these models, inequalities (2.5) are not needed since the objective function guarantees that the selected Δ paths are the ones with fewer hops.

For the min-max objective case, we want to compute the number of hops of the worst path of the best Δ paths for all commodities and thus, we will include the ranking inequalities. By adding an integer variable V that computes the length of the worst path among the best Δ paths for all commodities, we can derive the following model.

D-Max-Arc $\quad Min\ V$

subject to **(2.1)**, **(2.2)**, **(2.3)**, **(2.4)**, **(2.5)**

$$\sum_{(i,j)\in A} \sum_{h=1,...,H} z_{ij}^{h,pq,\Delta} \leq V,\ p,q \in S \qquad \textbf{(2.6)}$$

$V \geq 0$ and integer. **(2.7)**

A version of this model with a better linear programming bound will be described in the next section.

The disaggregated models are straightforward to write but have the disadvantage of containing a system of variables for each path in each commodity. Now we consider aggregated models that have a single set of variables for each commodity independently of the required number of disjoint paths. Thus, with fewer variables, it may (hopefully) be easier to solve the corresponding linear programming relaxations and integer linear programs. However, since we cannot distinguish the different paths for each commodity, the objective functions are not as easy to write as they were in the disaggregated models. We will add two sets of extra integer variables that will help us to write these objective functions.

We start by writing the core model that will guarantee the existence of D node disjoint H-paths for each commodity (p,q) under the given capacities (as defined by solving the problem defined in [2]). Following [1], let $w_{ij}^{h,pq}$ represent the number of (p,q)-*H-paths* traversing edge $\{i,j\}$ in the direction from i to j, in the h^{th} position. With these variables, the core model is described as follows:

$$
\begin{aligned}
&\sum_{j:(p,j)\in A} w_{pj}^{1,pq} = D,\ p,q \in S\\
&\sum_{j:(i,j)\in A} w_{ij}^{h+1,pq} - \sum_{j:(j,i)\in A} w_{ji}^{h,pq} = 0,\ i \neq p; h=1,...,H-1; p,q \in S && \textbf{(2.8)}\\
&\sum_{j:(j,q)\in A} w_{jq}^{H,pq} = D,\ p,q \in S\\
&\sum_{i:(i,j)\in A} \sum_{h=1,\ldots,H} w_{ij}^{h,pq} \leq 1,\ j \in X\backslash\{p,q\}; p,q \in S && \textbf{(2.9)}\\
&\sum_{p,q\in S} \beta r_{pq} \left(\sum_{h=1,\ldots,H} w_{ij}^{h,pq} + \sum_{h=1,\ldots,H} w_{ji}^{h,pq} \right) \leq b_e,\ e=\{i,j\} \in U && \textbf{(2.10)}\\
&w_{ij}^{h,pq}, w_{ji}^{h,pq} \in \{0,1,...,D\}\,,\ e=\{i,j\} \in U; h=1,...,H; p,q \in S\\
&w_{qq}^{h,pq} \in \{0,1,...,D\}\,,\ h=2,...,H; p,q \in S. && \textbf{(2.11)}
\end{aligned}
$$

Similarly to the "loop" flow variables used in the disaggregated models, the $w_{qq}^{h,pq}$ $(h=2,...,H)$ loop variables are used to represent situations when some of the D (p,q)-H-*paths* contain fewer than H arcs (that is, $w_{jq}^{h,pq} \geq 1$ for some $j \in X \setminus \{q\}$ and $1 \leq h \leq H-1$).

For the min-average case, let $U^{h,pq}$ denote the number of "interesting" arcs (that is, arcs included in the best Δ paths) in position h for each commodity (p,q). Recall that, for a given position h, all the arcs are interesting for Path Diversity $(\Delta = D)$, and for Path Protection $(\Delta = D-1)$ all the arcs minus 1 are interesting because we can assume that if k arcs are in position h, then one of the arcs is included in the worst path. We obtain a valid model for the min-average version by including the new variables and including constraints that define the new variables:

A-Ave-Arc $\quad Min \sum_{p,q\in S} \sum_{h=1,\ldots,H} U^{h,pq}$

subject to **(2.8)**, **(2.9)**, **(2.10)**, **(2.11)**

$$
\begin{aligned}
&U^{h,pq} \geq \sum_{(i,j)\in A} w_{ij}^{h,pq} - (D-\Delta),\ p,q \in S; h=1,...,H && \textbf{(2.12)}\\
&U^{h,pq} \in \{0,1,...,\Delta\},\ p,q \in S; h=1,...,H. && \textbf{(2.13)}
\end{aligned}
$$

The meaning of the new variables $U^{h,pq}$ is guaranteed by the objective function and the new inequalities (2.12) and (2.13). Note that in the case of Path Diversity, $D-\Delta=0$, we can obtain a much simpler model since the extra variables are not needed. We obtain, instead, the intuitive objective function given by $Min\ \sum_{p,q\in S} \sum_{h=1,\ldots,H} \sum_{(i,j)\in A} w_{ij}^{h,pq}$.

For the case with a min-max objective function, we only need to create a binary variable for each value of h (there is no need to distinguish the information by commodity). Let V^h be a binary variable that indicates whether the number of arcs in position h, for at least one of the paths of all commodities, is greater or equal than $(D-\Delta)+1$. For Path Diversity, the existence of at least one arc in position h $(D-\Delta=0)$, means that there is one path that has at least h arcs. For Path Protection, the existence of at least two arcs in position h, means that there is at least one relevant path that has at least h arcs since, as stated before, one of the arcs may be used by the non interesting path (that is, the path that is not among the best Δ paths). Under each mechanism, we want to compute the number of such positions in order to compute the number of arcs of the worst path.

Note that the arguments used in the previous paragraph are valid only because we know that if any path includes an arc in position h, then it contains arcs in positions $p=1,...,h-1$. This is clearly guaranteed by the hop-dependent flow conservation constraints. We can obtain a valid model for this version of the problem by including the new variables and including constraints that define the new variables:

A-Max-Arc $\quad Min \sum_{h=1,\ldots,H} V^h$

subject to **(2.8)**, **(2.9)**, **(2.10)**, **(2.11)**

$$\sum_{(i,j)\in A} w_{ij}^{h,pq} - (D - \Delta) \leq V^h \times \Delta,\ \ p,q \in S; h = 1,...,H \tag{2.14}$$

$$V^h \in \{0,1\},\ \ h = 1,...,H. \tag{2.15}$$

The meaning of the new variables V^h is guaranteed by the objective function and the new inequalities (2.14) and (2.15). It is also easy to check that the models A-Ave-Arc and A-Max-Arc are equivalent in terms of the linear programming relaxation to the models presented in [3]. In our opinion the models presented here are easier to understand.

3 Comparing the Models

In this section, we present results that compare the linear programming bounds of the models from each of the two classes and for each version of the problem. Details about the proofs will be given at the conference.

Result 1 i) When $D = \Delta$, $v(\textit{D-Ave-Arc}_L) = v(\textit{A-Ave-Arc}_L)$

ii) When $D > \Delta$, it is still open whether

a) $v(\textit{D-Ave-Arc}_L) = v(\textit{A-Ave-Arc}_L)$ or

b) $v(\textit{D-Ave-Arc}_L) \geq v(\textit{A-Ave-Arc}_L)$ and there are instances for which the inequality is strict.

This result is interesting because it tells us that for the case $D = \Delta$ we do not gain anything by using the disaggregated models. For the case $D > \Delta$, the equivalence is still open but our computational experience shows that using the aggregated model is strongly preferable since it uses fewer variables and as stated before, for all the instances tested the two models produce the same linear programming bound.

As our computational experiments will show, the linear programming relaxation of the disaggregated model does not dominate the linear programming relaxation of the aggregated model for the two min-max objective versions. These results have motivated us to produce an enhanced version of the D-Max-Arc version that can be obtained in the following way. We, first, add the following disaggregated version of the ranking inequalities (2.5)

$$\sum_{(i,j)\in A} z_{ij}^{h,pq,\Delta} \leq V^h,\ \ p,q \in S; h = 1,...,H \tag{3.1}$$

Second, we use the binary variables V^h that have been proposed for the aggregated model (note that these variables can also be interpreted as indicating whether, for all the commodities (p,q), there exists a Δ-th path that has one arc on position h), add

$$\sum_{(i,j)\in A} z_{ij}^{h,pq,\Delta} \leq V^h,\ \ p,q \in S; h = 1,...,H \tag{3.2}$$

to the model D-Max-Arc and rewrite the objective function as follows:

$$Min \sum_{h=1,...,H} V^h. \tag{3.3}$$

Note that (3.3) is the objective function of the aggregated model. We denote by ED-Max-Arc the model obtained in this way.

Result 2 When $D \geq \Delta$, it is still open whether

a) $v(ED\text{-}Max\text{-}Arc_L) = v(A\text{-}Max\text{-}Arc_L)$ or

b) $v(ED\text{-}Max\text{-}Arc_L) \geq v(A\text{-}Max\text{-}Arc_L)$ and there are instances for which the inequality is strict.

For all our experiments, the LP bounds were equal between aggregated and disaggregated models. Again, our computational experience shows that using the aggregated model for obtaining the optimal integer solution is strongly preferable since it uses fewer variables.

We have conducted several computational tests based on the network topologies obtained in [2] as the input for the traffic engineering problems. We considered $H = 4$, $|S| = 6$, different traffic demand matrices R (randomly generated using uniform distributions), $D = 2$ and $D = 3$ and both the Path Diversity and Path Protection survivability mechanisms (that is $\Delta = D$ and $\Delta = D - 1$, respectively).

In [3], the main aim of the computational results was to compare the costs of the different traffic engineering problems under the different traffic matrix scenarios. Here, we focus on the gaps given by the linear programming relaxations of the different models and the corresponding CPU times. Due to space considerations, we present average results for all the scenarios tested. We emphasize that the individual gaps are not much different from the average results.Tables 1 and 2 show the gaps for the min-average and min-max problems, respectively. The average gaps are expressed in percentage and the average CPU times (between brackets) in seconds.

Path Diversity ($\mathbf{\Delta = D}$)				Path Protection ($\mathbf{\Delta = D - 1}$)			
$\mathbf{D = 2}$		$\mathbf{D = 3}$		$\mathbf{D = 2}$		$\mathbf{D = 3}$	
D-Ave-Arc	A-Ave-Arc	D-Ave-Arc	A-Ave-Arc	D-Ave-Arc	A-Ave-Arc	D-Ave-Arc	A-Ave-Arc
2.5% (0.01)	2.5% (0.01)	2.3% (0.03)	2.3% (0.01)	0.3% (0.01)	0.3% (0.01)	2.0% (0.03)	2.0% (0.1)

Table 1: Computational results of the D-Ave-Arc and A-Ave-Arc models.

Path Diversity ($\mathbf{\Delta = D}$)				Path Protection ($\mathbf{\Delta = D - 1}$)			
$\mathbf{D = 2}$		$\mathbf{D = 3}$		$\mathbf{D = 2}$		$\mathbf{D = 3}$	
ED-Max-Arc	A-Max-Arc	ED-Max-Arc	A-Max-Arc	ED-Max-Arc	A-Max-Arc	ED-Max-Arc	A-Max-Arc
14.9% (0.01)	14.9% (0.01)	22.2% (0.03)	22.2% (0.01)	0% (0.02)	0% (0.01)	16.7% (0.04)	16.7% (0.01)

Table 2: Computational results of the D-Max-Arc, ED-Max-Arc and A-Max-Arc models.

The reported gaps show different magnitudes of the gaps for the different versions of the problem and give the idea that the min-average case is easier to solve than the min-max case. The results for the min-max case are quite curious when we compare $D = 2$ with $D = 3$ and the Path Protection case with the Path Diversity case. Interesting is also to compare the zero gaps for all instances tested with Path Protection and $D = 2$ with the remaining three cases. The CPU times times of the disaggregated model are slightly larger than the CPU times for the aggregated models. However, when attempting to solve the ILPs, the aggregated models are preferable since they have fewer variables. For instance for one case with $D = 3$ the agregated model took 0.42 seconds to obtain the optimal solution while the disaggregated took 12493 seconds. Future computational experiments include testing the models presented in this paper with $H = 3, 4, 5$ on different (denser) network topologies. We hope that looking at the linear programming relaxation optimal solutions will shed some light on the still open issues of results 1 and 2. In spite of wanting to know whether those inequalities are strict or not, in practice, the aggregated models seem to be much more efficient. Therefore, as future work, it would also be interesting to try and close the linear programming gaps obtained in the min-max case.

References

[1] Gouveia, L., Patrício, P.F., Sousa, A.F., *Compact models for hop-constrained node survivable network design*, in *Telecommunications Network Planning: Innovations in Pricing, Network Design and Management*, G. Anandaligam and S. Raghavan eds., Springer, 2006, pp. 167–180.

[2] Gouveia, L., Patrício, P.F., Sousa, A.F., *Hop-Constrained Node Survivable Network Design: An Application to MPLS over WDM*, Networks and Spatial Economics **8** (2008), No. 1, pp. 3–21.

[3] Gouveia, L., Patrício, P.F., Sousa, A.F., *Optimal Survivable Routing with a Small Number of Hops*, in *Telecommunications Modeling, Policy, and Technology*, S. Raghavan, Bruce Golden and Edward Wasil eds., Springer, 2008, pp. 253–274.

[4] Koster, A., Zymolka A, Jager, M., and Hulsermann, R., *Demand-wise shared protection for meshed optical networks*, Journal of Network and Systems Management **13** (2005), No. 1, pp. 35–55.

[5] Mannie, E., *Generalized Multi-Protocol Label Switching (GMPLS) Architecture*, RFC 3945, Internet Engineering Task Force, October 2004.

[6] Sharma, V., Hellstrand, F., *Framework for Multi-Protocol Label Switching (MPLS) – based Recovery*, RFC 3469, Internet Engineering Task Force, February 2003.